Introduction to the ZX-calculus

Miriam Backens (they/them)
Inria & Loria, Nancy

QCOMICAL school, 3rd November 2025

Outline

Outline

Motivation: quantum circuit optimisation

Quantum computational resources are limited, so we need to use them efficiently.

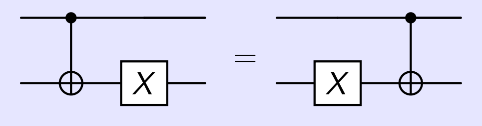
- Given a quantum circuit, can we find a more efficient circuit that describes the same linear map?

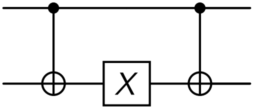
Motivation: quantum circuit optimisation

Quantum computational resources are limited, so we need to use them efficiently.

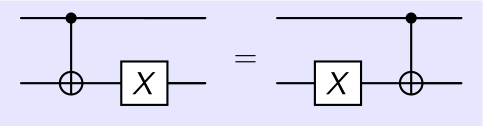
- Given a quantum circuit, can we find a more efficient circuit that describes the same linear map?
- If someone gives you a circuit and claims it is a more efficient version of the circuit you want to run, how can you check that the two circuits really describe the same linear map?

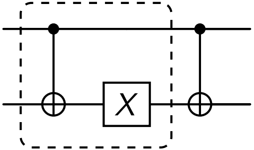
Rewriting quantum circuits



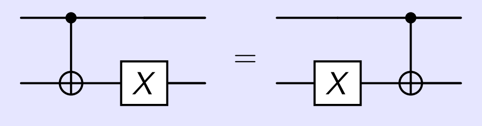


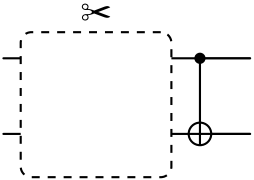
Rewriting quantum circuits



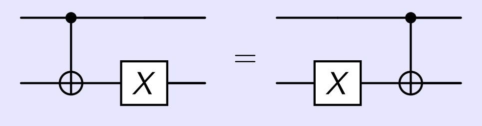


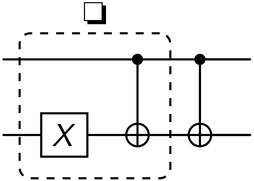
Rewriting quantum circuits



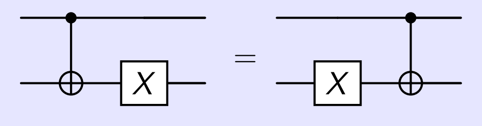


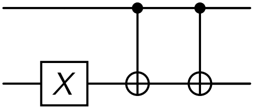
Rewriting quantum circuits



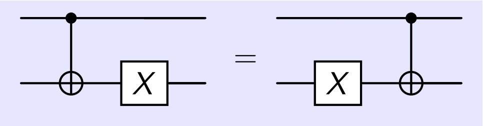


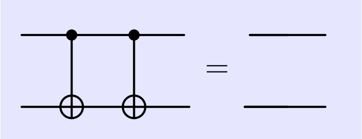
Rewriting quantum circuits

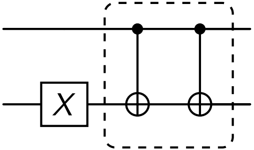




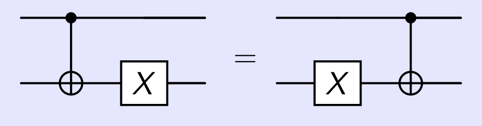
Rewriting quantum circuits

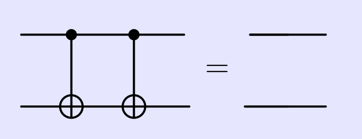


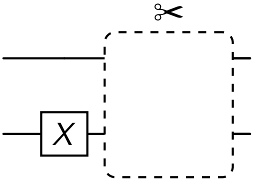




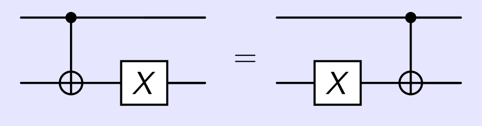
Rewriting quantum circuits

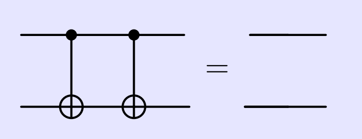


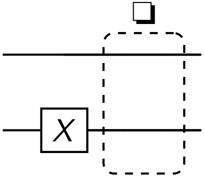




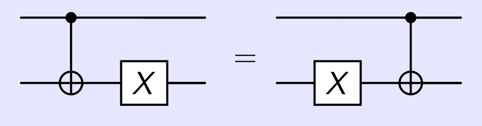
Rewriting quantum circuits

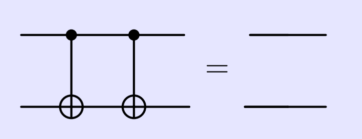


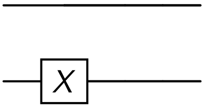




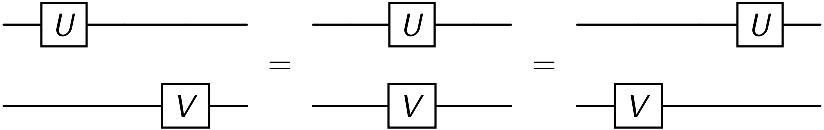
Rewriting quantum circuits



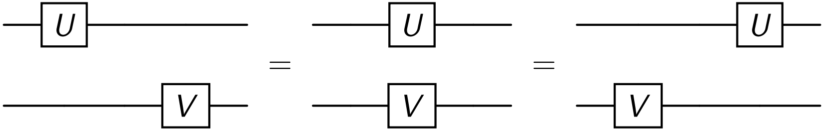




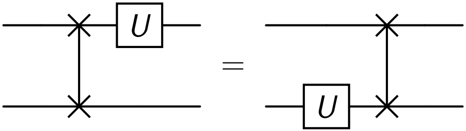
Some circuit equations are particularly intuitive



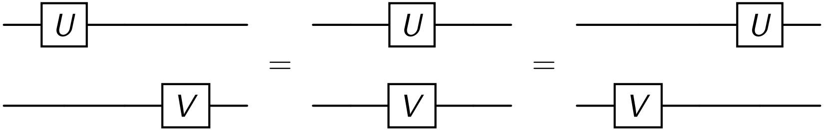
Some circuit equations are particularly intuitive



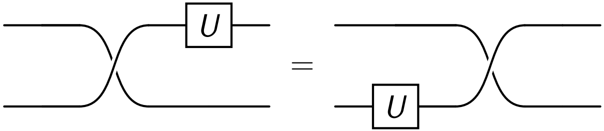
Notation can make a difference, for example the two symbols for the SWAP gate



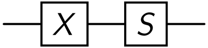
Some circuit equations are particularly intuitive



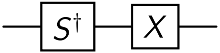
Notation can make a difference, for example the two symbols for the SWAP gate



Equality up to scalar factor



$$\rightsquigarrow \quad \begin{pmatrix} 1 & 0 \\ 0 & i \end{pmatrix} \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ i & 0 \end{pmatrix}$$



$$\rightsquigarrow \quad \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} 1 & 0 \\ 0 & -i \end{pmatrix} = \begin{pmatrix} 0 & -i \\ 1 & 0 \end{pmatrix} = -i \begin{pmatrix} 0 & 1 \\ i & 0 \end{pmatrix}$$

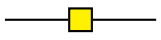
Outline

Outline

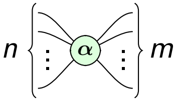
The ZX-calculus components: (mostly) spiders instead of gates

Hadamard gate

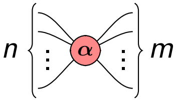
[Coecke & Duncan 2008]



Z-spider

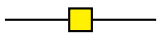


X-spider

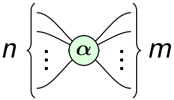


The ZX-calculus components: (mostly) spiders instead of gates

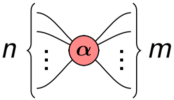
Hadamard gate [Coecke & Duncan 2008]

 $\rightsquigarrow$ $|+\rangle\langle 0| + |-\rangle\langle 1| = \frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$

Z-spider

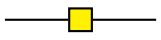


X-spider



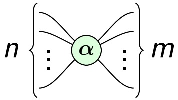
The ZX-calculus components: (mostly) spiders instead of gates

Hadamard gate [Coecke & Duncan 2008]



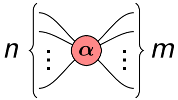
$$\rightsquigarrow \quad |+\rangle\langle 0| + |-\rangle\langle 1| = \frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$$

Z-spider



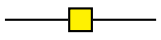
$$\rightsquigarrow \quad |\underbrace{0\ldots0}_{m}\rangle\langle\underbrace{0\ldots0}_{n}| + e^{i\alpha}|\underbrace{1\ldots1}_{m}\rangle\langle\underbrace{1\ldots1}_{n}| = \begin{pmatrix} 1 & \ldots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \ldots & e^{i\alpha} \end{pmatrix}$$

X-spider



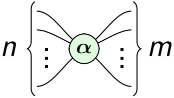
The ZX-calculus components: (mostly) spiders instead of gates

Hadamard gate [Coecke & Duncan 2008]



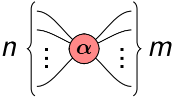
$$\rightsquigarrow \quad |+\rangle\langle 0| + |-\rangle\langle 1| = \frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$$

Z-spider



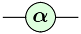
$$\rightsquigarrow \quad |\underbrace{0\ldots0}_{m}\rangle\langle\underbrace{0\ldots0}_{n}| + e^{i\alpha}|\underbrace{1\ldots1}_{m}\rangle\langle\underbrace{1\ldots1}_{n}| = \begin{pmatrix} 1 & \ldots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \ldots & e^{i\alpha} \end{pmatrix}$$

X-spider



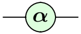
$$\rightsquigarrow \quad |\underbrace{+\ldots+}_{m}\rangle\langle\underbrace{+\ldots+}_{n}| + e^{i\alpha}|\underbrace{-\ldots-}_{m}\rangle\langle\underbrace{-\ldots-}_{n}|$$

A closer look at Z-spiders: unitaries



$$\rightsquigarrow \quad |0\rangle\langle 0| + e^{i\alpha}|1\rangle\langle 1| \quad = \begin{pmatrix} 1 & 0 \\ 0 & e^{i\alpha} \end{pmatrix}$$

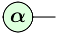
A closer look at Z-spiders: unitaries



$$\rightsquigarrow \quad |0\rangle\langle 0| + e^{i\alpha}\,|1\rangle\langle 1| = \begin{pmatrix} 1 & 0 \\ 0 & e^{i\alpha} \end{pmatrix}$$

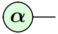
$$\rightsquigarrow \quad |0\rangle\langle 0| - |1\rangle\langle 1| = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} = Z$$

A closer look at Z-spiders: states

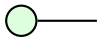


$$\rightsquigarrow \quad |0\rangle + e^{i\alpha}|1\rangle \quad = \begin{pmatrix} 1 \\ e^{i\alpha} \end{pmatrix}$$

A closer look at Z-spiders: states

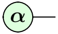


$$\rightsquigarrow \quad |0\rangle + e^{i\alpha}|1\rangle \qquad = \begin{pmatrix} 1 \\ e^{i\alpha} \end{pmatrix}$$

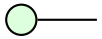


$$\rightsquigarrow \quad |0\rangle + |1\rangle = \sqrt{2}\,|+\rangle \qquad = \begin{pmatrix} 1 \\ 1 \end{pmatrix}$$

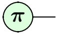
A closer look at Z-spiders: states



$$\rightsquigarrow \quad |0\rangle + e^{i\alpha}|1\rangle = \begin{pmatrix} 1 \\ e^{i\alpha} \end{pmatrix}$$

