Geometry of Interaction for ZX-Diagrams

Kostia Chardonnet, Benoît Valiron, Renaud Vilmart

Univ. Paris Saclay, LMF
Univ. Paris, IRIF

kostia@lri.fr

Journées GT Scalp 2021

- Classical bits as vectors: $|0\rangle := \begin{pmatrix} 1 \\ 0 \end{pmatrix}$ and $|1\rangle := \begin{pmatrix} 0 \\ 1 \end{pmatrix}$

- Classical bits as vectors: $|0\rangle := \begin{pmatrix}1\\0\end{pmatrix}$ and $|1\rangle := \begin{pmatrix}0\\1\end{pmatrix}$
- Arbitrary quantum bits (qubits): $\alpha\,|0\rangle + \beta\,|1\rangle$

- Classical bits as vectors: $|0\rangle := \begin{pmatrix} 1 \\ 0 \end{pmatrix}$ and $|1\rangle := \begin{pmatrix} 0 \\ 1 \end{pmatrix}$

- Arbitrary quantum bits (qubits): $\alpha\,|0\rangle + \beta\,|1\rangle$

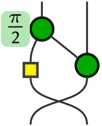
- Larger systems: $q_0 \otimes q_1$, $|01\rangle = |0\rangle \otimes |1\rangle = \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \end{pmatrix}$

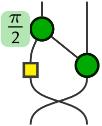
- Classical bits as vectors: $|0\rangle := \begin{pmatrix} 1 \\ 0 \end{pmatrix}$ and $|1\rangle := \begin{pmatrix} 0 \\ 1 \end{pmatrix}$
- Arbitrary quantum bits (qubits): $\alpha |0\rangle + \beta |1\rangle$
- Larger systems: $q_0 \otimes q_1$, $|01\rangle = |0\rangle \otimes |1\rangle = \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \end{pmatrix}$
- Operation are *linear maps*

- Was introduced by Coecke and Duncan in 2008

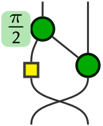
- Was introduced by Coecke and Duncan in 2008

- Is part of the Categorical Quantum Mechanics program (Abramsky&Coecke'04)

- Was introduced by Coecke and Duncan in 2008
- Is part of the Categorical Quantum Mechanics program (Abramsky&Coecke'04)
- Manipulates string diagrams e.g. 

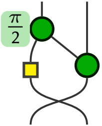
- Was introduced by Coecke and Duncan in 2008
- Is part of the Categorical Quantum Mechanics program (Abramsky&Coecke'04)
- Manipulates string diagrams e.g. 

- Relaxes unitarity

- Was introduced by Coecke and Duncan in 2008
- Is part of the Categorical Quantum Mechanics program (Abramsky&Coecke'04)
- Manipulates string diagrams e.g. 

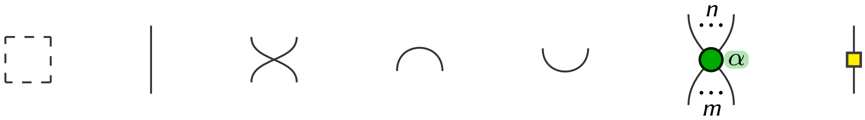
- Relaxes unitarity
- Is Universal (can encode any linear map)

- Was introduced by Coecke and Duncan in 2008
- Is part of the Categorical Quantum Mechanics program (Abramsky&Coecke'04)
- Manipulates string diagrams e.g.

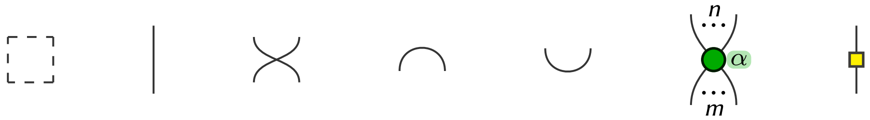


- Relaxes unitarity
- Is Universal (can encode any linear map)
- Lack a direct operational interpretation (this talk !)

