

# Langages graphiques pour programmer et raisonner en informatique quantique

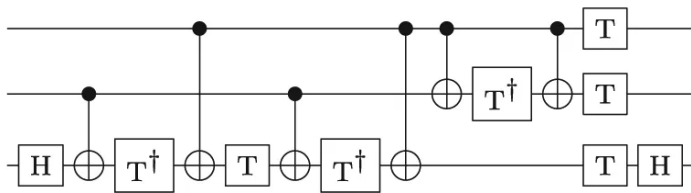
Simon Perdrix  
CNRS, LORIA, Nancy  
simon.perdrix@loria.fr

Séminaire Collège de France — 5 mai 2021  
dans le cadre du cours de Frédéric Magniez



# Langages graphiques en informatique quantique

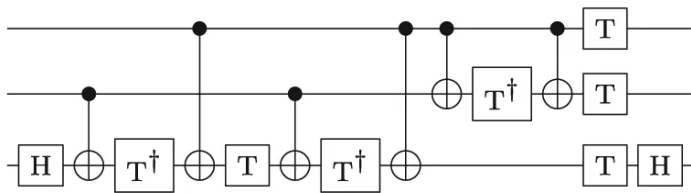
## Evolutions quantiques



Circuits quantiques

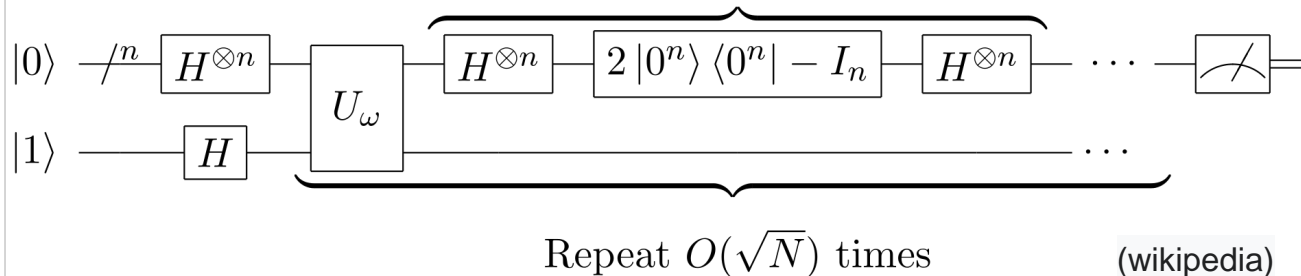
# Langages graphiques en informatique quantique

## Evolutions quantiques



Circuits quantiques

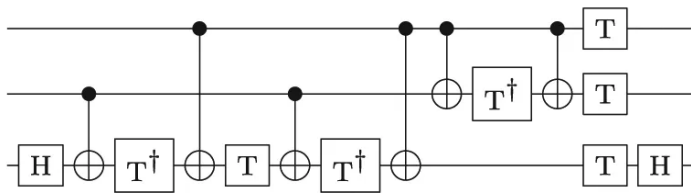
## Grover diffusion operator



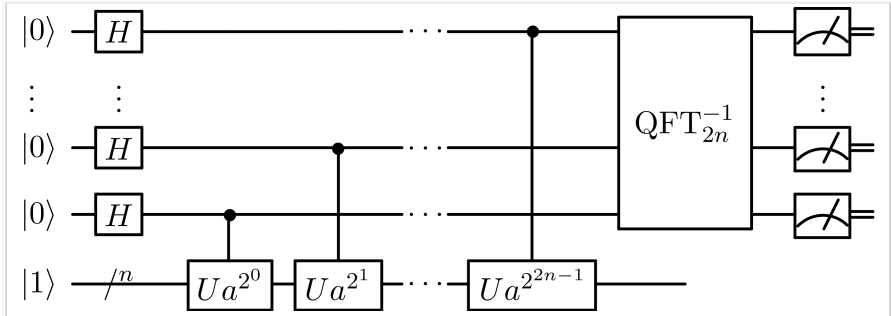
(wikipedia)

# Langages graphiques en informatique quantique

## Evolutions quantiques

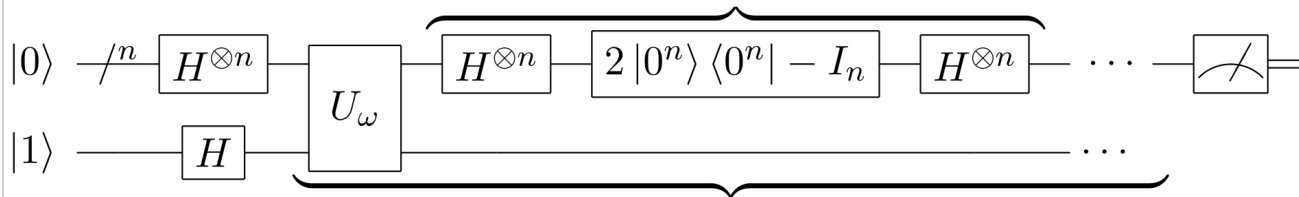


Circuits quantiques



Quantum subroutine in Shor's algorithm (wikipedia)

## Grover diffusion operator

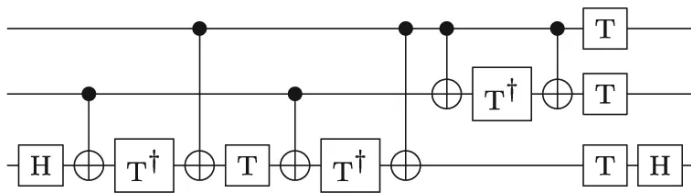


Repeat  $O(\sqrt{N})$  times

(wikipedia)

# Langages graphiques en informatique quantique

## Evolutions quantiques



Circuits quantiques

# Langages graphiques en informatique quantique

## Plan

- Contexte / historique
- Pourquoi un langage graphique ?
- ZX-calcul
- Applications
- Extensions

# Premiers modèles de calcul quantique

Machine de Turing Quantique [Deutsch 1985] :

$$(a,0) \xrightarrow{\frac{1}{\sqrt{2}}} (b,1, \rightarrow )$$

$$(b,1) \xrightarrow{\frac{-1}{\sqrt{2}}} (d,1, \rightarrow )$$

$$(d,0) \xrightarrow{\frac{-i}{\sqrt{3}}} (f,1, \leftarrow )$$

⋮

- Modèle de référence [Bernstein, Vazirani '92]
- Difficile à programmer
- Peu réaliste

# Premiers modèles de calcul quantique

Machine de Turing Quantique [Deutsch 1985] :

$$(a,0) \xrightarrow{\frac{1}{\sqrt{2}}} (b,1, \rightarrow )$$

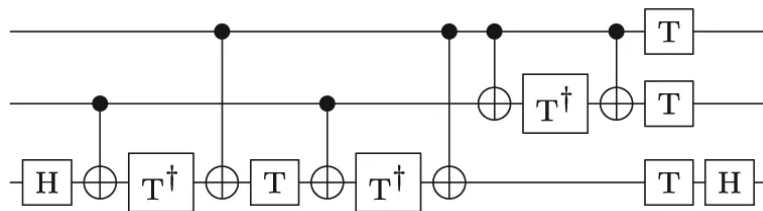
$$(b,1) \xrightarrow{\frac{-1}{\sqrt{2}}} (d,1, \rightarrow )$$

$$(d,0) \xrightarrow{\frac{-i}{\sqrt{3}}} (f,1, \leftarrow )$$

⋮

- Modèle de référence [Bernstein, Vazirani '92]
- Difficile à programmer
- Peu réaliste

Circuits Quantiques [Deutsch 1989] :



- Equivalents aux MT quantiques [Yao '93]
- Valide par construction
- Modèle réaliste d'implémentation



# Langages modernes de programmation quantique

Quipper, LNL-FPC, Qiskit, Cirq...

Quipper :

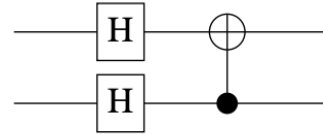
```
mycirc :: Qubit -> Qubit -> Circ (Qubit, Qubit)
mycirc a b = do
  a <- hadamard a
  b <- hadamard b
  (a,b) <- controlled_not a b
  return (a,b)
```

# Langages modernes de programmation quantique

Quipper, LNL-FPC, Qiskit, Cirq...

Quipper :

```
mycirc :: Qubit -> Qubit -> Circ (Qubit, Qubit)
mycirc a b = do
  a <- hadamard a
  b <- hadamard b
  (a,b) <- controlled_not a b
  return (a,b)
```

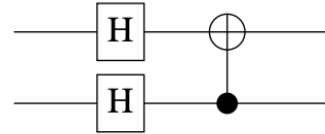


# Langages modernes de programmation quantique

Quipper, LNL-FPC, Qiskit, Cirq... -> Langages de description de circuits/diagrammes.

Quipper :

```
mycirc :: Qubit -> Qubit -> Circ (Qubit, Qubit)
mycirc a b = do
  a <- hadamard a
  b <- hadamard b
  (a,b) <- controlled_not a b
  return (a,b)
```

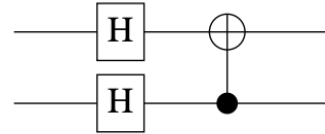


# Langages modernes de programmation quantique

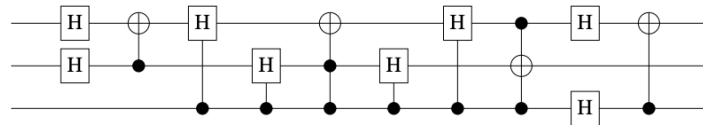
Quipper, LNL-FPC, Qiskit, Cirq... -> Langages de description de circuits/diagrammes.