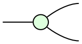


$$\rightsquigarrow \quad |0\rangle + |1\rangle = \sqrt{2}\,|+\rangle = \begin{pmatrix} 1 \\ 1 \end{pmatrix}$$



$$\rightsquigarrow \quad |0\rangle - |1\rangle = \sqrt{2}\,|-\rangle = \begin{pmatrix} 1 \\ -1 \end{pmatrix}$$

A closer look at Z-spiders: the copy map



$$\rightsquigarrow \quad |00\rangle\langle 0| + |11\rangle\langle 1| \quad = \begin{pmatrix} 1 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 1 \end{pmatrix}$$

A closer look at X-spiders: unitaries

$$\rightsquigarrow \qquad |+\rangle\langle+| + e^{i\alpha}\,|-\rangle\langle-| \qquad = e^{i\alpha/2}\begin{pmatrix} \cos\frac{\alpha}{2} & i\sin\frac{\alpha}{2} \\ i\sin\frac{\alpha}{2} & \cos\frac{\alpha}{2} \end{pmatrix}$$

A closer look at X-spiders: unitaries

$$\rightsquigarrow \quad |+\rangle\langle+| + e^{i\alpha}\,|-\rangle\langle-| \quad = e^{i\alpha/2}\begin{pmatrix} \cos\frac{\alpha}{2} & i\sin\frac{\alpha}{2} \\ i\sin\frac{\alpha}{2} & \cos\frac{\alpha}{2} \end{pmatrix}$$

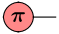
$$\rightsquigarrow \quad |+\rangle\langle+| - |-\rangle\langle-| \quad = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} = X$$

A closer look at X-spiders: states

$\rightsquigarrow \quad |+\rangle + |-\rangle = \sqrt{2}\,|0\rangle \qquad = \sqrt{2}\begin{pmatrix}1\\0\end{pmatrix}$

A closer look at X-spiders: states

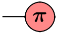
$$\rightsquigarrow \quad |+\rangle + |-\rangle = \sqrt{2}\,|0\rangle \qquad = \sqrt{2}\begin{pmatrix}1\\0\end{pmatrix}$$



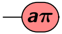
$$\rightsquigarrow \quad |+\rangle - |-\rangle = \sqrt{2}\,|1\rangle \qquad = \sqrt{2}\begin{pmatrix}0\\1\end{pmatrix}$$

A closer look at X-spiders: measurement outcomes

$$\rightsquigarrow \quad \langle +| + \langle -| = \sqrt{2}\,\langle 0| \quad = \sqrt{2}\begin{pmatrix}1 & 0\end{pmatrix}$$

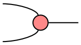


$$\rightsquigarrow \quad \langle +| - \langle -| = \sqrt{2}\,\langle 1| \quad = \sqrt{2}\begin{pmatrix}0 & 1\end{pmatrix}$$



$$\rightsquigarrow \quad \langle +| + (-1)^a\,\langle -| \quad = \sqrt{2}\begin{pmatrix}1-a & a\end{pmatrix}$$

A closer look at X-spiders: the parity map



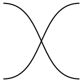
$$\rightsquigarrow \quad |+\rangle\langle ++| + |-\rangle\langle --| \quad = \frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 \end{pmatrix}$$

Wires in the ZX-calculus

 $\rightsquigarrow \quad |0\rangle\langle 0| + |1\rangle\langle 1| \qquad = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$

Wires in the ZX-calculus

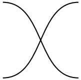
$$\rightsquigarrow \quad |0\rangle\langle 0| + |1\rangle\langle 1| = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$



$$\rightsquigarrow \quad |00\rangle\langle 00| + |10\rangle\langle 01| + |01\rangle\langle 10| + |11\rangle\langle 11| = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Wires in the ZX-calculus

$$\rightsquigarrow \quad |0\rangle\langle 0| + |1\rangle\langle 1| = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

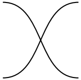


$$\rightsquigarrow \quad |00\rangle\langle 00| + |10\rangle\langle 01| + |01\rangle\langle 10| + |11\rangle\langle 11| = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

$$\rightsquigarrow \quad \langle 00| + \langle 11| = \begin{pmatrix} 1 & 0 & 0 & 1 \end{pmatrix}$$

Wires in the ZX-calculus

$$\rightsquigarrow \quad |0\rangle\langle 0| + |1\rangle\langle 1| = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

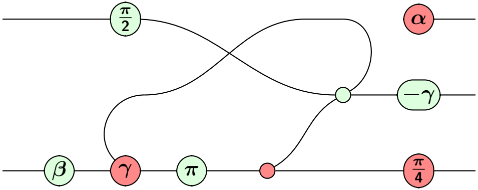


$$\rightsquigarrow \quad |00\rangle\langle 00| + |10\rangle\langle 01| + |01\rangle\langle 10| + |11\rangle\langle 11| = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

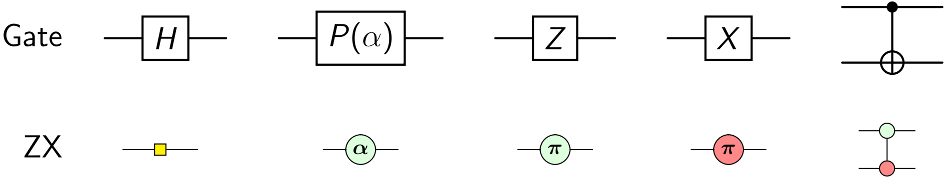
$$\rightsquigarrow \quad \langle 00| + \langle 11| = \begin{pmatrix} 1 & 0 & 0 & 1 \end{pmatrix}$$

$$\rightsquigarrow \quad |00\rangle + |11\rangle = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 1 \end{pmatrix}$$

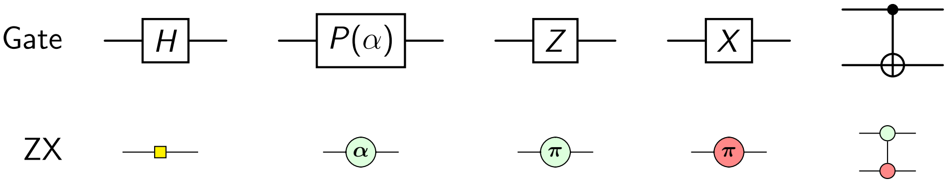
Building ZX-diagrams

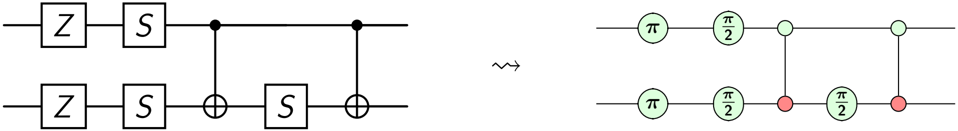


Translating circuits into ZX-diagrams



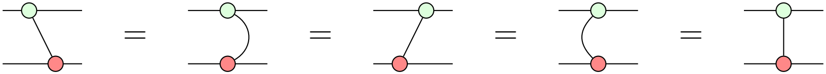
Translating circuits into ZX-diagrams



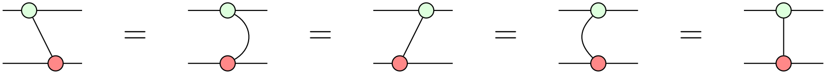


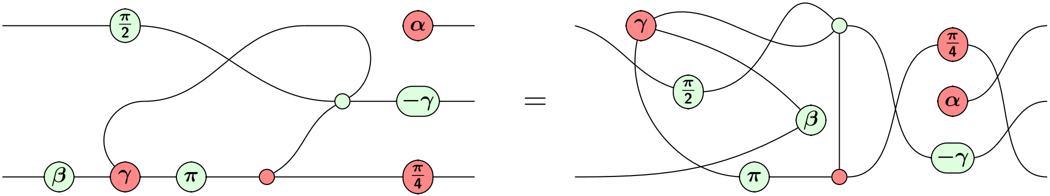
Outline

Only connectivity matters

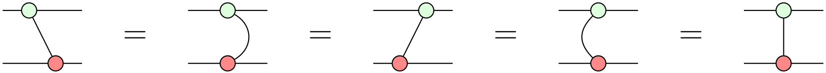


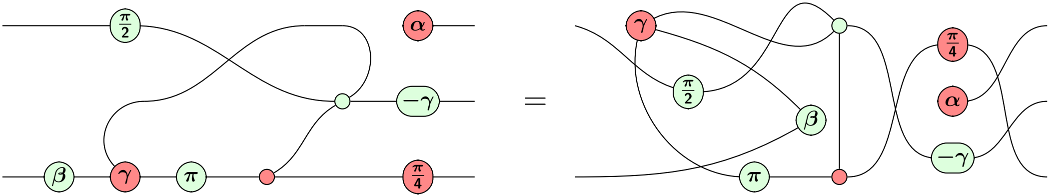
Only connectivity matters





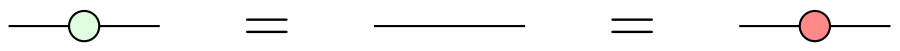
Only connectivity matters



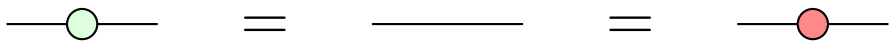


This is made mathematically rigorous using monoidal category theory.

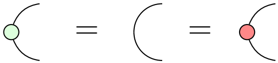
The identity rule



The identity rule

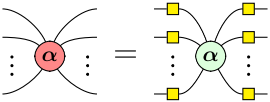


Because only connectivity matters, we also get:

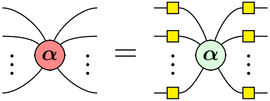


and

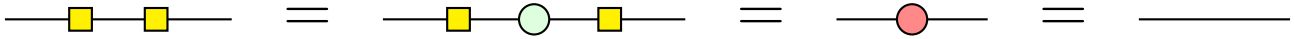
The colour change rule



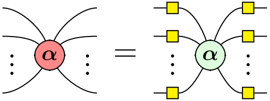
The colour change rule



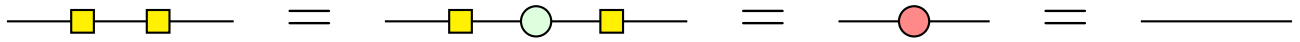
Two Hadamard gates in a row cancel:



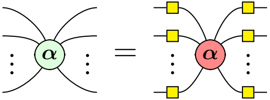
The colour change rule



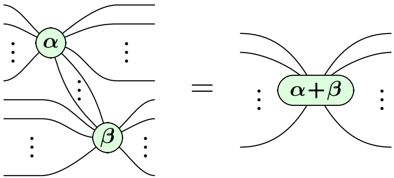
Two Hadamard gates in a row cancel:



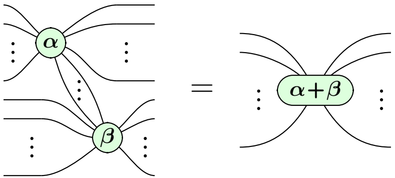
All diagram equations hold with Z- and X-spiders swapped, for example:



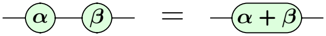
The spider rule



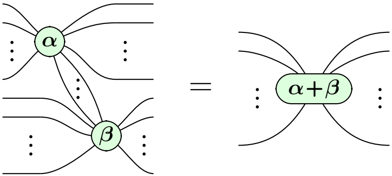
The spider rule



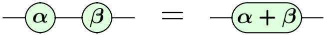
For example:

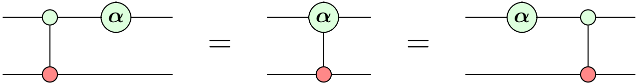


The spider rule

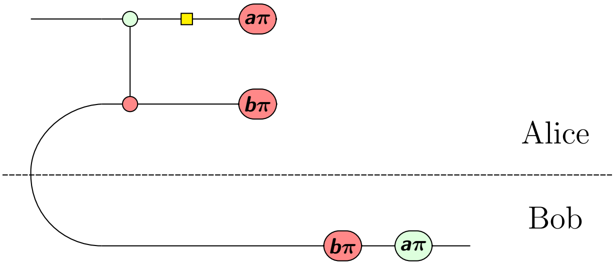


For example:

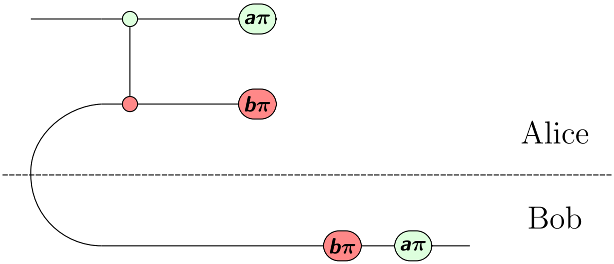




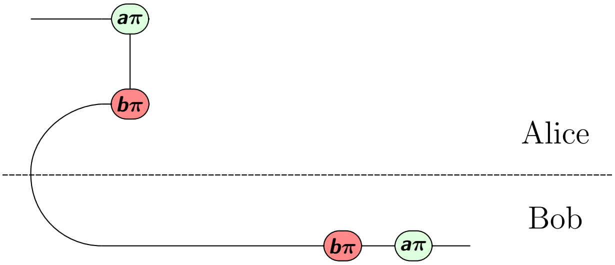
Example: quantum teleportation



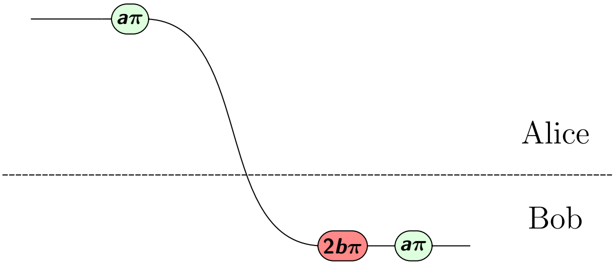
Example: quantum teleportation



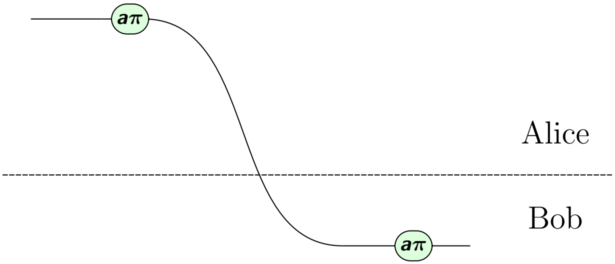
Example: quantum teleportation



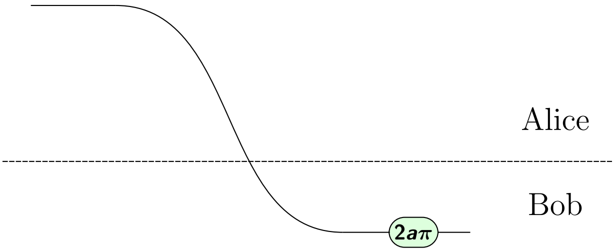
Example: quantum teleportation



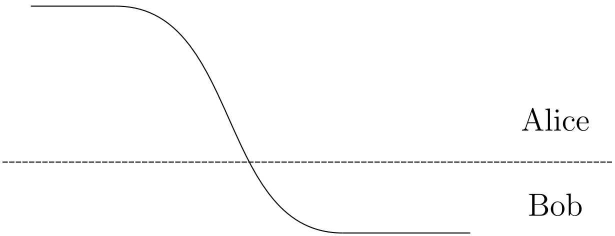
Example: quantum teleportation



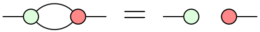
Example: quantum teleportation



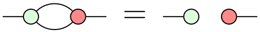
Example: quantum teleportation



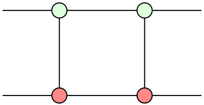
The Hopf rule



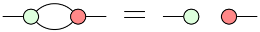
The Hopf rule



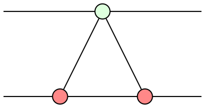
Example: two CNOTs cancel



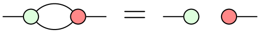
The Hopf rule



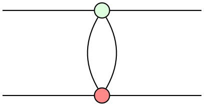
Example: two CNOTs cancel



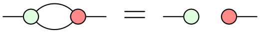
The Hopf rule



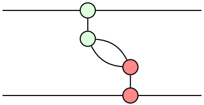
Example: two CNOTs cancel



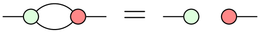
The Hopf rule



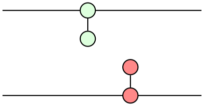
Example: two CNOTs cancel



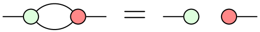
The Hopf rule



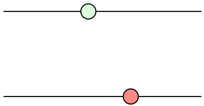
Example: two CNOTs cancel



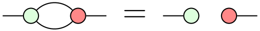
The Hopf rule



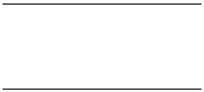
Example: two CNOTs cancel



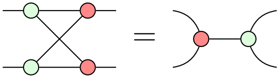
The Hopf rule



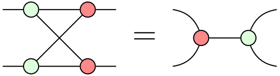
Example: two CNOTs cancel



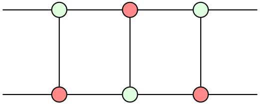
The bialgebra rule



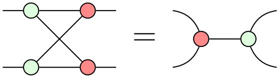
The bialgebra rule



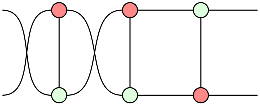
Example: three CNOTs make a SWAP



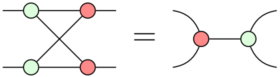
The bialgebra rule



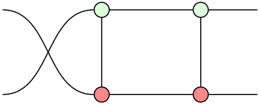
Example: three CNOTs make a SWAP



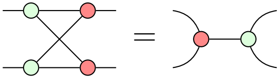
The bialgebra rule



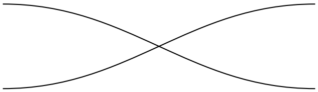
Example: three CNOTs make a SWAP



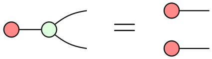
The bialgebra rule



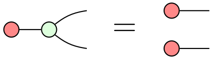
Example: three CNOTs make a SWAP



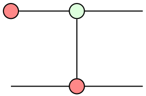
The copy rule



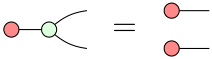
The copy rule



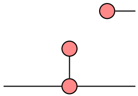
Example: plugging $|0\rangle$ into the control of a CNOT means the gate does nothing



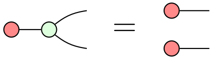
The copy rule



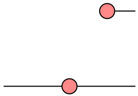
Example: plugging $|0\rangle$ into the control of a CNOT means the gate does nothing



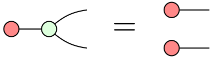
The copy rule



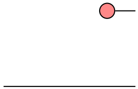
Example: plugging $|0\rangle$ into the control of a CNOT means the gate does nothing



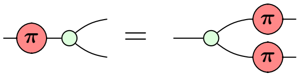
The copy rule



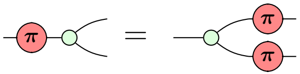
Example: plugging $|0\rangle$ into the control of a CNOT means the gate does nothing



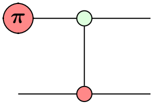
The π-copy rule



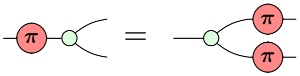
The π-copy rule



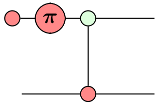
Example: plugging $|1\rangle$ into the control of a CNOT flips the second qubit



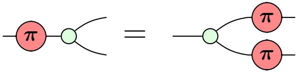
The π-copy rule



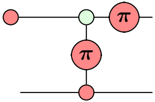
Example: plugging $|1\rangle$ into the control of a CNOT flips the second qubit



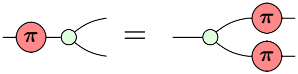
The π-copy rule



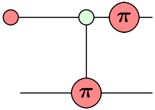
Example: plugging $|1\rangle$ into the control of a CNOT flips the second qubit



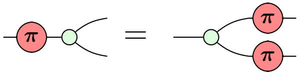
The π-copy rule



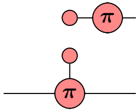
Example: plugging $|1\rangle$ into the control of a CNOT flips the second qubit



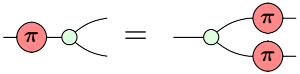
The π-copy rule



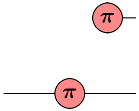
Example: plugging $|1\rangle$ into the control of a CNOT flips the second qubit



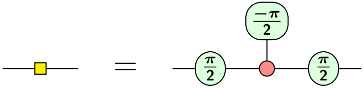
The π-copy rule



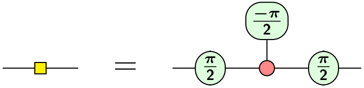
Example: plugging $|1\rangle$ into the control of a CNOT flips the second qubit

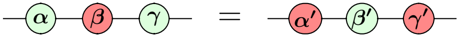


Two more rules about single-qubit operations



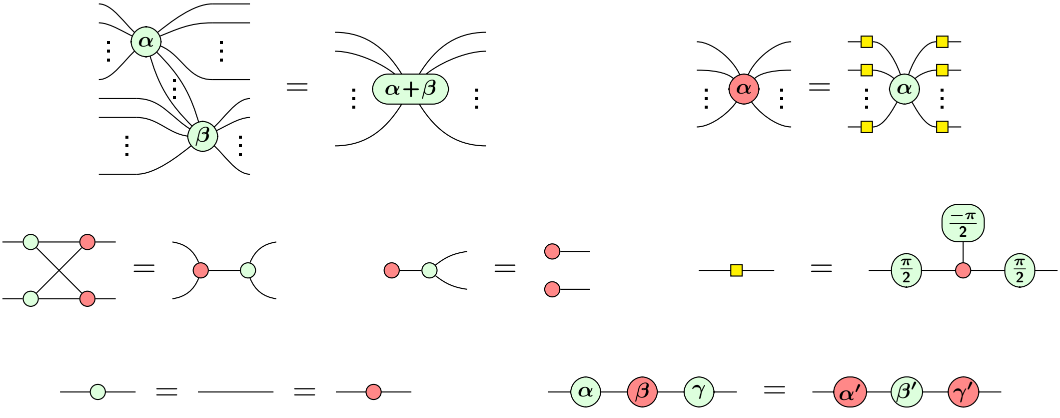
Two more rules about single-qubit operations



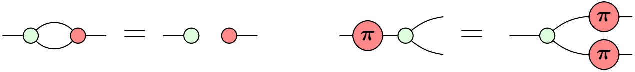


(with a complicated relationship between the phase angles)

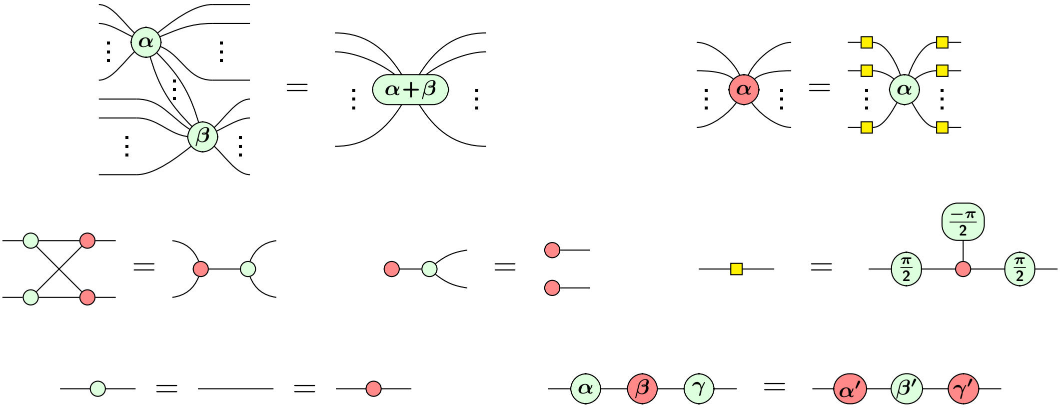
The complete set of (scalar-free) ZX-calculus rewrite rules



Only connectivity matters.