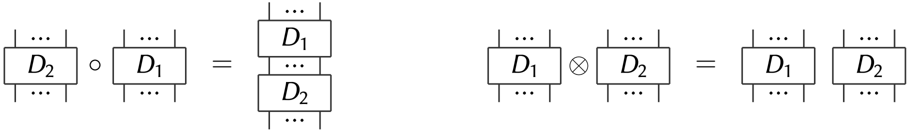
Generators



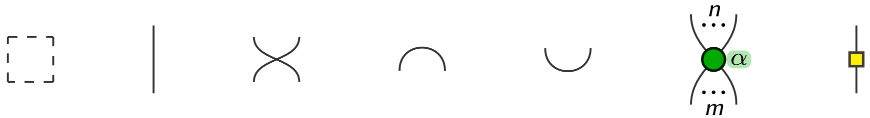
Generators



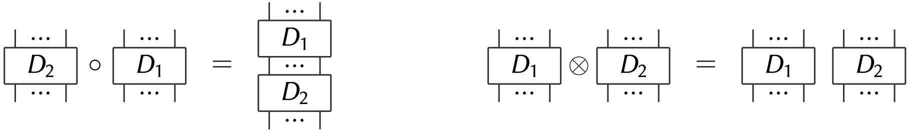
Compositions



Generators

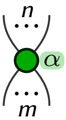


Compositions

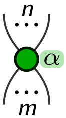


Standard Interpretation

$$[\![.]\!] : \mathbf{ZX} \to \mathcal{M}(\mathbb{C})$$

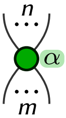
- A spider: 

$$:: \; 2^m \left\{ \overbrace{\begin{pmatrix} 1 & 0 & \cdots & \cdots & 0 \\ 0 & 0 & & & \vdots \\ \vdots & & \ddots & & \vdots \\ \vdots & & & 0 & 0 \\ 0 & \cdots & \cdots & 0 & e^{i\alpha} \end{pmatrix}}^{2^n} \right. = \begin{cases} |0\ldots0\rangle & \mapsto \; |0\ldots0\rangle \\ |1\ldots1\rangle & \mapsto \; e^{i\alpha}\,|1\ldots1\rangle \\ \_ & \mapsto \; 0 \end{cases}$$

- A spider: 

$$:: \quad 2^m \left\{ \overbrace{\begin{pmatrix} 1 & 0 & \cdots & \cdots & 0 \\ 0 & 0 & & & \vdots \\ \vdots & & \ddots & & \vdots \\ \vdots & & & 0 & 0 \\ 0 & \cdots & \cdots & 0 & e^{i\alpha} \end{pmatrix}}^{2^n} \right. = \begin{cases} |0\dots0\rangle \mapsto |0\dots0\rangle \\ |1\dots1\rangle \mapsto e^{i\alpha}|1\dots1\rangle \\ \_ \mapsto 0 \end{cases}$$

- A change of basis: 
$$:: \quad \frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} = \begin{cases} |0\rangle \mapsto \frac{|0\rangle+|1\rangle}{\sqrt{2}} \\ |1\rangle \mapsto \frac{|0\rangle-|1\rangle}{\sqrt{2}} \end{cases}$$

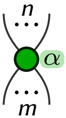
- A spider: 

$$:: \quad 2^m \left\{ \overbrace{\begin{pmatrix} 1 & 0 & \cdots & \cdots & 0 \\ 0 & 0 & & & \vdots \\ \vdots & & \ddots & & \vdots \\ \vdots & & & 0 & 0 \\ 0 & \cdots & \cdots & 0 & e^{i\alpha} \end{pmatrix}}^{2^n} \right. = \begin{cases} |0\ldots0\rangle & \mapsto \quad |0\ldots0\rangle \\ |1\ldots1\rangle & \mapsto \quad e^{i\alpha}|1\ldots1\rangle \\ \_ & \mapsto 0 \end{cases}$$

- A change of basis: 
$$:: \quad \frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} = \begin{cases} |0\rangle \mapsto \frac{|0\rangle+|1\rangle}{\sqrt{2}} \\ |1\rangle \mapsto \frac{|0\rangle-|1\rangle}{\sqrt{2}} \end{cases}$$

- Wires:
$$:: \quad \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} = |x\rangle \mapsto |x\rangle \,,$$

$$:: \quad \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} = |xy\rangle \mapsto |yx\rangle$$