Quipper :

```
mycirc :: Qubit -> Qubit -> Circ (Qubit, Qubit)
mycirc a b = do
  a <- hadamard a
  b <- hadamard b
  (a,b) <- controlled_not a b
  return (a,b)
```



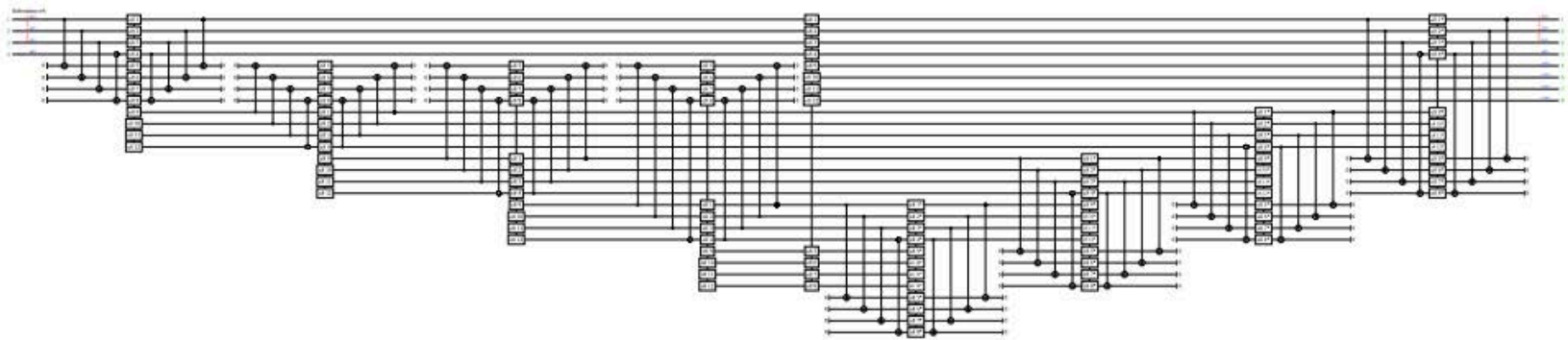
```
mycirc2 :: Qubit -> Qubit -> Qubit
-> Circ (Qubit, Qubit, Qubit)
mycirc2 a b c = do
  mycirc a b
  with_controls c $ do
    mycirc a b
    mycirc b a
  mycirc a c
  return (a,b,c)
```



# Langages modernes de programmation quantique

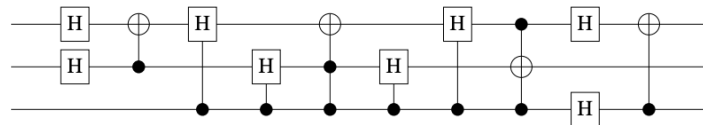
Quipper, LNL-FPC, Qiskit, Cirq... -> Langages de description de circuits/diagrammes.

Quipper :



**Figure 2.** The circuit for o4\_POW17

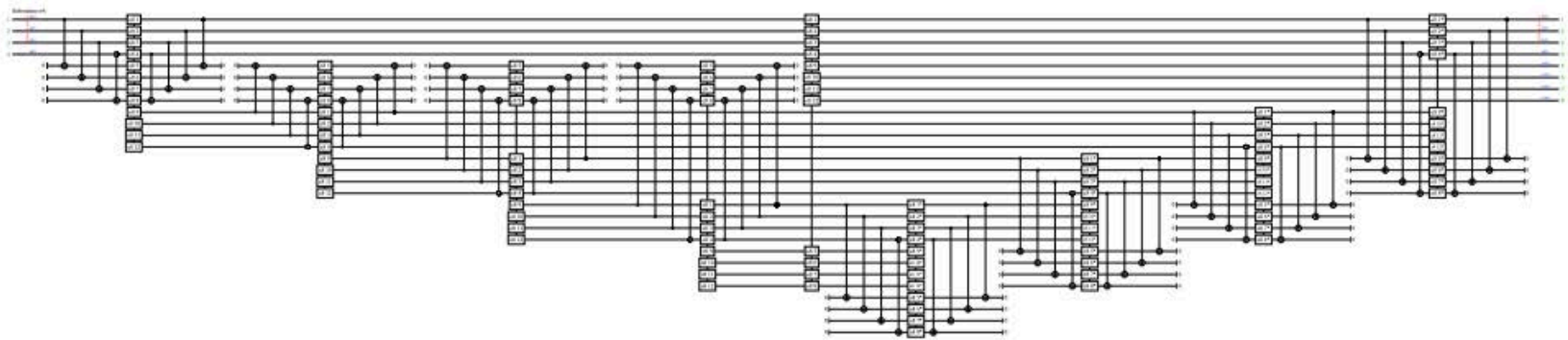
```
mycirc a b
with_controls c $ do
  mycirc a b
  mycirc b a
mycirc a c
return (a,b,c)
```



# Langages modernes de programmation quantique

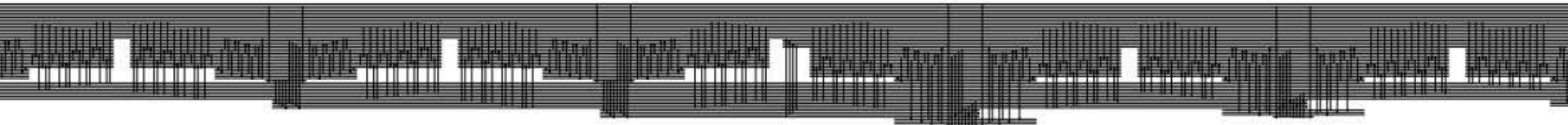
Quipper, LNL-FPC, Qiskit, Cirq... -> Langages de description de circuits/diagrammes.

Quipper :



**Figure 2.** The circuit for o4\_POW17

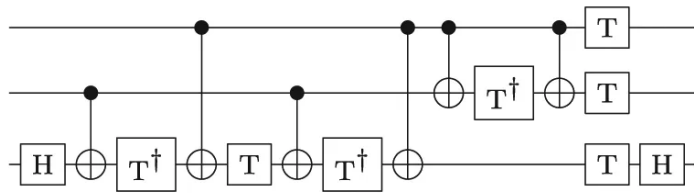
шусirc a b



**Figure 3.** The circuit for o8\_MUL

# Langages graphiques en informatique quantique

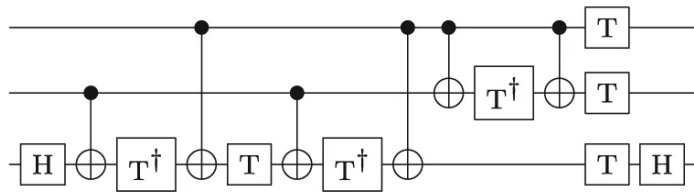
## Evolutions quantiques



Circuits quantiques

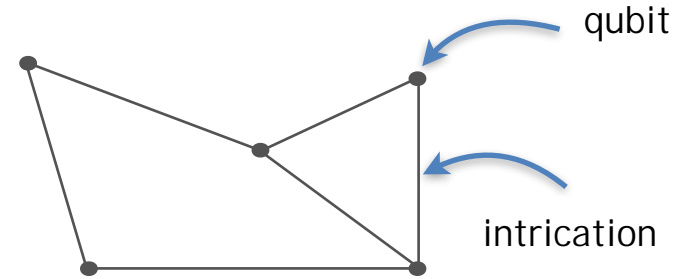
# Langages graphiques en informatique quantique

## Evolutions quantiques



Circuits quantiques

## Etats quantiques

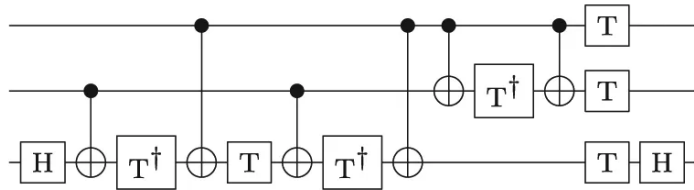


Graph States



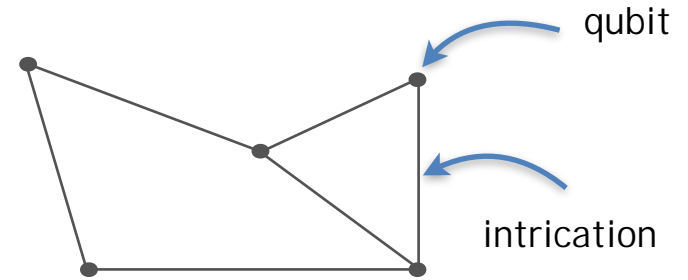
# Langages graphiques en informatique quantique

## Evolutions quantiques

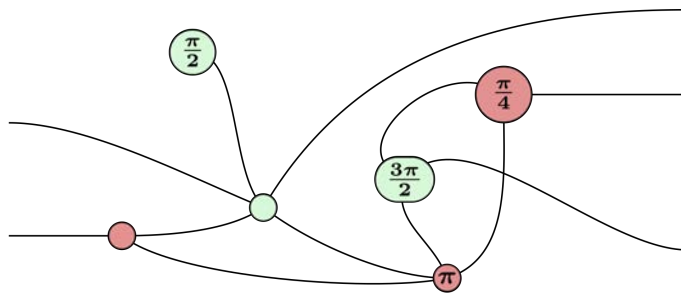


Circuits quantiques

## Etats quantiques



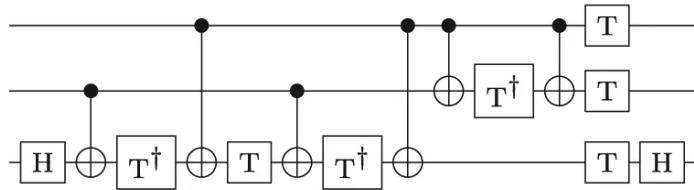
Graph States



ZX-calcul

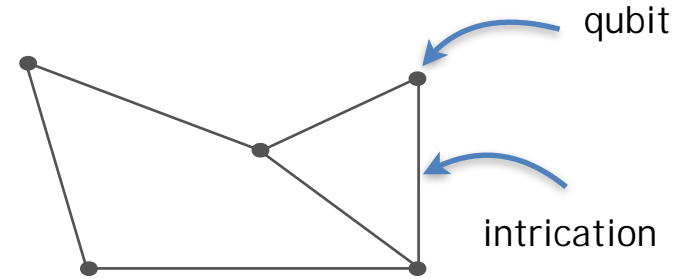
# Langages graphiques en informatique quantique