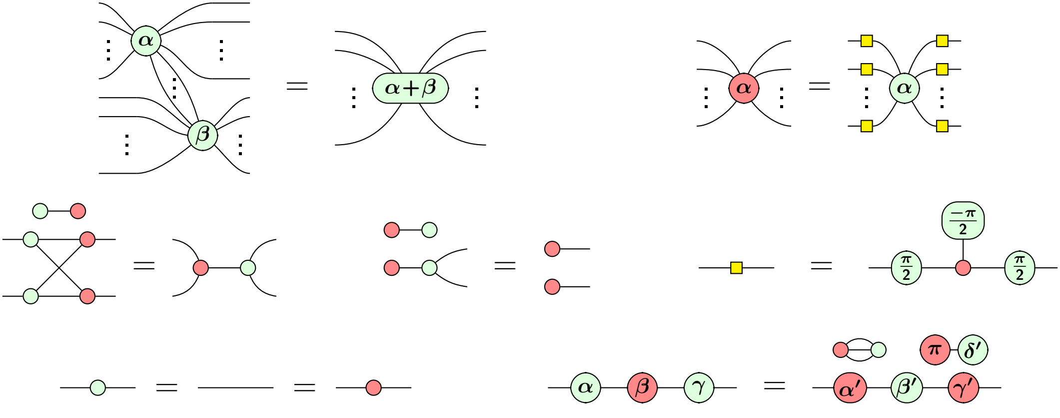


The complete set of (scalar-free) ZX-calculus rewrite rules

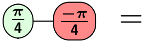


Only connectivity matters.

The complete set of ZX-calculus rewrite rules



Only connectivity matters.



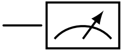
[B. 2014; Ng and Wang 2017; Jeandel, Perdrix, Vilmart 2018; Vilmart 2019]

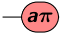
Outline

Outline

Single-qubit measurements

Have already seen computational basis measurements:

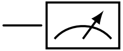


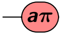


$$\rightsquigarrow \quad \sqrt{2}\begin{pmatrix} a & 1-a \end{pmatrix} = \begin{cases} \sqrt{2}\,\langle 0| & \text{if } a = 0 \\ \sqrt{2}\,\langle 1| & \text{if } a = 1 \end{cases}$$

Single-qubit measurements

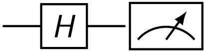
Have already seen computational basis measurements:

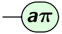




$$\rightsquigarrow \quad \sqrt{2}\begin{pmatrix} a & 1-a \end{pmatrix} = \begin{cases} \sqrt{2}\,\langle 0| & \text{if } a = 0 \\ \sqrt{2}\,\langle 1| & \text{if } a = 1 \end{cases}$$

Other measurement shorthands:

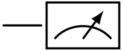


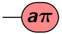


$$\rightsquigarrow \quad \sqrt{2}\begin{pmatrix} 1 & (-1)^a \end{pmatrix} = \begin{cases} \sqrt{2}\,\langle +| & \text{if } a = 0 \\ \sqrt{2}\,\langle -| & \text{if } a = 1 \end{cases}$$

Single-qubit measurements

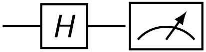
Have already seen computational basis measurements:

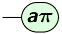




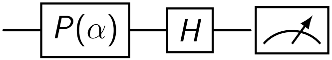
$$\rightsquigarrow \quad \sqrt{2}\begin{pmatrix} a & 1-a \end{pmatrix} = \begin{cases} \sqrt{2}\,\langle 0| & \text{if } a = 0 \\ \sqrt{2}\,\langle 1| & \text{if } a = 1 \end{cases}$$

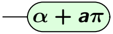
Other measurement shorthands:





$$\rightsquigarrow \quad \sqrt{2}\begin{pmatrix} 1 & (-1)^a \end{pmatrix} = \begin{cases} \sqrt{2}\,\langle +| & \text{if } a = 0 \\ \sqrt{2}\,\langle -| & \text{if } a = 1 \end{cases}$$

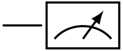


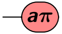


$$\rightsquigarrow \quad \sqrt{2}\begin{pmatrix} 1 & \pm e^{i\alpha} \end{pmatrix} = \langle 0| \pm e^{i\alpha}\,\langle 1|$$

Single-qubit measurements

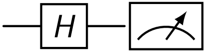
Have already seen computational basis measurements:

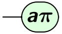




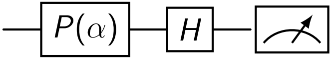
$$\rightsquigarrow \quad \sqrt{2}\begin{pmatrix} a & 1-a \end{pmatrix} = \begin{cases} \sqrt{2}\,\langle 0| & \text{if } a = 0 \\ \sqrt{2}\,\langle 1| & \text{if } a = 1 \end{cases}$$

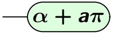
Other measurement shorthands:



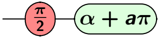


$$\rightsquigarrow \quad \sqrt{2}\begin{pmatrix} 1 & (-1)^a \end{pmatrix} = \begin{cases} \sqrt{2}\,\langle +| & \text{if } a = 0 \\ \sqrt{2}\,\langle -| & \text{if } a = 1 \end{cases}$$





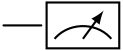
$$\rightsquigarrow \quad \sqrt{2}\begin{pmatrix} 1 & \pm e^{i\alpha} \end{pmatrix} = \langle 0| \pm e^{i\alpha}\,\langle 1|$$

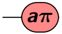
Similarly, can also consider more complicated ZX-diagrams to represent a single measurement, e.g. 

.

Single-qubit measurements

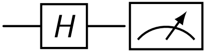
Have already seen computational basis measurements:

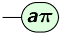




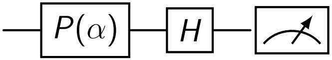
$$\rightsquigarrow \quad \sqrt{2}\begin{pmatrix} a & 1-a \end{pmatrix} = \begin{cases} \sqrt{2}\langle 0| & \text{if } a = 0 \\ \sqrt{2}\langle 1| & \text{if } a = 1 \end{cases}$$

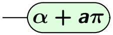
Other measurement shorthands:



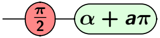


$$\rightsquigarrow \quad \sqrt{2}\begin{pmatrix} 1 & (-1)^a \end{pmatrix} = \begin{cases} \sqrt{2}\langle +| & \text{if } a = 0 \\ \sqrt{2}\langle -| & \text{if } a = 1 \end{cases}$$

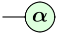




$$\rightsquigarrow \quad \sqrt{2}\begin{pmatrix} 1 & \pm e^{i\alpha} \end{pmatrix} = \langle 0| \pm e^{i\alpha}\langle 1|$$

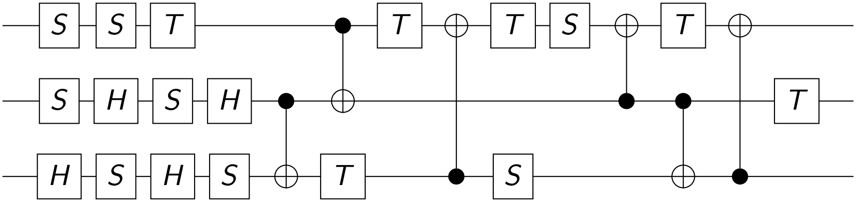
Similarly, can also consider more complicated ZX-diagrams to represent a single measurement, e.g. 

.

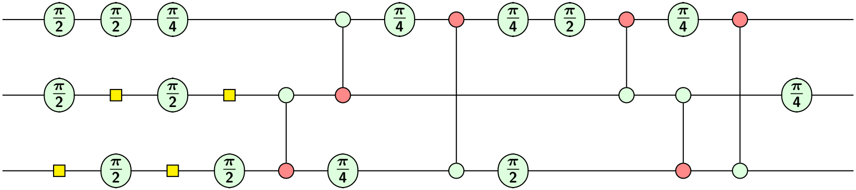
Leave out variable to represent post-selected measurements, e.g. 

.

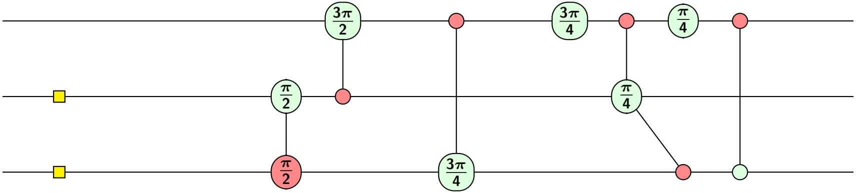
Optimising quantum computations using the ZX-calculus



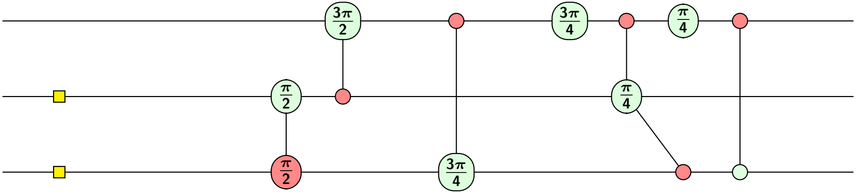
Optimising quantum computations using the ZX-calculus

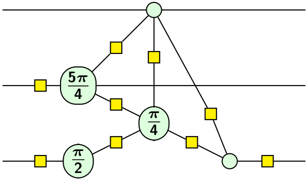


Optimising quantum computations using the ZX-calculus

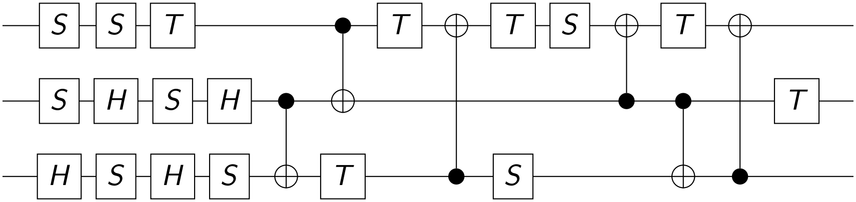


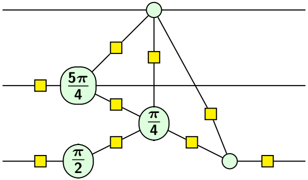
Optimising quantum computations using the ZX-calculus





Optimising quantum computations using the ZX-calculus





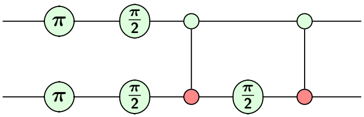
Problem: translating ZX-diagrams to circuits is #P-hard [de Beaudrap et al. 2022]

Two models of quantum computation

Two models of quantum computation

quantum circuit model

[Deutsch 1989]

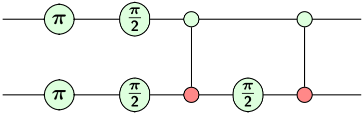


Two models of quantum computation

quantum circuit model

[Deutsch 1989]

- initialise classical state
  $0 \ldots 0$

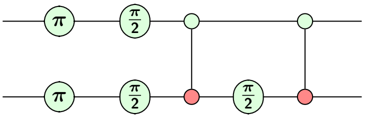


Two models of quantum computation

quantum circuit model

[Deutsch 1989]