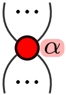
- A spider:  $:: \; 2^m \left\{ \overbrace{\begin{pmatrix} 1 & 0 & \cdots & \cdots & 0 \\ 0 & 0 & & & \vdots \\ \vdots & & \ddots & & \vdots \\ \vdots & & & 0 & 0 \\ 0 & \cdots & \cdots & 0 & e^{i\alpha} \end{pmatrix}}^{2^n} \right. = \begin{cases} |0\dots0\rangle & \mapsto \; |0\dots0\rangle \\ |1\dots1\rangle & \mapsto \; e^{i\alpha}\,|1\dots1\rangle \\ \_ & \mapsto 0 \end{cases}$

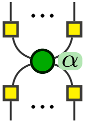
- A change of basis:  $:: \; \frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} = \begin{cases} |0\rangle \mapsto \frac{|0\rangle+|1\rangle}{\sqrt{2}} \\ |1\rangle \mapsto \frac{|0\rangle-|1\rangle}{\sqrt{2}} \end{cases}$

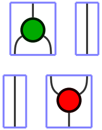
- Wires: $:: \; \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} = |x\rangle \mapsto |x\rangle$,  $:: \; \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} = |xy\rangle \mapsto |yx\rangle$

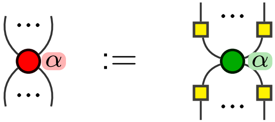
-  $:: \; \begin{pmatrix} 1 \\ 0 \\ 0 \\ 1 \end{pmatrix} = \alpha \mapsto \alpha\,|00\rangle + \alpha\,|11\rangle : \mathbb{C} \to \mathbb{C}^2$

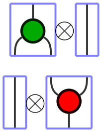
-  $:: \; \begin{pmatrix} 1 & 0 & 0 & 1 \end{pmatrix} = |xy\rangle \mapsto \delta_{x=y} : \mathbb{C}^2 \to \mathbb{C}$

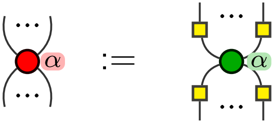
We define a new spider : 

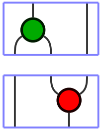
:= 

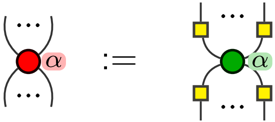
CNOT = 

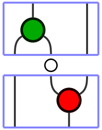
We define a new spider : 

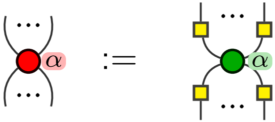
CNOT = 

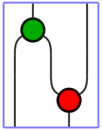
We define a new spider : 

CNOT = 

We define a new spider : 

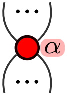
CNOT = 

We define a new spider : 

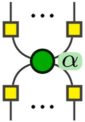
CNOT = 

Example

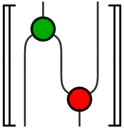
We define a new spider :



:=

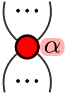


CNOT =

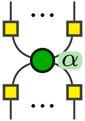


=

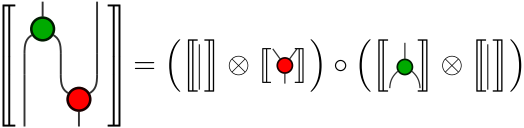
We define a new spider :

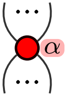


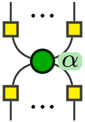
:=

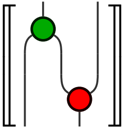


CNOT =



We define a new spider : 

$:=$ 

CNOT $=$  $= \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}$

- GoI as a **Token Machine**
- Proof Nets / ZX seen as graphs
- Tokens travel through the graph
- Proof Nets ⇒ Capture computational content of the proof
- Our Token Machine : superposition of tokens, multiple tokens, collisions, ...

⇒ Bring Operational Semantic on ZX-Diagrams.