## Evolutions quantiques

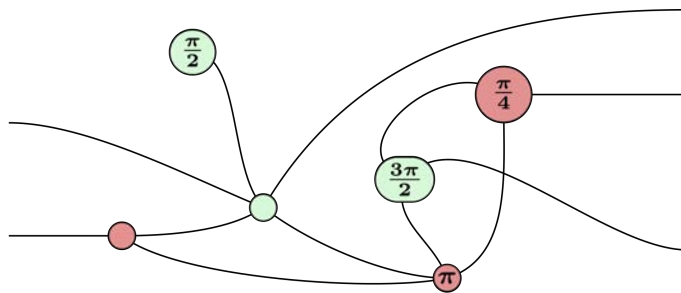


Circuits quantiques

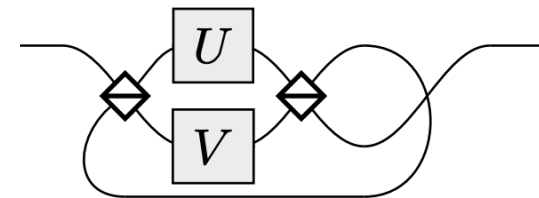
## Etats quantiques



Graph States



ZX-calcul



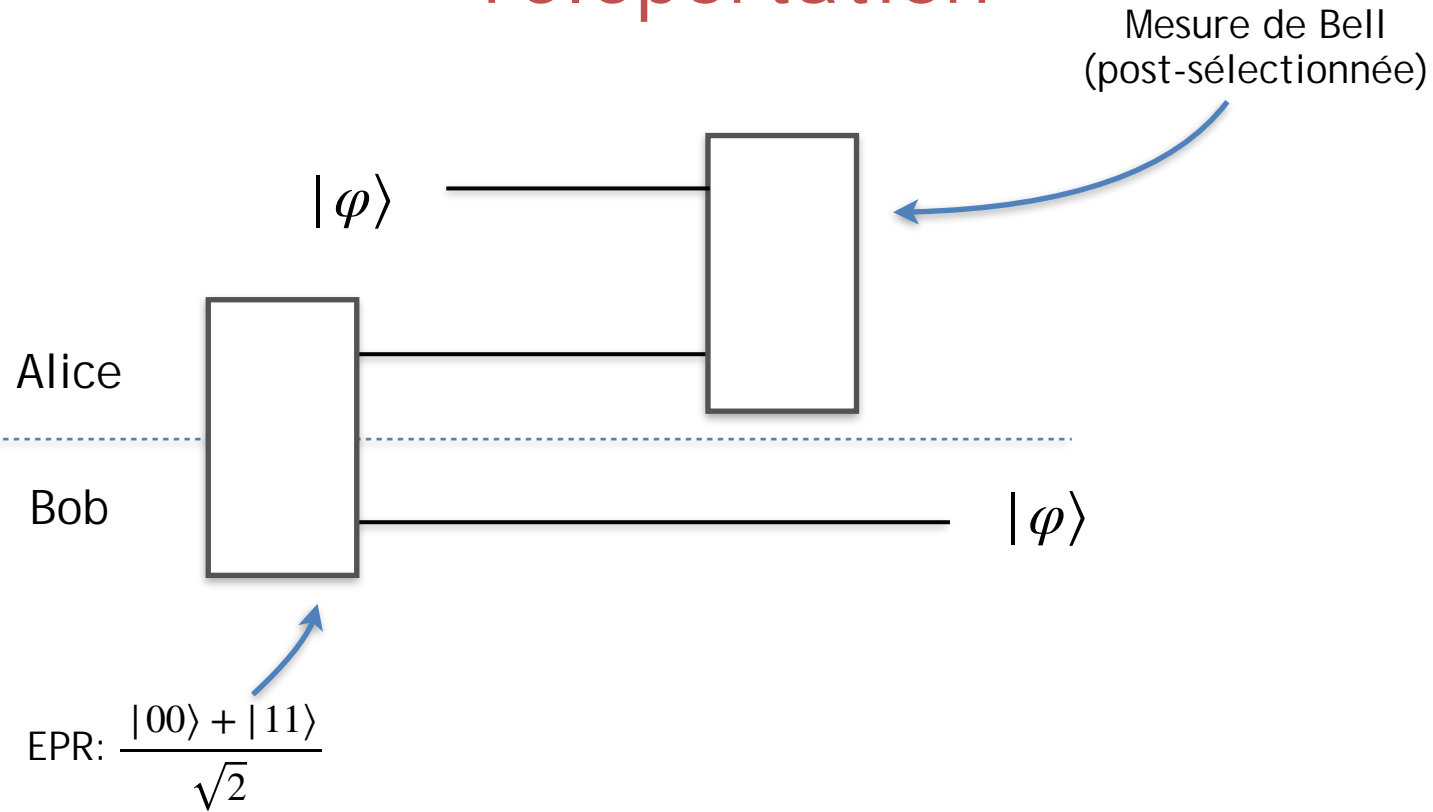
PBS-calcul

# Langages graphiques en informatique quantique

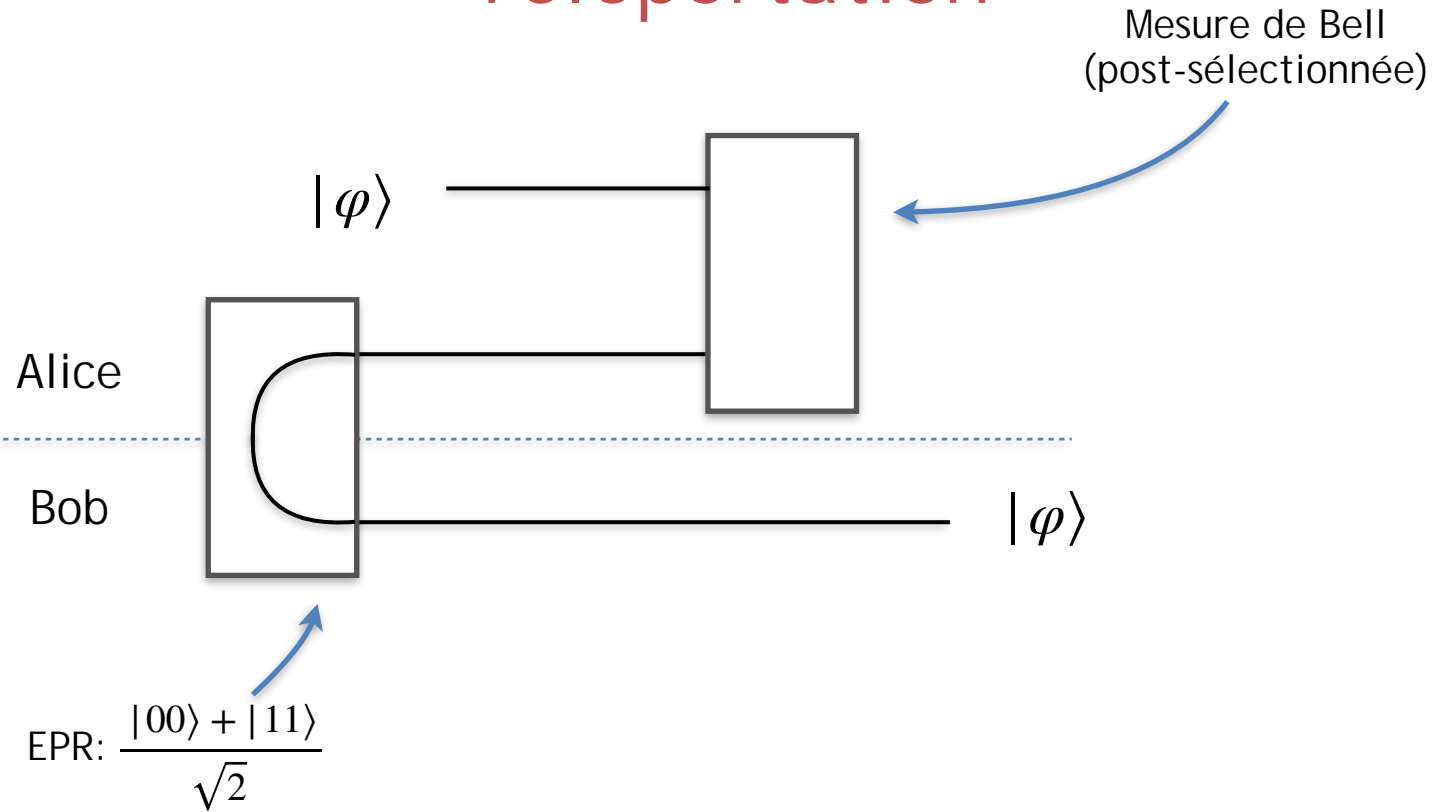
## Plan

- Contexte / historique
- Pourquoi un langage graphique ?
  - Téléportation
  - Non-cloning
  - Principe d'incertitude
- ZX-calcul
- Applications
- Extensions