- initialise classical state $0 \ldots 0$
- computation driven by (reversible) unitary gates

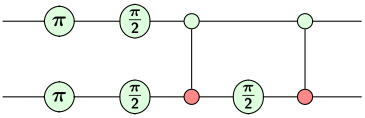


Two models of quantum computation

quantum circuit model

[Deutsch 1989]

- initialise classical state $0 \ldots 0$
- computation driven by (reversible) unitary gates
- simple measurements read out data at end

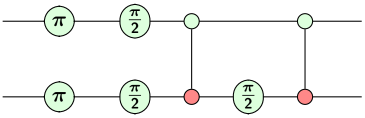


Two models of quantum computation

quantum circuit model

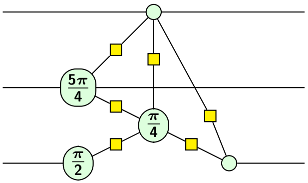
[Deutsch 1989]

- initialise classical state $0 \dots 0$
- computation driven by (reversible) unitary gates
- simple measurements read out data at end



one-way model

[Raussendorf & Briegel 2001]

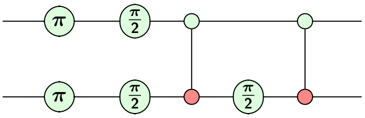


Two models of quantum computation

quantum circuit model

[Deutsch 1989]

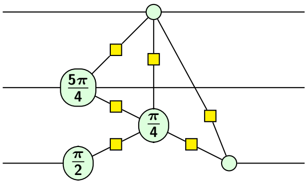
- initialise classical state $0 \dots 0$
- computation driven by (reversible) unitary gates
- simple measurements read out data at end



one-way model

[Raussendorf & Briegel 2001]

- initialise entangled 'graph state' (can be made independent of computation)

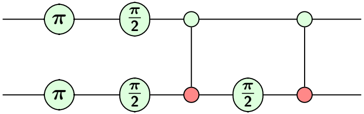


Two models of quantum computation

quantum circuit model

[Deutsch 1989]

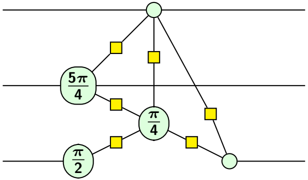
- initialise classical state $0 \ldots 0$
- computation driven by (reversible) unitary gates
- simple measurements read out data at end



one-way model

[Raussendorf & Briegel 2001]

- initialise entangled ‘graph state’ (can be made independent of computation)
- computation driven by successive adaptive single-qubit measurements

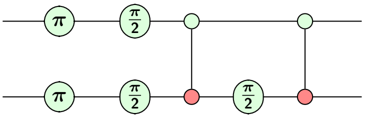


Two models of quantum computation

quantum circuit model

[Deutsch 1989]

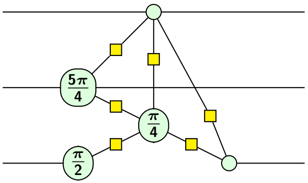
- initialise classical state $0 \ldots 0$
- computation driven by (reversible) unitary gates
- simple measurements read out data at end



one-way model

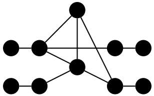
[Raussendorf & Briegel 2001]

- initialise entangled 'graph state' (can be made independent of computation)
- computation driven by successive adaptive single-qubit measurements
- if goal is state preparation, may need Pauli corrections at the end



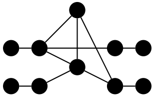
Graph states as ZX-diagrams

Take a simple graph, e.g.

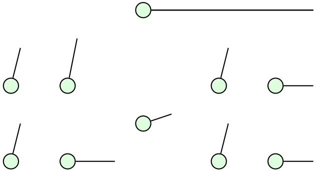


Graph states as ZX-diagrams

Take a simple graph, e.g.

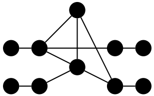


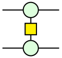
- For each vertex, prepare a qubit in the state $|+\rangle$.

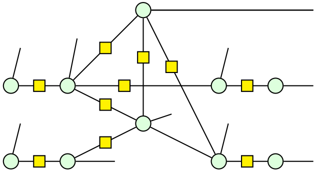


Graph states as ZX-diagrams

Take a simple graph, e.g.

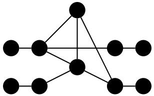


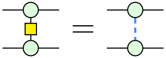
- For each vertex, prepare a qubit in the state $|+\rangle$.
- For each edge, apply a controlled-Z gate  .

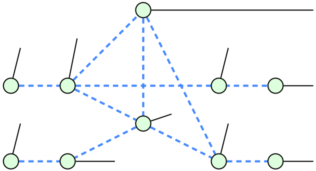


Graph states as ZX-diagrams

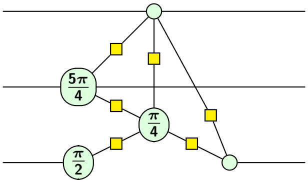
Take a simple graph, e.g.

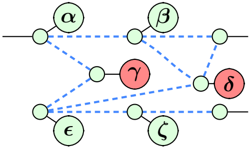


- For each vertex, prepare a qubit in the state $|+\rangle$.
- For each edge, apply a controlled-Z gate  .

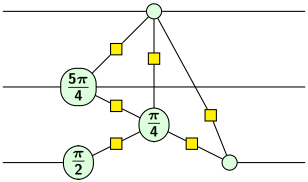


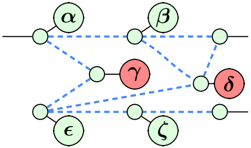
Representing one-way computations in the ZX-calculus





Representing one-way computations in the ZX-calculus

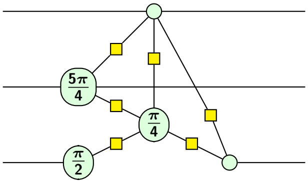


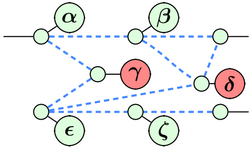


Bring any ZX-calculus diagram into 'MBQC-form' by

- Colour-changing all red spiders to green.

Representing one-way computations in the ZX-calculus

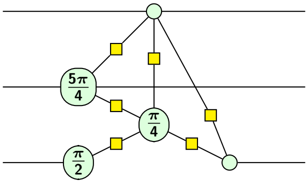


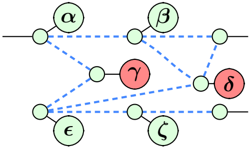


Bring any ZX-calculus diagram into ‘MBQC-form’ by

- Colour-changing all red spiders to green.
- Merging adjacent green spiders.

Representing one-way computations in the ZX-calculus

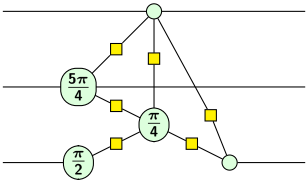


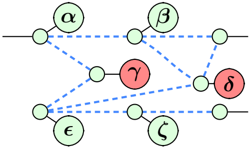


Bring any ZX-calculus diagram into 'MBQC-form' by

- Colour-changing all red spiders to green.
- Merging adjacent green spiders.

Representing one-way computations in the ZX-calculus



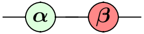


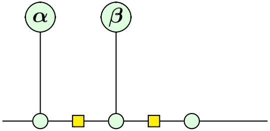
Bring any ZX-calculus diagram into 'MBQC-form' by

- Colour-changing all red spiders to green.
- Merging adjacent green spiders.

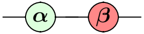
Can allow different measurement effects:

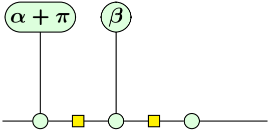
Correcting undesired measurement outcomes

Suppose we want to implement the operation  in the one-way model:



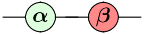
Correcting undesired measurement outcomes

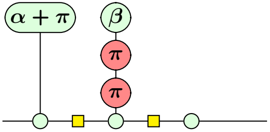
Suppose we want to implement the operation  in the one-way model:



Suppose the first measurement yields the undesired outcome.

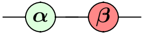
Correcting undesired measurement outcomes

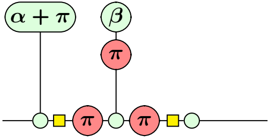
Suppose we want to implement the operation  in the one-way model:



Suppose the first measurement yields the undesired outcome.

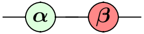
Correcting undesired measurement outcomes

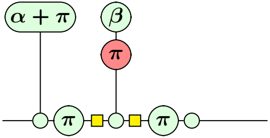
Suppose we want to implement the operation  in the one-way model:



Suppose the first measurement yields the undesired outcome.

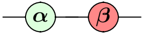
Correcting undesired measurement outcomes

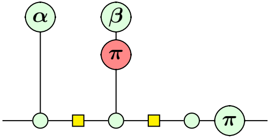
Suppose we want to implement the operation  in the one-way model:



Suppose the first measurement yields the undesired outcome.

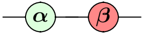
Correcting undesired measurement outcomes

Suppose we want to implement the operation  in the one-way model:

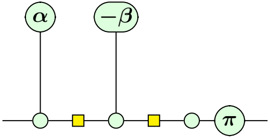


Suppose the first measurement yields the undesired outcome.

Correcting undesired measurement outcomes

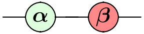
Suppose we want to implement the operation 

in the one-way model:

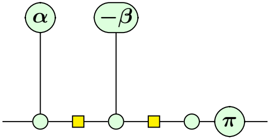


Suppose the first measurement yields the undesired outcome. This can be compensated by flipping the second angle to $-\beta$ and applying Z to the output.

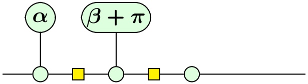
Correcting undesired measurement outcomes

Suppose we want to implement the operation 

in the one-way model:

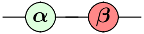


Suppose the first measurement yields the undesired outcome. This can be compensated by flipping the second angle to $-\beta$ and applying Z to the output.

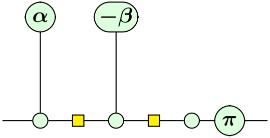


Similarly, suppose the second measurement yields the undesired outcome.

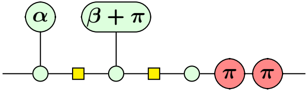
Correcting undesired measurement outcomes

Suppose we want to implement the operation 

in the one-way model:

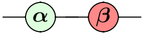


Suppose the first measurement yields the undesired outcome. This can be compensated by flipping the second angle to $-\beta$ and applying Z to the output.

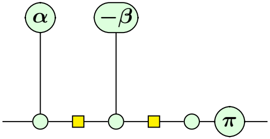


Similarly, suppose the second measurement yields the undesired outcome.

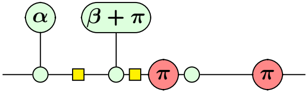
Correcting undesired measurement outcomes

Suppose we want to implement the operation 

in the one-way model:

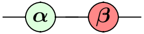


Suppose the first measurement yields the undesired outcome. This can be compensated by flipping the second angle to $-\beta$ and applying Z to the output.

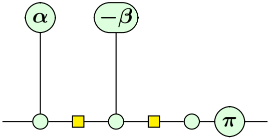


Similarly, suppose the second measurement yields the undesired outcome.

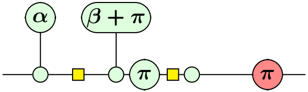
Correcting undesired measurement outcomes

Suppose we want to implement the operation 

in the one-way model:

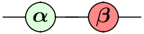


Suppose the first measurement yields the undesired outcome. This can be compensated by flipping the second angle to $-\beta$ and applying Z to the output.