Already existing work: “The geometry of parallelism. classical, probabilistic, and quantum effects” [Dal Lago, Faggian, Valiron, Yoshimizu]

	Dal Lago et al.	This Work
Superposition	✗	✓
Asynchronicity	✗	✓
Types	✓	✗

Token

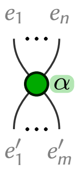
3-tuple $(e, d, b) \in \mathcal{E}(D) \times \{\uparrow, \downarrow\} \times \{0, 1\}$ where:

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

Token

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

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

e.g :  with $(e_1 \downarrow 0)(e'_3 \uparrow 1) + (e_1 \downarrow 1)$

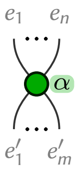
Token State

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

Token

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

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

e.g : 

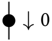
with $(e_1 \downarrow 0)(e'_3 \uparrow 1) + (e_1 \downarrow 1)$

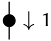
Token State

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

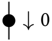
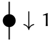
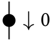
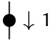
- The tokens modify the global token state as they move.

- Tokens represent information  $\downarrow 0$ and  $\downarrow 1$

- Tokens represent information 

and 

- Multiple tokens on a diagram

- Tokens represent information  and 
- Multiple tokens on a diagram
- Weighted sum of these diagrams : superposition (e.g. α  $+ \beta$ )

- Tokens represent information $\bullet{\downarrow}0$ and $\bullet{\downarrow}1$
- Multiple tokens on a diagram
- Weighted sum of these diagrams : superposition (e.g. $\alpha\,\bullet{\downarrow}0 + \beta\,\bullet{\downarrow}1$)
- Collisions : $\begin{smallmatrix}\bullet{\downarrow}x \\ \bullet{\uparrow}x\end{smallmatrix} \rightsquigarrow \;|$

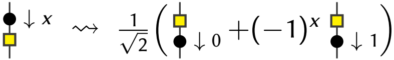
- Tokens represent information $\downarrow 0$ and $\downarrow 1$
- Multiple tokens on a diagram
- Weighted sum of these diagrams : superposition (e.g. $\alpha \downarrow 0 + \beta \downarrow 1$)
- Collisions :

- Tokens represent information $\bullet\!\downarrow 0$ and $\bullet\!\downarrow 1$
- Multiple tokens on a diagram
- Weighted sum of these diagrams : superposition (e.g. $\alpha \bullet\!\downarrow 0 + \beta \bullet\!\downarrow 1$)
- Collisions : 

- Diffusions : 

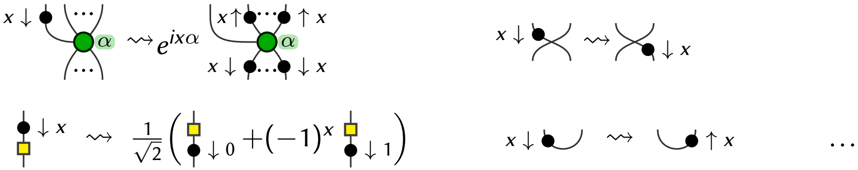
- Tokens represent information $\downarrow 0$ and $\downarrow 1$
- Multiple tokens on a diagram
- Weighted sum of these diagrams : superposition (e.g. $\alpha \downarrow 0 + \beta \downarrow 1$)
- Collisions : 

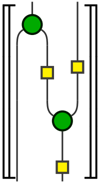
- Diffusions : 



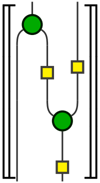
- Tokens represent information and
- Multiple tokens on a diagram
- Weighted sum of these diagrams : superposition (e.g.
- Collisions :

- Diffusions :

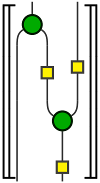




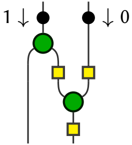
$$= \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}$$

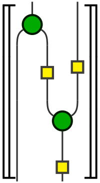


$$= \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} \implies |10\rangle \mapsto \frac{1}{\sqrt{2}} |11\rangle$$

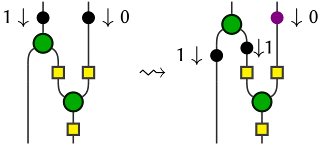


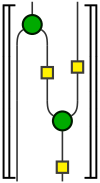
$$= \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} \implies |10\rangle \mapsto \frac{1}{\sqrt{2}} |11\rangle$$



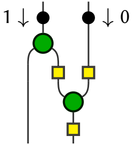


$$= \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} \implies |10\rangle \mapsto \frac{1}{\sqrt{2}} |11\rangle$$