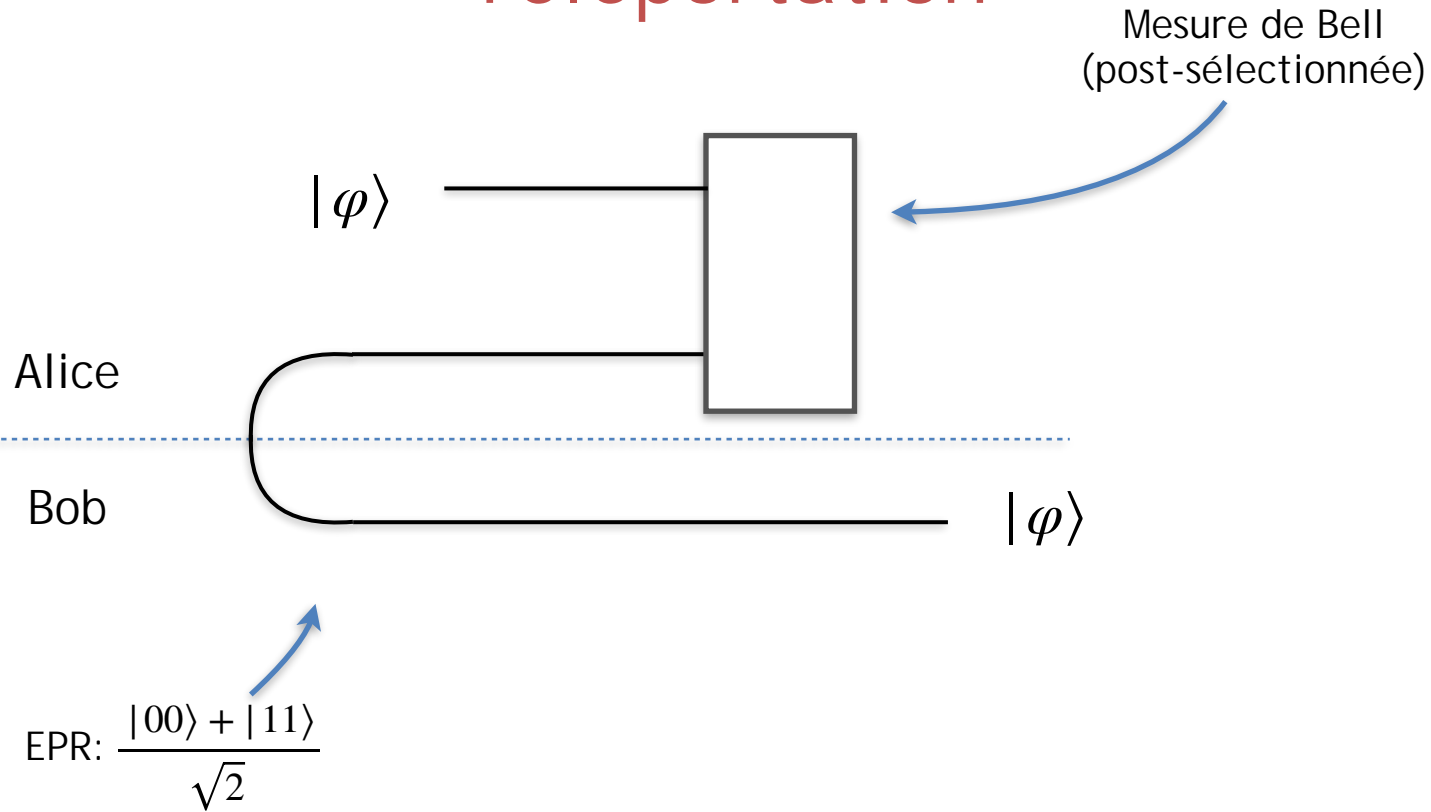
# Téléportation



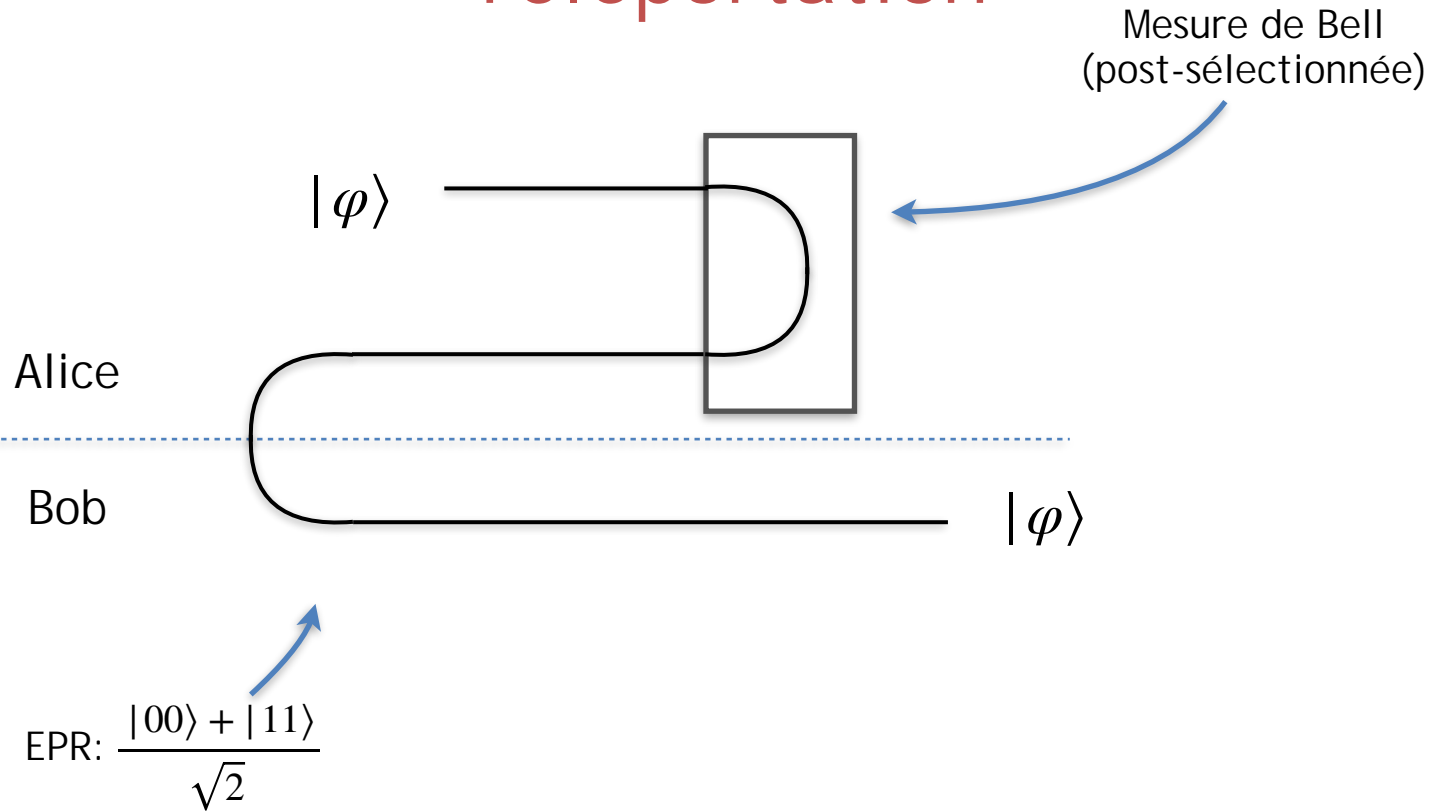
# Téléportation



# Téléportation



# Téléportation



# Téléportation

Mesure de Bell  
(post-sélectionnée)

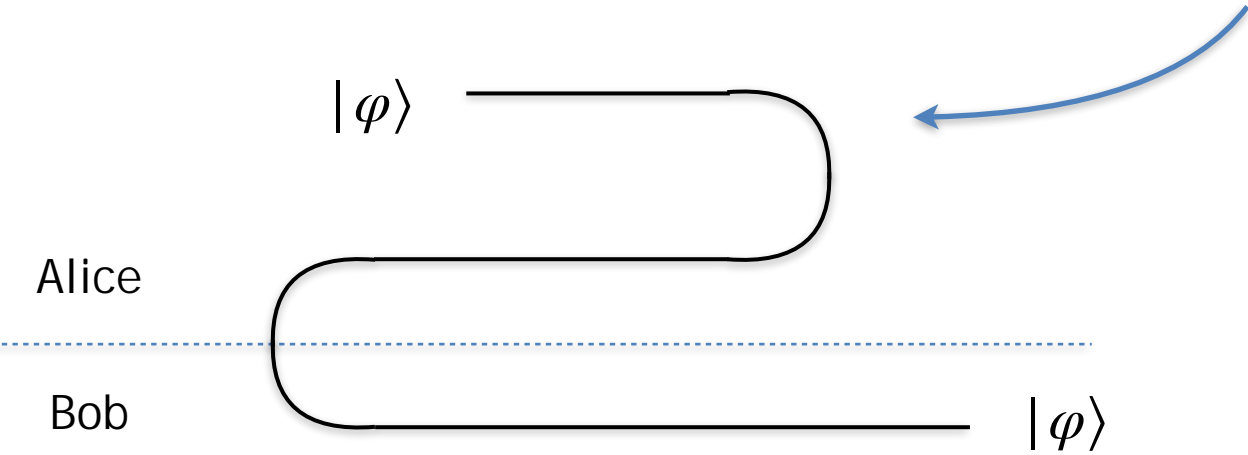
$|\varphi\rangle$

Alice

Bob

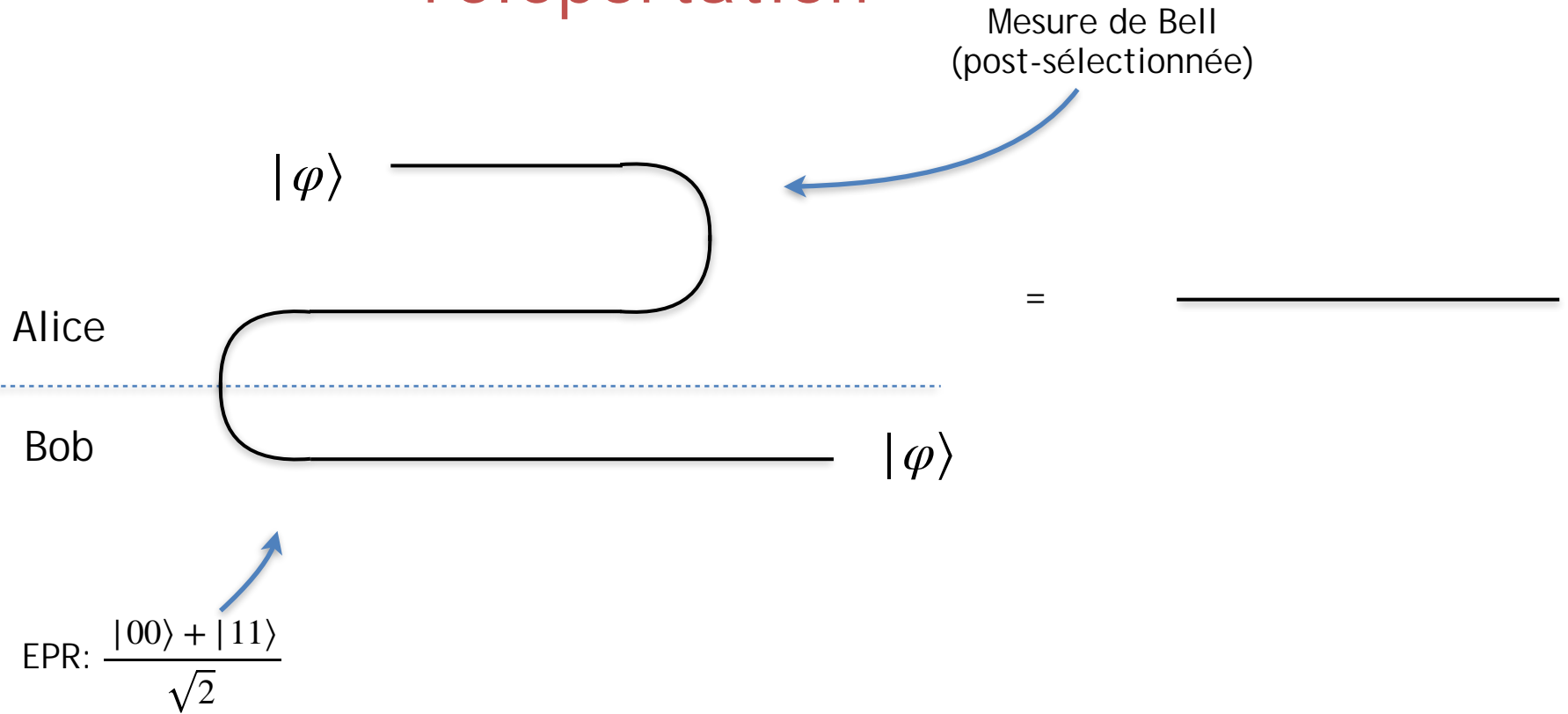
$|\varphi\rangle$

EPR:  $\frac{|00\rangle + |11\rangle}{\sqrt{2}}$





# Téléportation



# Téléportation

Mesure de Bell  
(post-sélectionnée)