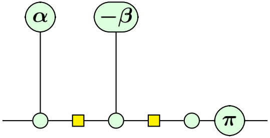


Similarly, suppose the second measurement yields the undesired outcome.

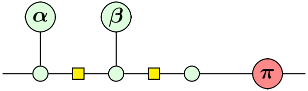
Correcting undesired measurement outcomes

Suppose we want to implement the operation 

in the one-way model:

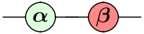


Suppose the first measurement yields the undesired outcome. This can be compensated by flipping the second angle to $-\beta$ and applying Z to the output.

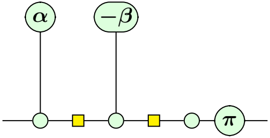


Similarly, suppose the second measurement yields the undesired outcome. This can be corrected by applying X to the output, without affecting the first measurement.

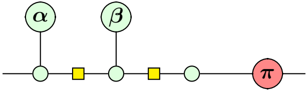
Correcting undesired measurement outcomes

Suppose we want to implement the operation 

in the one-way model:



Suppose the first measurement yields the undesired outcome. This can be compensated by flipping the second angle to $-\beta$ and applying Z to the output.



Similarly, suppose the second measurement yields the undesired outcome. This can be corrected by applying X to the output, without affecting the first measurement. The two correction procedures are compatible with each other.

Determinism in the one-way model allows circuit extraction

If for a one-way computation, there exist:

- a partial order on the qubits, and

then the computation can be implemented deterministically (in a suitably strong sense). [Danos & Kashefi, 2006; Browne et al. 2007]

Determinism in the one-way model allows circuit extraction

If for a one-way computation, there exist:

- a partial order on the qubits, and
- a correction procedure for each measured qubit, which has non-trivial side effects only on qubits that come later in the partial order

then the computation can be implemented deterministically (in a suitably strong sense). [Danos & Kashefi, 2006; Browne et al. 2007]

Determinism in the one-way model allows circuit extraction

If for a one-way computation, there exist:

- a partial order on the qubits, and
- a correction procedure for each measured qubit, which has non-trivial side effects only on qubits that come later in the partial order

then the computation can be implemented deterministically (in a suitably strong sense).
[Danos & Kashefi, 2006; Browne et al. 2007]

Determinism in the one-way model allows circuit extraction

If for a one-way computation, there exist:

- a partial order on the qubits, and
- a correction procedure for each measured qubit, which has non-trivial side effects only on qubits that come later in the partial order

then the computation can be implemented deterministically (in a suitably strong sense). [Danos & Kashefi, 2006; Browne et al. 2007]

The pair of partial order and correction procedure is called a flow.

Determinism in the one-way model allows circuit extraction

If for a one-way computation, there exist:

- a partial order on the qubits, and
- a correction procedure for each measured qubit, which has non-trivial side effects only on qubits that come later in the partial order

then the computation can be implemented deterministically (in a suitably strong sense). [Danos & Kashefi, 2006; Browne et al. 2007]

The pair of partial order and correction procedure is called a flow.

There exists an efficient algorithm which 'extracts a circuit' from any ZX-diagram that corresponds to a one-way computation with flow.

[Duncan et al. 2020; B. et al. 2021; Simmons 2021]

Outline

The quantum bit-flip code

Goal: want to protect quantum information against bit flip error $X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$

The quantum bit-flip code

Goal: want to protect quantum information against bit flip error $X = \left(\begin{smallmatrix} 0 & 1 \\ 1 & 0 \end{smallmatrix}\right)$

Encode single-qubit states via $|0\rangle \mapsto |000\rangle$ and $|1\rangle \mapsto |111\rangle$, so

$$\alpha\,|0\rangle + \beta\,|1\rangle \quad \mapsto \quad \alpha\,|000\rangle + \beta\,|111\rangle$$

The quantum bit-flip code

Goal: want to protect quantum information against bit flip error $X = \left(\begin{smallmatrix} 0 & 1 \\ 1 & 0 \end{smallmatrix}\right)$

Encode single-qubit states via $|0\rangle \mapsto |000\rangle$ and $|1\rangle \mapsto |111\rangle$, so

$$\alpha|0\rangle + \beta|1\rangle \quad \mapsto \quad \alpha|000\rangle + \beta|111\rangle$$

Measuring parities does not destroy the superposition:

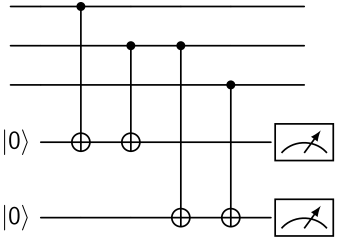
The quantum bit-flip code

Goal: want to protect quantum information against bit flip error $X = \left(\begin{smallmatrix} 0 & 1 \\ 1 & 0 \end{smallmatrix}\right)$

Encode single-qubit states via $|0\rangle \mapsto |000\rangle$ and $|1\rangle \mapsto |111\rangle$, so

$$\alpha\,|0\rangle + \beta\,|1\rangle \quad \mapsto \quad \alpha\,|000\rangle + \beta\,|111\rangle$$

Measuring parities does not destroy the superposition:



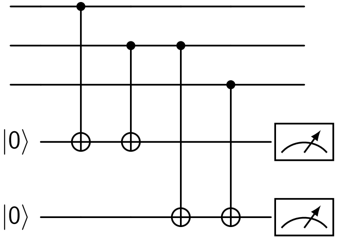
The quantum bit-flip code

Goal: want to protect quantum information against bit flip error $X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$

Encode single-qubit states via $|0\rangle \mapsto |000\rangle$ and $|1\rangle \mapsto |111\rangle$, so

$$\alpha\,|0\rangle + \beta\,|1\rangle \quad \mapsto \quad \alpha\,|000\rangle + \beta\,|111\rangle$$

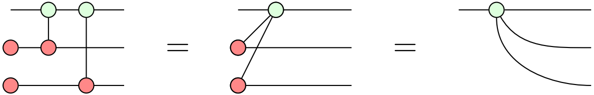
Measuring parities does not destroy the superposition:



$$\alpha\,|000\rangle + \beta\,|111\rangle \rightsquigarrow 00$$
$$\alpha\,|100\rangle + \beta\,|011\rangle \rightsquigarrow 10$$
$$\alpha\,|010\rangle + \beta\,|101\rangle \rightsquigarrow 11$$
$$\alpha\,|001\rangle + \beta\,|110\rangle \rightsquigarrow 01$$

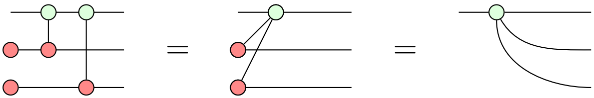
The bit flip code in the ZX-calculus

Encoder:

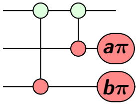


The bit flip code in the ZX-calculus

Encoder:

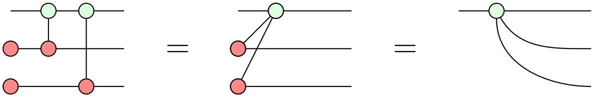


Use the adjoint of this map to decode; this will involve measurements:

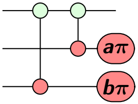


The bit flip code in the ZX-calculus

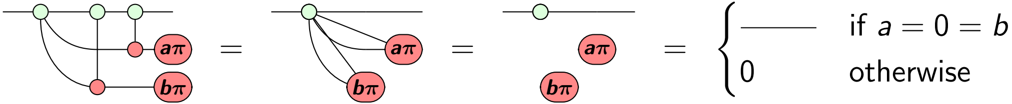
Encoder:



Use the adjoint of this map to decode; this will involve measurements:

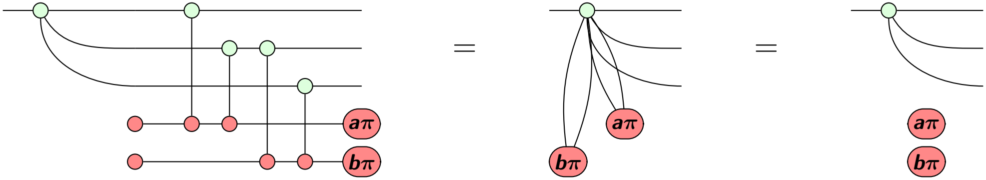


Encoding and decoding when there are no errors yields outcomes $a = 0 = b$:



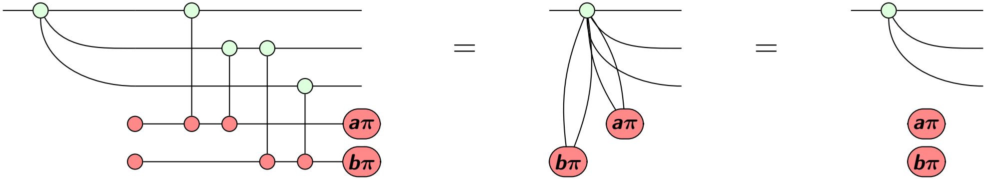
The detectors

No errors:

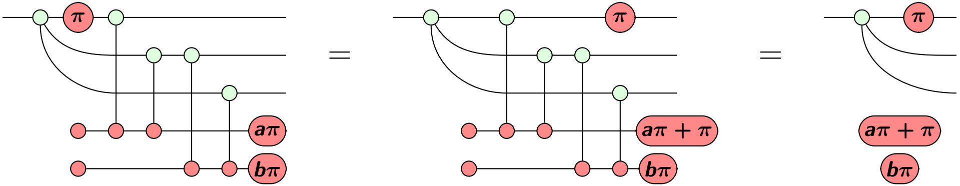


The detectors

No errors:



A bit-flip error on the first qubit changes the first parity measurement:



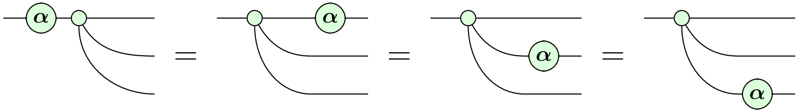
Fault-tolerant computation on encoded data

Transversality: each physical gate should only touch one of the encoded qubits to avoid spreading errors.

Fault-tolerant computation on encoded data

Transversality: each physical gate should only touch one of the encoded qubits to avoid spreading errors.

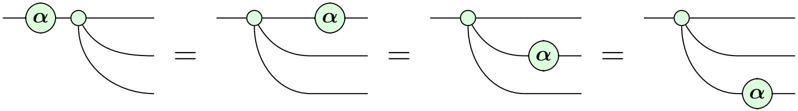
Logical phase gate:



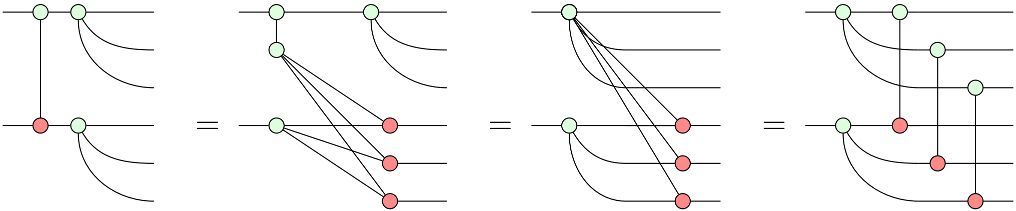
Fault-tolerant computation on encoded data

Transversality: each physical gate should only touch one of the encoded qubits to avoid spreading errors.