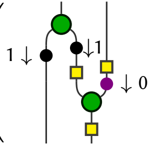




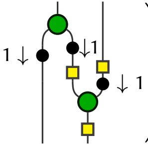
$$= \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} \implies |10\rangle \mapsto \frac{1}{\sqrt{2}} |11\rangle$$



$$\rightsquigarrow^* \frac{1}{\sqrt{2}} \Bigg($$

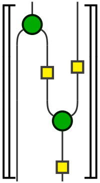


$$+$$

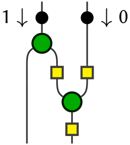


$$\Bigg)$$

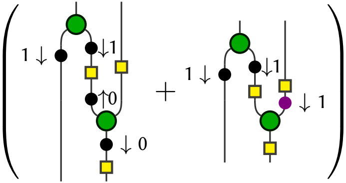
Example

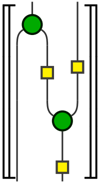


$$= \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} \implies |10\rangle \mapsto \frac{1}{\sqrt{2}} |11\rangle$$

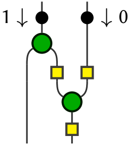


$$\rightsquigarrow^* \frac{1}{\sqrt{2}}$$

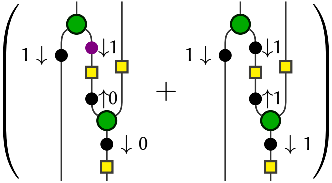


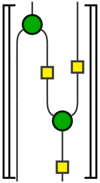


$$= \frac{1}{\sqrt{2}}\begin{pmatrix}1 & 0 & 0 & 0\\ 0 & 1 & 0 & 0\\ 0 & 0 & 0 & 1\\ 0 & 0 & 1 & 0\end{pmatrix} \implies |10\rangle \mapsto \frac{1}{\sqrt{2}}|11\rangle$$

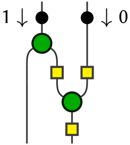


$$\rightsquigarrow^* \frac{1}{\sqrt{2}}$$

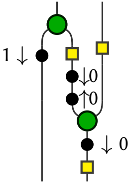




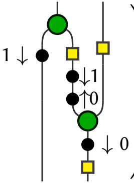
$$= \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} \implies |10\rangle \mapsto \frac{1}{\sqrt{2}} |11\rangle$$



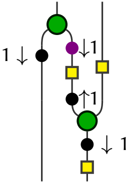
$$\rightsquigarrow^* \frac{1}{\sqrt{2}} \left(\frac{1}{\sqrt{2}} \left(\right. \right.$$



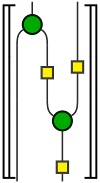
$$-$$



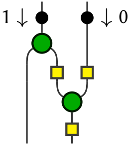
$$\left. \right) +$$



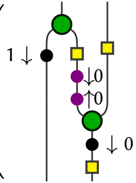
$$\left. \right)$$



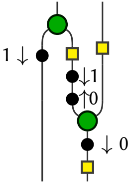
$$= \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} \implies |10\rangle \mapsto \frac{1}{\sqrt{2}} |11\rangle$$



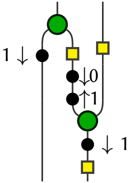
$$\rightsquigarrow^* \frac{1}{2} \Bigg($$