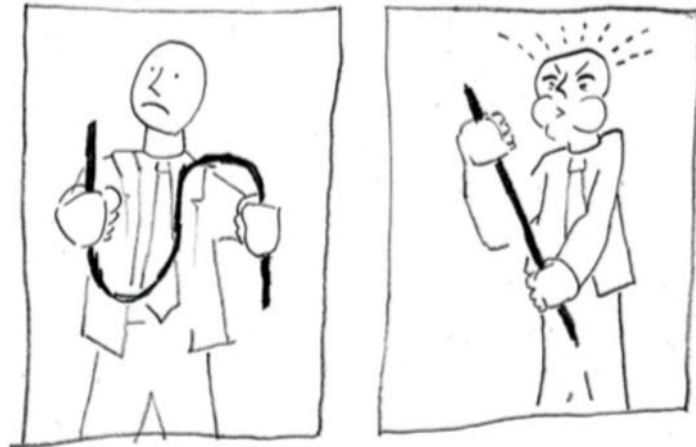
$|\varphi\rangle$

Alice

Bob

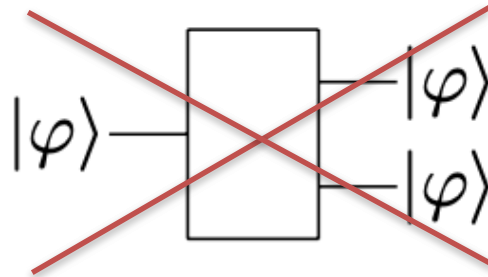
$|\varphi\rangle$

EPR:  $\frac{|00\rangle + |11\rangle}{\sqrt{2}}$

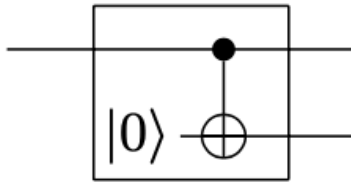
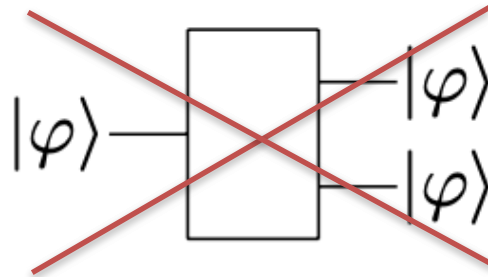


© Aleks Kissinger

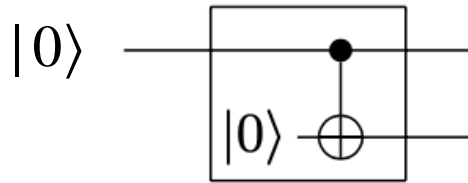
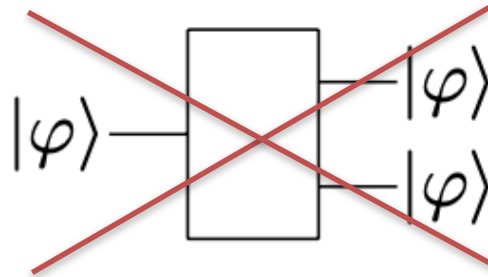
# No-cloning



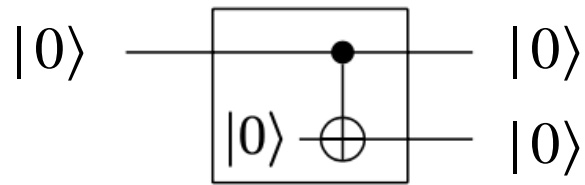
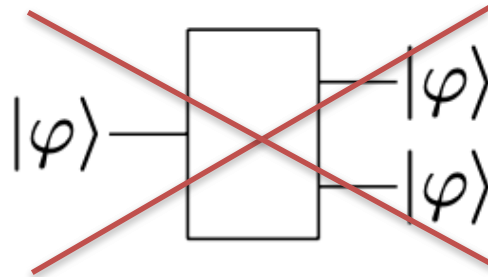
# No-cloning



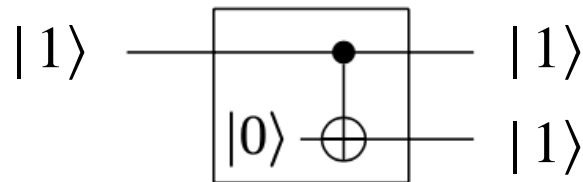
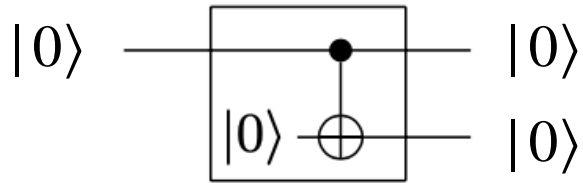
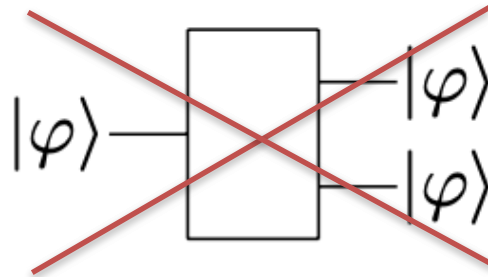
# No-cloning



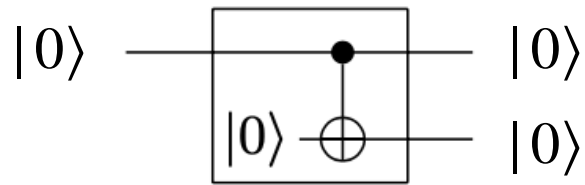
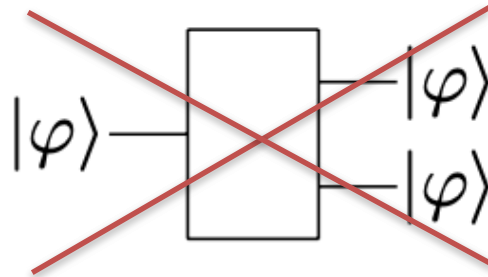
# No-cloning



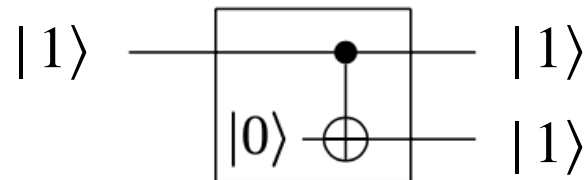
# No-cloning



# No-cloning

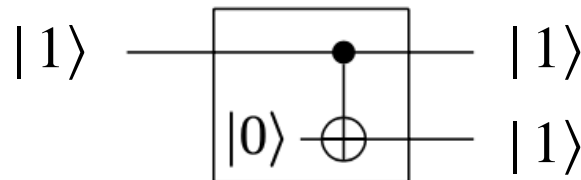
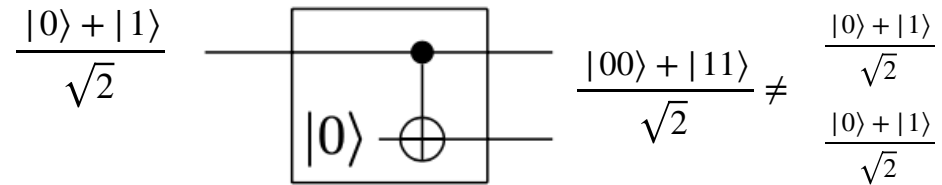
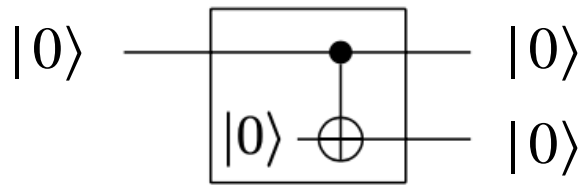
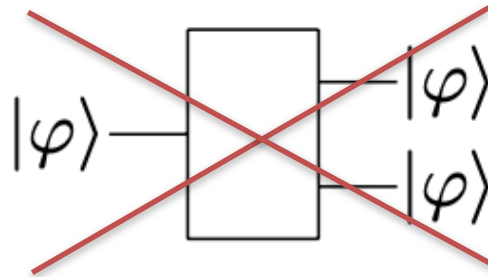


$$\frac{|0\rangle + |1\rangle}{\sqrt{2}} \text{ --- } \boxed{\text{CNOT}} \text{ --- } \frac{|00\rangle + |11\rangle}{\sqrt{2}} \neq \frac{|0\rangle + |1\rangle}{\sqrt{2}} \otimes \frac{|0\rangle + |1\rangle}{\sqrt{2}}$$



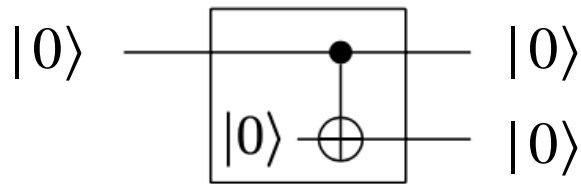
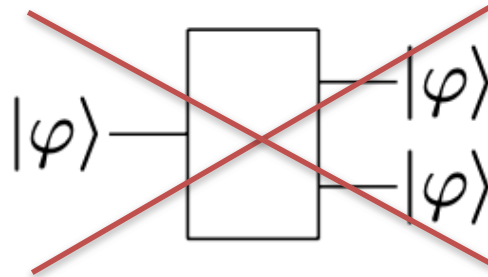


# No-cloning

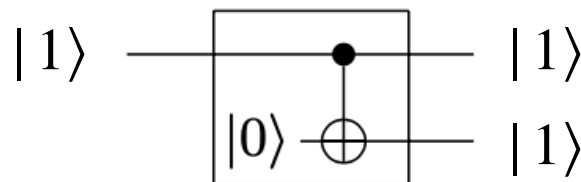


Copie  $|0\rangle$  et  $|1\rangle$  mais pas les autres états quantiques.