Logical phase gate:

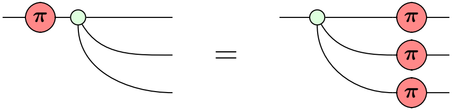


Logical controlled-X gate:



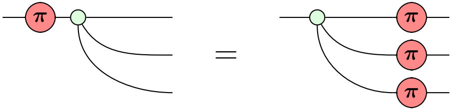
Fault-tolerant computation on encoded data

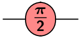
Logical X gate:

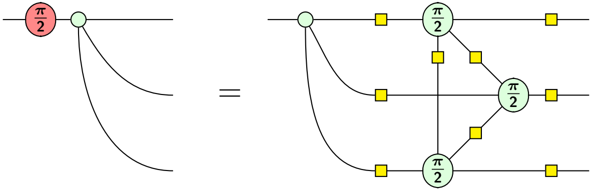


Fault-tolerant computation on encoded data

Logical X gate:



Other X-rotations are not transverse, e.g. for :

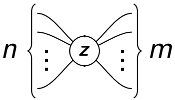


Need to use more complicated techniques to implement these gates fault-tolerantly.

Outline

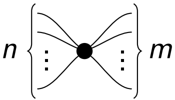
The ZW-calculus

Z spider, $z \in \mathbb{C}$ [Hadzihasanovic 2015; Hadzihasanovic, Ng, Wang 2018]

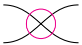


$$\rightsquigarrow \quad |\underbrace{0\ldots0}_{m}\rangle\langle\underbrace{0\ldots0}_{n}| + z|\underbrace{1\ldots1}_{m}\rangle\langle\underbrace{1\ldots1}_{n}| = \begin{pmatrix} 1 & \ldots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \ldots & z \end{pmatrix}$$

W pseudo-spider

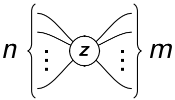


Fermionic swap



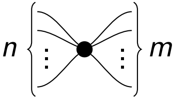
The ZW-calculus

Z spider, $z \in \mathbb{C}$ [Hadzihasanovic 2015; Hadzihasanovic, Ng, Wang 2018]



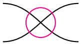
$$\rightsquigarrow \quad |\underbrace{0\dots0}_{m}\rangle\langle\underbrace{0\dots0}_{n}| + z|\underbrace{1\dots1}_{m}\rangle\langle\underbrace{1\dots1}_{n}| = \begin{pmatrix} 1 & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & z \end{pmatrix}$$

W pseudo-spider



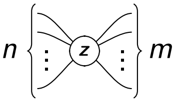
$$\rightsquigarrow \quad \sum |\mathbf{x}\rangle\langle\mathbf{y}|, \qquad \text{where } \mathbf{x} \in \{0,1\}^m, \mathbf{y} \in \{0,1\}^n, |\mathbf{xy}| = 1$$

Fermionic swap



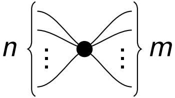
The ZW-calculus

Z spider, $z \in \mathbb{C}$ [Hadzihasanovic 2015; Hadzihasanovic, Ng, Wang 2018]



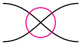
$$\rightsquigarrow \quad |\underbrace{0\dots0}_{m}\rangle\langle\underbrace{0\dots0}_{n}| + z|\underbrace{1\dots1}_{m}\rangle\langle\underbrace{1\dots1}_{n}| = \begin{pmatrix} 1 & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & z \end{pmatrix}$$

W pseudo-spider



$$\rightsquigarrow \quad \sum |\mathbf{x}\rangle\langle\mathbf{y}|\,, \qquad \text{where } \mathbf{x} \in \{0,1\}^m, \mathbf{y} \in \{0,1\}^n, |\mathbf{xy}| = 1$$

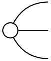
Fermionic swap

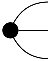


$$\rightsquigarrow \quad |00\rangle\langle00| + |01\rangle\langle10| + |10\rangle\langle01| - |11\rangle\langle11| = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & -1 \end{pmatrix}$$

Some properties of the ZW-calculus

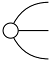
The two three-legged spiders of the ZW-calculus represent the two inequivalent types of 3-qubit entangled states:

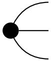
 $\rightsquigarrow \quad |000\rangle + |111\rangle$

 $\rightsquigarrow \quad |001\rangle + |010\rangle + |100\rangle$

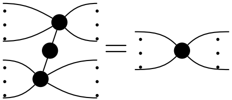
Some properties of the ZW-calculus

The two three-legged spiders of the ZW-calculus represent the two inequivalent types of 3-qubit entangled states:

 $\rightsquigarrow \quad |000\rangle + |111\rangle$

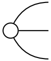
 $\rightsquigarrow \quad |001\rangle + |010\rangle + |100\rangle$

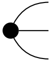
W is a pseudo-spider because its 'spider fusion' rule is slightly different:



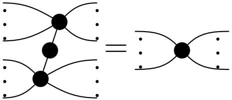
Some properties of the ZW-calculus

The two three-legged spiders of the ZW-calculus represent the two inequivalent types of 3-qubit entangled states:

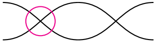
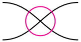
 $\rightsquigarrow \quad |000\rangle + |111\rangle$

 $\rightsquigarrow \quad |001\rangle + |010\rangle + |100\rangle$

W is a pseudo-spider because its ‘spider fusion’ rule is slightly different:

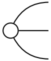


The fermionic swap is not ‘flexsymmetric’ – unlike for other generators, swapping its ‘legs’ changes the operation:

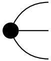
 $\neq$ 

Some properties of the ZW-calculus

The two three-legged spiders of the ZW-calculus represent the two inequivalent types of 3-qubit entangled states:

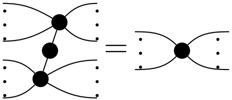


$\rightsquigarrow \quad |000\rangle + |111\rangle$



$\rightsquigarrow \quad |001\rangle + |010\rangle + |100\rangle$

W is a pseudo-spider because its 'spider fusion' rule is slightly different:



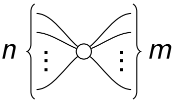
The fermionic swap is not 'flexsymmetric' – unlike for other generators, swapping its 'legs' changes the operation:

The ZW-calculus has a complete equational theory, in fact it was the first universal graphical calculus to be proved complete.

The ZH-calculus

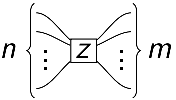
Z spider

[B. & Kissinger 2018]



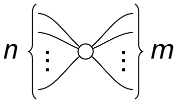
$$\rightsquigarrow \quad |\underbrace{0\ldots0}_{m}\rangle\langle\underbrace{0\ldots0}_{n}| + |\underbrace{1\ldots1}_{m}\rangle\langle\underbrace{1\ldots1}_{n}| = \begin{pmatrix}1 & \ldots & 0\\ \vdots & \ddots & \vdots\\ 0 & \ldots & 1\end{pmatrix}$$

H box, where $z \in \mathbb{C}$



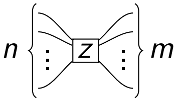
The ZH-calculus

Z spider [B. & Kissinger 2018]



$$\rightsquigarrow \quad |\underbrace{0\dots0}_{m}\rangle\langle\underbrace{0\dots0}_{n}| + |\underbrace{1\dots1}_{m}\rangle\langle\underbrace{1\dots1}_{n}| = \begin{pmatrix} 1 & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & 1 \end{pmatrix}$$

H box, where $z \in \mathbb{C}$



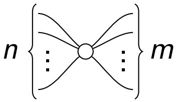
$$\rightsquigarrow \quad \sum_{\mathbf{x}\in\{0,1\}^m, \mathbf{y}\in\{0,1\}^n} z^{x_1\dots x_m y_1\dots y_n} |\mathbf{x}\rangle\langle\mathbf{y}| = \begin{pmatrix} 1 & \dots & 1 \\ \vdots & \ddots & \vdots \\ 1 & \dots & z \end{pmatrix}$$

An *H*-box with no label should be read as having parameter $z = -1$.

The ZH-calculus

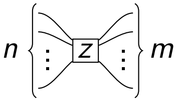
Z spider

[B. & Kissinger 2018]



$$\rightsquigarrow \quad |\underbrace{0\ldots0}_{m}\rangle\langle\underbrace{0\ldots0}_{n}| + |\underbrace{1\ldots1}_{m}\rangle\langle\underbrace{1\ldots1}_{n}| = \begin{pmatrix} 1 & \ldots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \ldots & 1 \end{pmatrix}$$

H box, where $z \in \mathbb{C}$



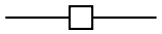
$$\rightsquigarrow \quad \sum_{\mathbf{x}\in\{0,1\}^m, \mathbf{y}\in\{0,1\}^n} z^{x_1\ldots x_m y_1 \ldots y_n} |\mathbf{x}\rangle\langle\mathbf{y}| = \begin{pmatrix} 1 & \ldots & 1 \\ \vdots & \ddots & \vdots \\ 1 & \ldots & z \end{pmatrix}$$

An H-box with no label should be read as having parameter $z = -1$.

The ZH-calculus has a complete equational theory.

Some properties of the ZH-calculus

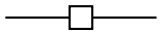
The binary H-box with label -1 is a Hadamard gate (up to scaling):



$$\rightsquigarrow \quad \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$$

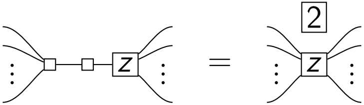
Some properties of the ZH-calculus

The binary H-box with label -1 is a Hadamard gate (up to scaling):



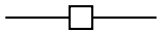
$$\rightsquigarrow \quad \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$$

The H-box is also a pseudo-spider:



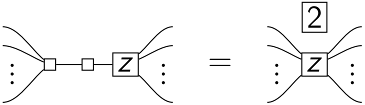
Some properties of the ZH-calculus

The binary H-box with label -1 is a Hadamard gate (up to scaling):

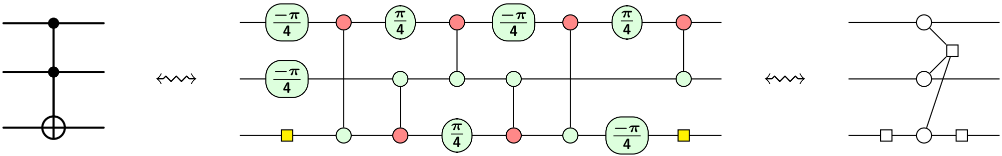


$$\rightsquigarrow \quad \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$$

The H-box is also a pseudo-spider:



The ZH-calculus is particularly useful for representing 'classical non-linearity', such as in multi-controlled gates, for example the Toffoli gate:



Probabilistic mixtures of quantum states

So far, we have only considered what are called pure states and unitary transformations, yet quantum physics offers options beyond that.