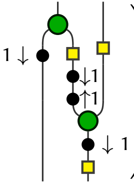
$-$



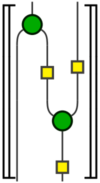
$+$



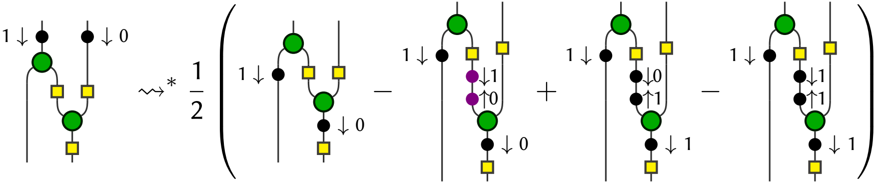
$-$

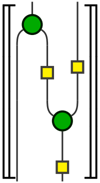


$$\Bigg)$$

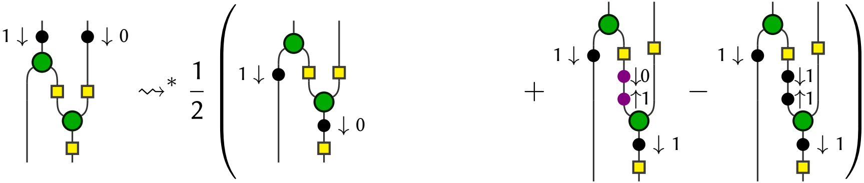


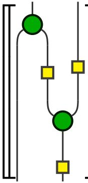
$$= \frac{1}{\sqrt{2}}\begin{pmatrix}1 & 0 & 0 & 0\\ 0 & 1 & 0 & 0\\ 0 & 0 & 0 & 1\\ 0 & 0 & 1 & 0\end{pmatrix} \implies |10\rangle \mapsto \frac{1}{\sqrt{2}}|11\rangle$$

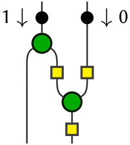




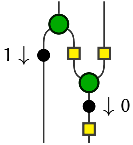
$$= \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} \implies |10\rangle \mapsto \frac{1}{\sqrt{2}} |11\rangle$$



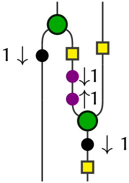
$[\![$  $]\!] = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} \implies |10\rangle \mapsto \frac{1}{\sqrt{2}} |11\rangle$



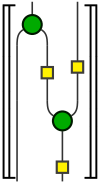
$\rightsquigarrow^* \frac{1}{2} \Bigg($



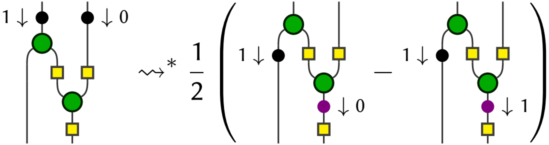
$-$

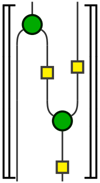


$\Bigg)$

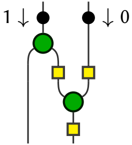


$$= \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} \implies |10\rangle \mapsto \frac{1}{\sqrt{2}} |11\rangle$$

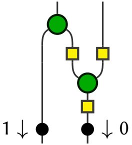




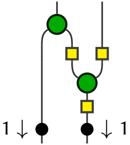
$$= \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} \implies |10\rangle \mapsto \frac{1}{\sqrt{2}} |11\rangle$$



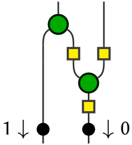
$$\rightsquigarrow^* \frac{1}{2\sqrt{2}} \Bigg($$



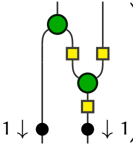
$$+$$



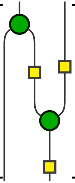
$$-$$

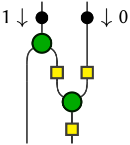


$$+$$

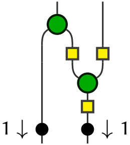


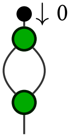
$$\Bigg)$$

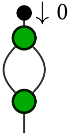
$[\![$  $]\!]$ $= \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} \implies |10\rangle \mapsto \frac{1}{\sqrt{2}} |11\rangle$



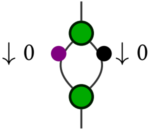
$\rightsquigarrow^* \dfrac{1}{\sqrt{2}}$



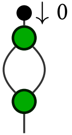
Take 

Take 

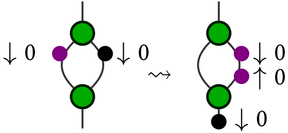
With good rewriting order:

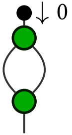


Example

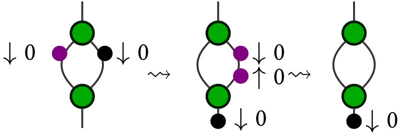
Take 

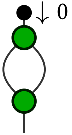
With good rewriting order:



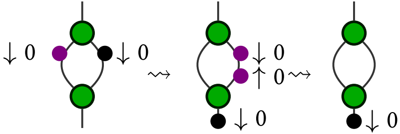
Take 

With good rewriting order:

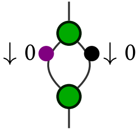


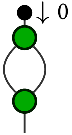
Take 

With good rewriting order:

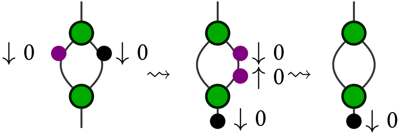


With arbitrary rewriting order:

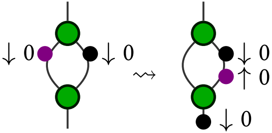


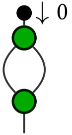
Take 

With good rewriting order:

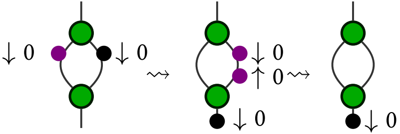


With arbitrary rewriting order:

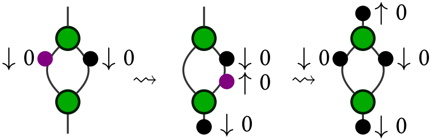


Take 

With good rewriting order:



With arbitrary rewriting order:



Rewriting System

We define a transition system $\rightsquigarrow$ as *exactly* one **diffusion rule** rule followed by all possible **collision** rules until none apply

Want to avoid:

- Having multiple tokens on the same edge that don't collide (i.e $(a \downarrow x)(a \downarrow x)$)
- Non-termination

Want to avoid:

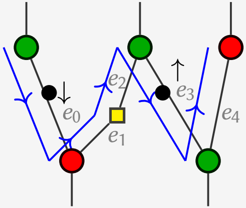
- Having multiple tokens on the same edge that don't collide (i.e $(a \downarrow x)(a \downarrow x)$)
- Non-termination

Two invariants:

- **Well-Formedness** : Avoid bad configuration
- **Cycle-Balancedness** : Termination, Confluence

Polarity in a Path

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



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

Example:

- Here, polarity 0

Well-Formed Token State

We say that a ZX-Diagram D is **Well-Formed** if for every path p its Polarity $\in \{-1, 0, 1\}$

Well-Formed Token State

We say that a ZX-Diagram D is **Well-Formed** if for every path p its Polarity $\in \{-1, 0, 1\}$

Theorem (Invariance of Well-Formedness)

Well-Formedness is preserved under $\rightsquigarrow$.

Well-Formedness of Token State

Well-Formed Token State

We say that a ZX-Diagram D is **Well-Formed** if for every path p its Polarity $\in \{-1, 0, 1\}$

Theorem (Invariance of Well-Formedness)

Well-Formedness is preserved under $\rightsquigarrow$.

Theorem (Characterisation of Well-Formedness)

Well-formed states cannot reach “bad configurations”.

Cycle-Balanced Token State

We say that a ZX-Diagram D is **Cycle-Balanced** if for every cycle c its Polarity $= 0$

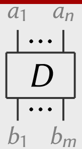
Cycle-Balanced Token State

We say that a ZX-Diagram D is **Cycle-Balanced** if for every cycle c its Polarity $= 0$

Theorem (Termination of well-formed, cycle-balanced token state)

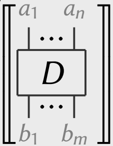
Let D be a ZX-Diagram, and s a well-formed, cycle-balanced token state, then s terminates.

Theorem (Simulation of Standard Interpretation)

Let 

a ZX-Diagram and let $t = (e \downarrow 0)(e \uparrow 0) + (e \downarrow 1)(e \uparrow 1)$.

t is a well-formed, cycle balanced token state.

We have that $t \rightsquigarrow^*$ 

$\Rightarrow$ we recover the **standard interpretation**.

Conclusion

- The Token Machine gives us a very general framework to study ZX-Calculus.
- More operational approach to ZX-Calculus.
- Already extended to : SOP, Mixed-Processes.

Conclusion

- The Token Machine gives us a very general framework to study ZX-Calculus.
- More operational approach to ZX-Calculus.
- Already extended to : SOP, Mixed-Processes.

Future Work

- Stronger relation with Linear Logic and Proof Nets (WIP)
- Hope it can help in defining new extensions of ZX-Calculus such as recursion (Future Work).