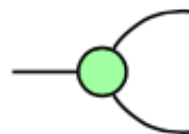
# No-cloning



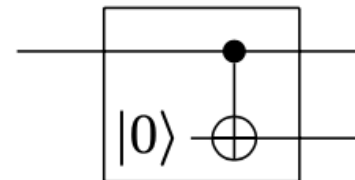
$$\frac{|0\rangle + |1\rangle}{\sqrt{2}} \text{ --- } \boxed{\text{CNOT}} \text{ --- } \frac{|00\rangle + |11\rangle}{\sqrt{2}} \neq \frac{|0\rangle + |1\rangle}{\sqrt{2}} \otimes \frac{|0\rangle + |1\rangle}{\sqrt{2}}$$



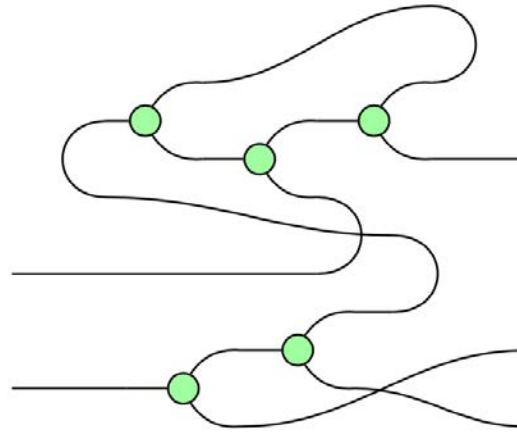
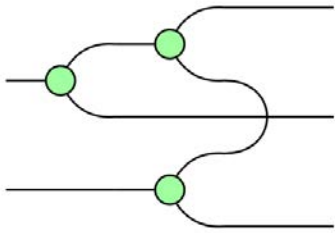
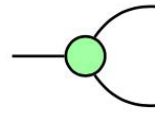
Copie  $|0\rangle$  et  $|1\rangle$  mais pas les autres états quantiques.



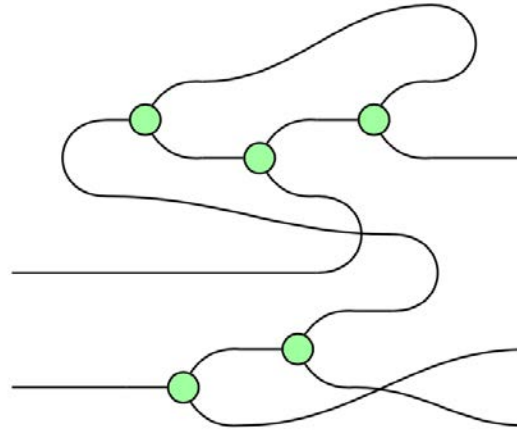
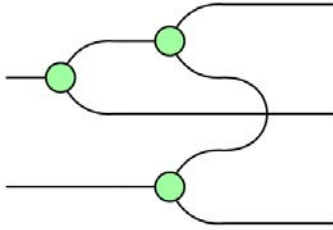
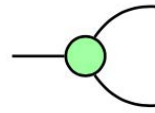
:=



# Copie

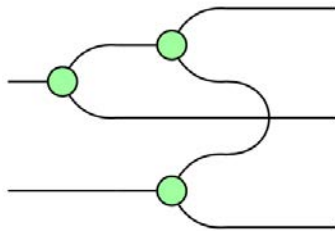
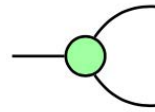


# Copie

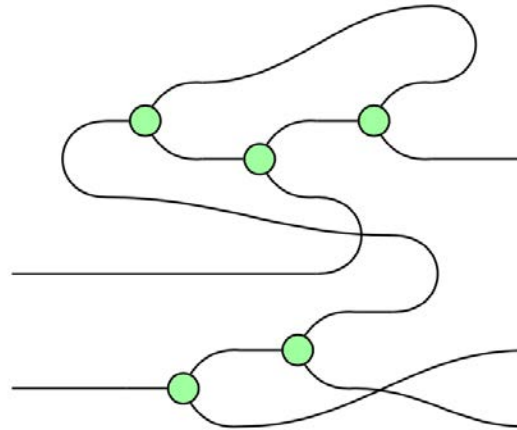


**Propriété :** Tous les  $(-\bullet, (, ))$ -diagrammes  $n \rightarrow m$  connexes sont égaux.

# Copie

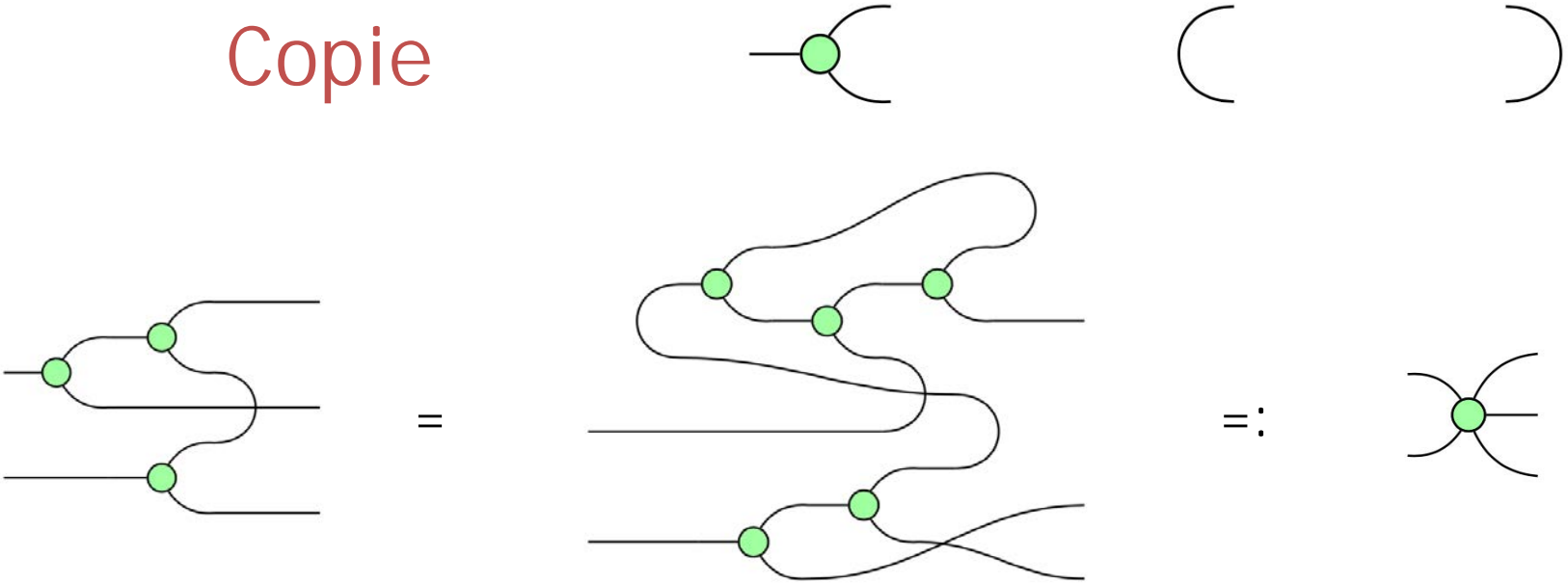


=



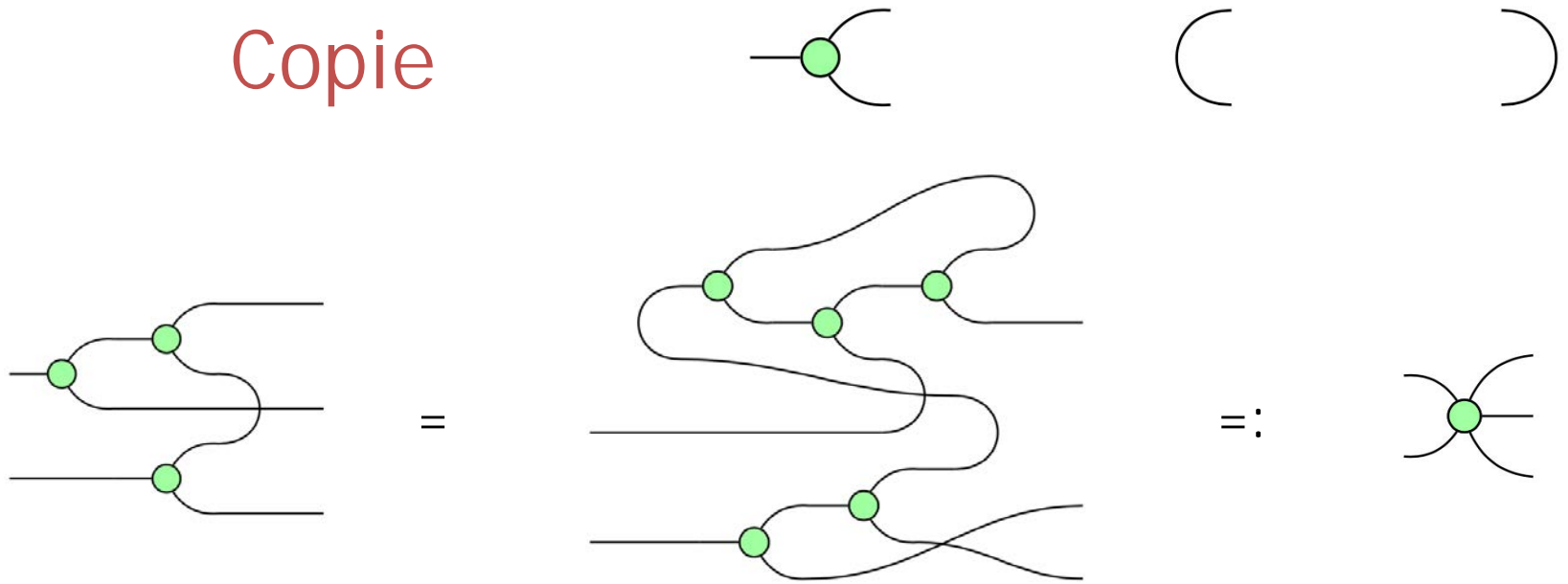
**Propriété :** Tous les  $(-\bullet, (, ))$ -diagrammes  $n \rightarrow m$  connexes sont égaux.