Probabilistic mixtures of quantum states

So far, we have only considered what are called pure states and unitary transformations, yet quantum physics offers options beyond that.

quantum superposition

$$|+\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$$

Probabilistic mixtures of quantum states

So far, we have only considered what are called pure states and unitary transformations, yet quantum physics offers options beyond that.

quantum superposition

$$|+\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$$

probabilistic mixture (e.g. after a measurement)

$|0\rangle$ with probability $\frac{1}{2}$ and $|1\rangle$ with probability $\frac{1}{2}$

Probabilistic mixtures of quantum states

So far, we have only considered what are called pure states and unitary transformations, yet quantum physics offers options beyond that.

quantum superposition

$$|+\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$$

probabilistic mixture (e.g. after a measurement)

$|0\rangle$ with probability $\frac{1}{2}$ and $|1\rangle$ with probability $\frac{1}{2}$

Probabilistic mixtures of quantum states

So far, we have only considered what are called pure states and unitary transformations, yet quantum physics offers options beyond that.

quantum superposition | probabilistic mixture (e.g. after a measurement)

$|+\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$ | $|0\rangle$ with probability $\frac{1}{2}$ and $|1\rangle$ with probability $\frac{1}{2}$

Mathematically, represent a mixed state – a probabilistic mixture of states $|\psi_k\rangle$ with probabilities p_k – as a density matrix:

$$\sum_k p_k |\psi_k\rangle\langle\psi_k|$$

Probabilistic mixtures of quantum states

So far, we have only considered what are called pure states and unitary transformations, yet quantum physics offers options beyond that.

quantum superposition

$$|+\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$$

probabilistic mixture (e.g. after a measurement)

$|0\rangle$ with probability $\frac{1}{2}$ and $|1\rangle$ with probability $\frac{1}{2}$

Mathematically, represent a mixed state – a probabilistic mixture of states $|\psi_k\rangle$ with probabilities p_k – as a density matrix:

$$\sum_k p_k |\psi_k\rangle\langle\psi_k|$$

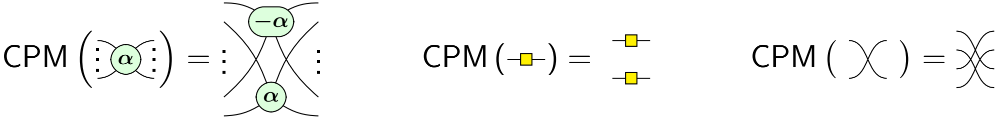
The above examples become:

$$|+\rangle\langle+| = \frac{1}{2}\begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix}$$

$$\frac{1}{2}|0\rangle\langle 0| + \frac{1}{2}|1\rangle\langle 1| = \frac{1}{2}\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

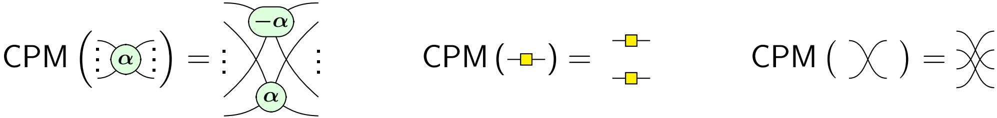
Graphical calculi for mixed states and classical information

Change the interpretation of the existing generators by combining each generator with its complex conjugate, e.g. for the ZX-calculus:

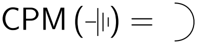


Graphical calculi for mixed states and classical information

Change the interpretation of the existing generators by combining each generator with its complex conjugate, e.g. for the ZX-calculus:

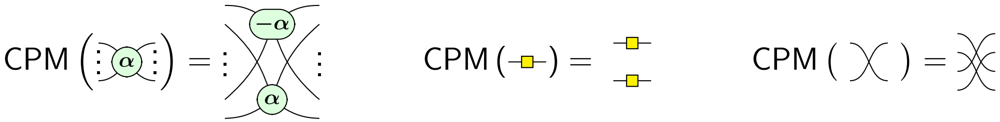


Add one new generator, which represents ‘measure and delete outcome’

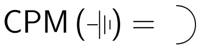


Graphical calculi for mixed states and classical information

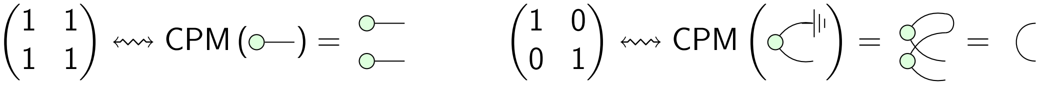
Change the interpretation of the existing generators by combining each generator with its complex conjugate, e.g. for the ZX-calculus:



Add one new generator, which represents ‘measure and delete outcome’

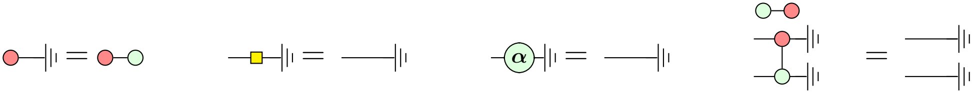


Then we can represent superpositions and probabilistic mixtures:



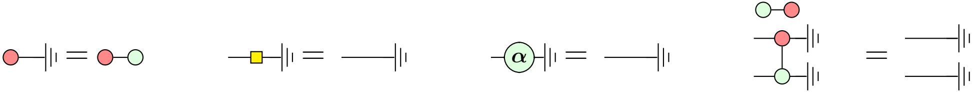
Equations for the new generator, and application

All existing ZX-calculus equations remain true and it suffices to add four new equations involving  [Carette et al. 2019]

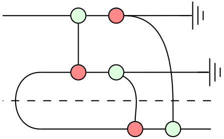


Equations for the new generator, and application

All existing ZX-calculus equations remain true and it suffices to add four new equations involving  [Carette et al. 2019]

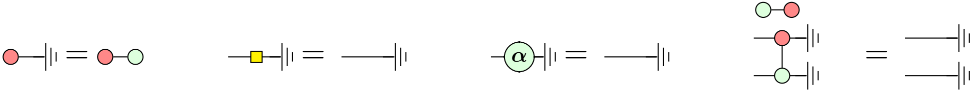


For example, this allows to represent the classical information flow in quantum teleportation:

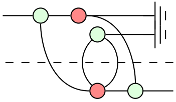


Equations for the new generator, and application

All existing ZX-calculus equations remain true and it suffices to add four new equations involving [Carette et al. 2019]

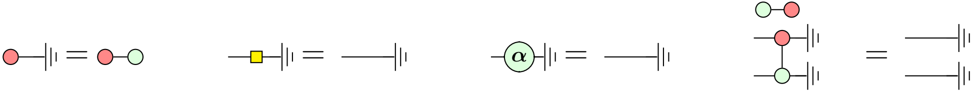


For example, this allows to represent the classical information flow in quantum teleportation:

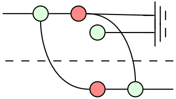


Equations for the new generator, and application

All existing ZX-calculus equations remain true and it suffices to add four new equations involving  [Carette et al. 2019]

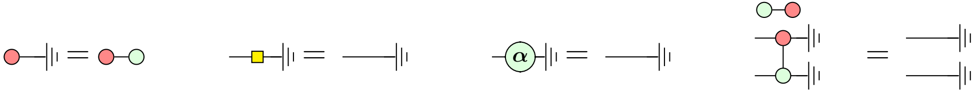


For example, this allows to represent the classical information flow in quantum teleportation:

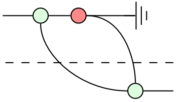


Equations for the new generator, and application

All existing ZX-calculus equations remain true and it suffices to add four new equations involving [Carette et al. 2019]

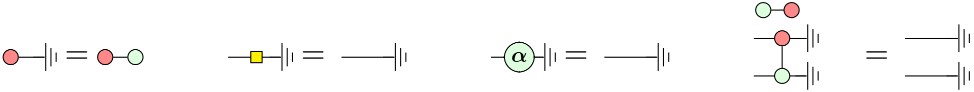


For example, this allows to represent the classical information flow in quantum teleportation:

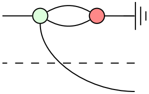


Equations for the new generator, and application

All existing ZX-calculus equations remain true and it suffices to add four new equations involving  [Carette et al. 2019]

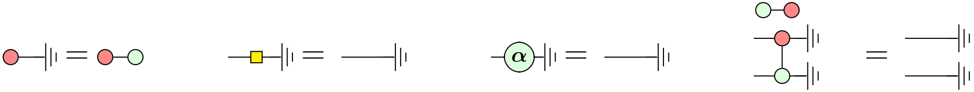


For example, this allows to represent the classical information flow in quantum teleportation:

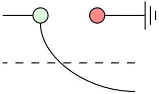


Equations for the new generator, and application

All existing ZX-calculus equations remain true and it suffices to add four new equations involving  [Carette et al. 2019]

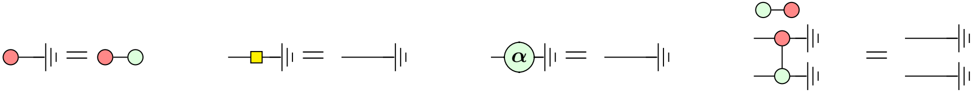


For example, this allows to represent the classical information flow in quantum teleportation:

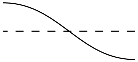


Equations for the new generator, and application

All existing ZX-calculus equations remain true and it suffices to add four new equations involving  [Carette et al. 2019]



For example, this allows to represent the classical information flow in quantum teleportation:



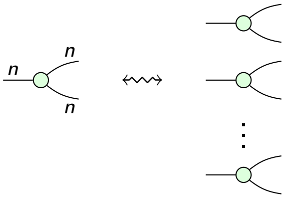
Scalable notation

Bundle multiple wires into one and introduce two new components to gather and split these bundles:

Scalable notation

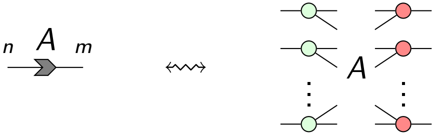
Bundle multiple wires into one and introduce two new components to gather and split these bundles:

In addition to combining multiple wires, also allow multiple copies of other generators, e.g.

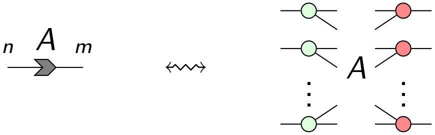


[Carette, Horsman, Perdrix 2019]

Unified notation and reasoning for diagrams with similar structure



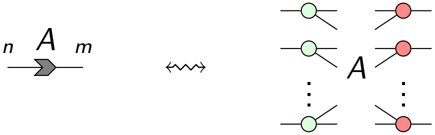
Unified notation and reasoning for diagrams with similar structure



Applying the matrix arrow labelled by a biadjacency matrix A to a computational basis state $|\mathbf{x}\rangle$ for some $\mathbf{x} \in \{0,1\}^n$:

$$|\mathbf{x}\rangle \xrightarrow{\quad A \quad} |A\mathbf{x}\rangle$$

Unified notation and reasoning for diagrams with similar structure



Applying the matrix arrow labelled by a biadjacency matrix A to a computational basis state $|\mathbf{x}\rangle$ for some $\mathbf{x} \in \{0,1\}^n$:

$$|\mathbf{x}\rangle \xrightarrow{\quad A \quad} |A\mathbf{x}\rangle$$

This has applications in quantum error correcting codes and many other areas of quantum computing.

Outline

Summary

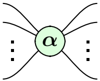
Graphical languages for representing and reasoning about quantum computations.

- Graphical rewriting: cutting and pasting of diagrams ✂ ❏

Summary

Graphical languages for representing and reasoning about quantum computations.

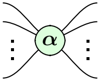
- Graphical rewriting: cutting and pasting of diagrams ✂ ❏
- ZX-calculus: spiders instead of gates



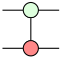
Summary

Graphical languages for representing and reasoning about quantum computations.

- Graphical rewriting: cutting and pasting of diagrams ✂ ❏
- ZX-calculus: spiders instead of gates



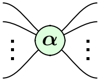
- More flexibility: only connectivity matters



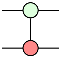
Summary

Graphical languages for representing and reasoning about quantum computations.

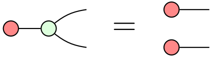
- Graphical rewriting: cutting and pasting of diagrams ✂ ❏
- ZX-calculus: spiders instead of gates



- More flexibility: only connectivity matters



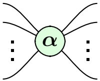
- A set of (fairly) simple rewrite rules



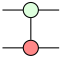
Summary

Graphical languages for representing and reasoning about quantum computations.

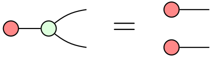
- Graphical rewriting: cutting and pasting of diagrams ✂ ❏
- ZX-calculus: spiders instead of gates



- More flexibility: only connectivity matters



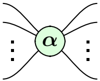
- A set of (fairly) simple rewrite rules



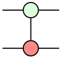
Summary

Graphical languages for representing and reasoning about quantum computations.

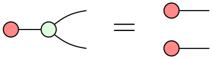
- Graphical rewriting: cutting and pasting of diagrams ✂ ❏
- ZX-calculus: spiders instead of gates



- More flexibility: only connectivity matters



- A set of (fairly) simple rewrite rules



More information about the ZX-calculus is at `https://zxcalculus.com/`. See also ⟨hal-05322779⟩ (in French).