# Copie



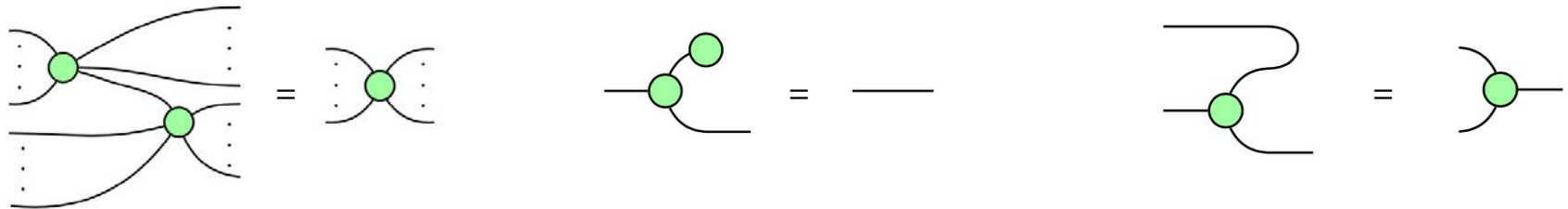
**Propriété :** Tous les  $(-\bullet, (, ))$ -diagrammes  $n \rightarrow m$  connexes sont égaux.

# Copie

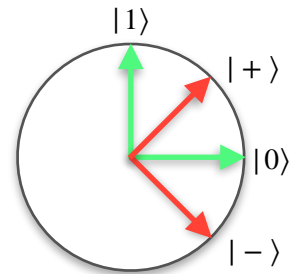
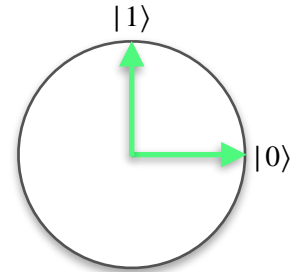
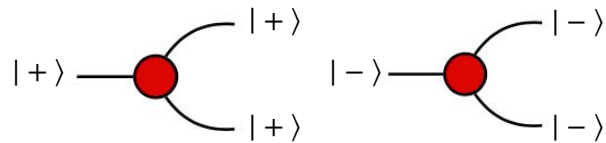
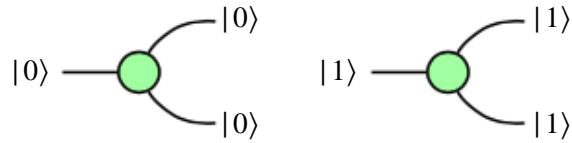


**Propriété :** Tous les  $(\text{---}\bullet, (, ))$ -diagrammes  $n \rightarrow m$  connexes sont égaux.

Exemples :

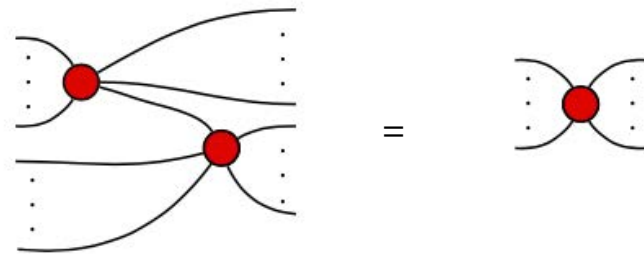
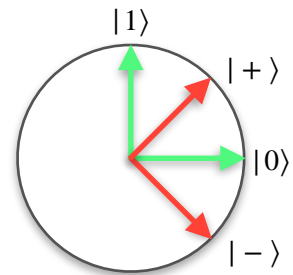
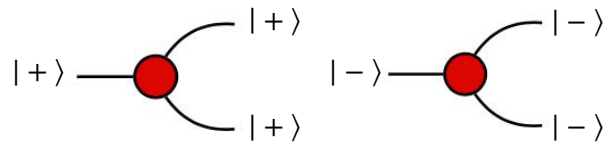
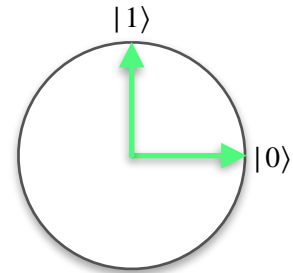
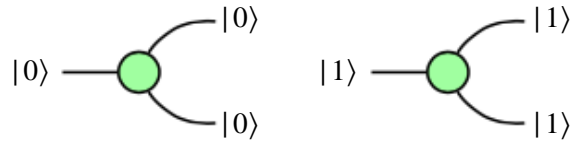


# Copies et Bases Complémentaires

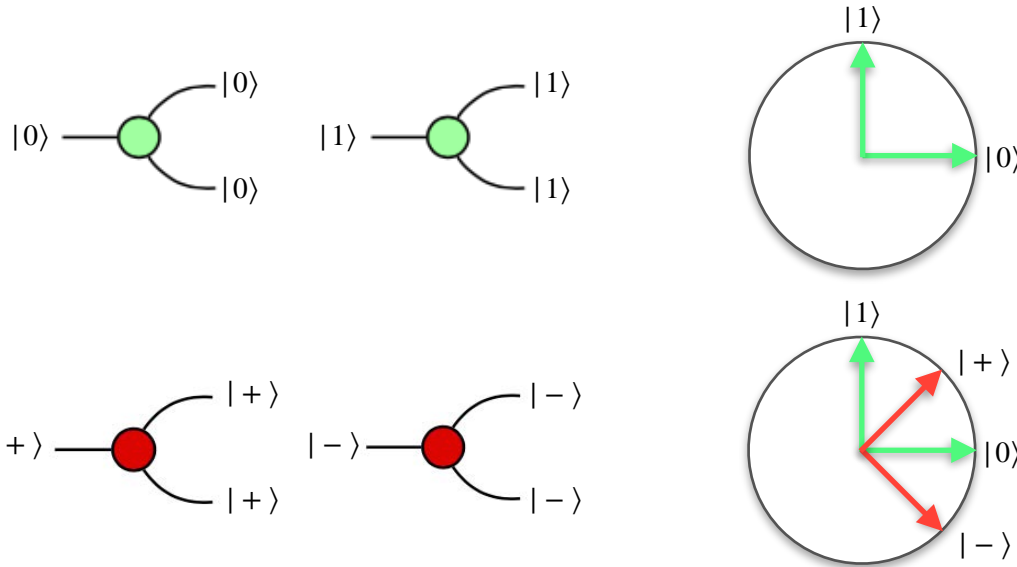




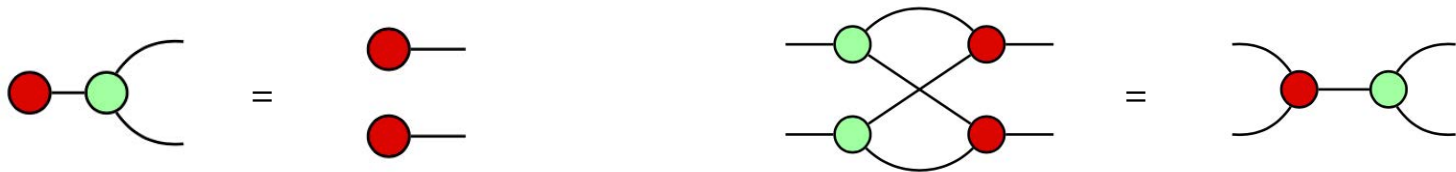
# Copies et Bases Complémentaires



# Copies et Bases Complémentaires



Graphiquement, deux bases sont complémentaires si :



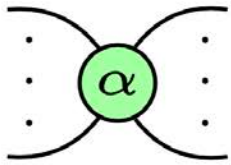
**Principe d'incertitude d'Heisenberg** : Des propriétés dans des bases complémentaires ne sont pas accessibles en même temps.

# Langages graphiques en informatique quantique

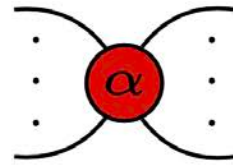
## Plan

- Contexte / historique
- Pourquoi un langage graphique ?
- ZX-calcul
- Applications
- Extensions

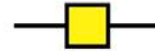
# ZX-diagrammes [Coecke, Duncan'08]



$|0..0\rangle \mapsto |0..0\rangle$   
 $|1..1\rangle \mapsto e^{i\alpha}|1..1\rangle$



$|+..+\rangle \mapsto |+..+\rangle$   
 $|-\dots-\rangle \mapsto e^{i\alpha}|-\dots-\rangle$

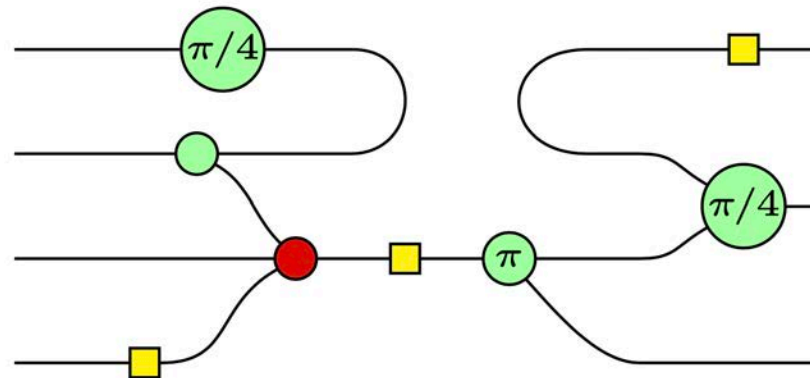


$|0\rangle \leftrightarrow |+\rangle$   
 $|1\rangle \leftrightarrow |-\rangle$

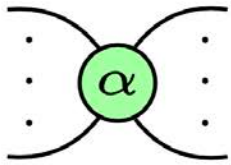


$\frac{|00\rangle + |11\rangle}{\sqrt{2}}$

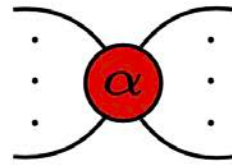
Mesure de Bell  
(postselectionnée)



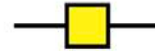
# ZX-diagrammes [Coecke, Duncan'08]



$|0..0\rangle \mapsto |0..0\rangle$   
 $|1..1\rangle \mapsto e^{i\alpha} |1..1\rangle$



$|+..+\rangle \mapsto |+..+\rangle$   
 $|-..-\rangle \mapsto e^{i\alpha} |-..-\rangle$

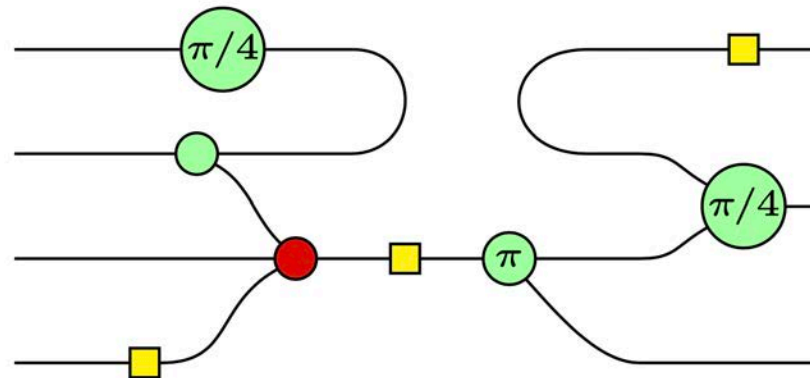


$|0\rangle \leftrightarrow |+\rangle$   
 $|1\rangle \leftrightarrow |-\rangle$



$\frac{|00\rangle + |11\rangle}{\sqrt{2}}$

Mesure de Bell  
(postselectionnée)



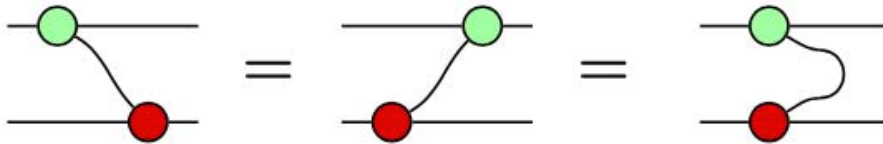
Universalité : Toute évolution quantique post-sélectionnée peut être représentée par un ZX-diagramme.

-> Strictement plus expressif que les circuits quantiques

# “Seule la connectivité compte”



Propriété : Seule la connectivité compte.



# “Seule la connectivité compte”



Propriété : Seule la connectivité compte.



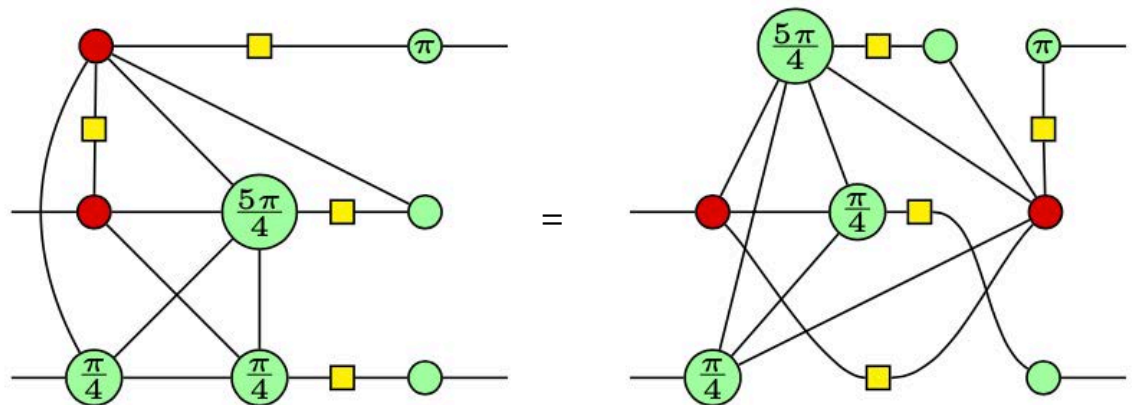
# “Seule la connectivité compte”



Propriété : Seule la connectivité compte.

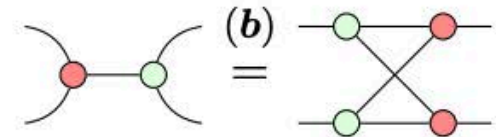
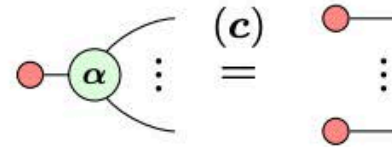
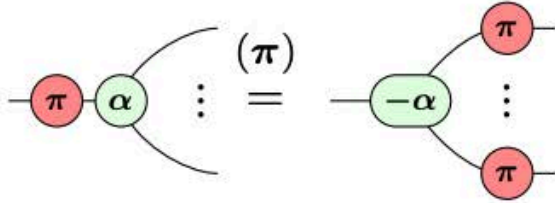
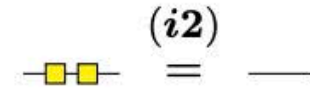
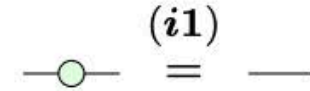
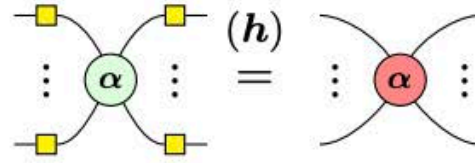
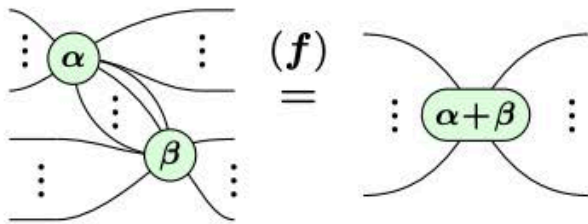


Les ZX-diagrammes peuvent être déformés à volonté.





# ZX-calcul



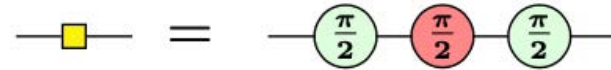
Complétude ?

Etant donné deux ZX-diagrammes représentant la même évolution quantique, est-ce qu'on peut toujours transformer l'un en l'autre en utilisant les équations du ZX-calcul ?

# (In-)Complétude(s)

Incomplétudes :

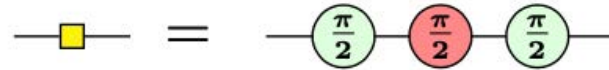
- Décomposition d'Euler est nécessaire [Duncan, Perdrix'09] :



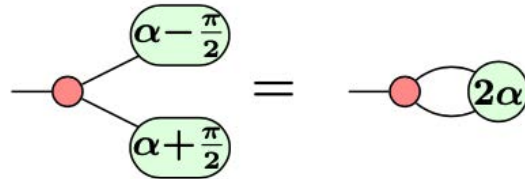
# (In-)Complétude(s)

Incomplétudes :

- Décomposition d'Euler est nécessaire [Duncan, Perdrix'09] :



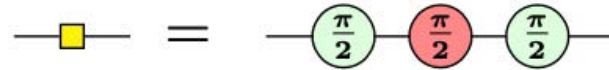
- Supplémentarité est nécessaire [Perdrix, Wang'15] :



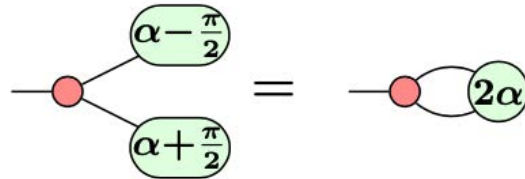
# (In-)Complétude(s)

Incomplétudes :

- Décomposition d'Euler est nécessaire [Duncan, Perdrix'09] :



- Supplémentarité est nécessaire [Perdrix, Wang'15] :



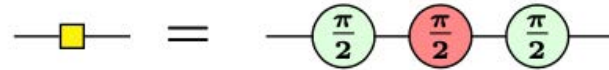
Complétudes :

- $ZX_{\frac{\pi}{2}}$  est complet pour le fragment *stabilisable* (ou *Clifford*) [Backens'14]

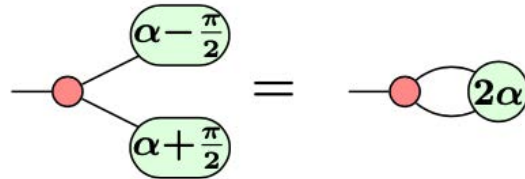
# (In-)Complétude(s)

Incomplétudes :

- Décomposition d'Euler est nécessaire [Duncan, Perdrix'09] :



- Supplémentarité est nécessaire [Perdrix, Wang'15] :



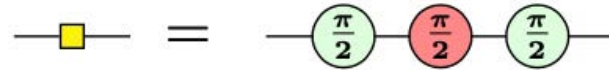
Complétudes :

- $ZX_{\frac{\pi}{2}}$  est complet pour le fragment *stabilisable* (ou *Clifford*) [Backens'14]
- $ZX_{\frac{\pi}{4}}$  est complet pour le fragment universel *Clifford+T* [Jeandel, Perdrix, Vilmart'18]

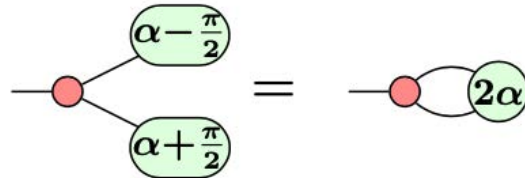
# (In-)Complétude(s)

Incomplétudes :

- Décomposition d'Euler est nécessaire [Duncan, Perdrix'09] :



- Supplémentarité est nécessaire [Perdrix, Wang'15] :



Complétudes :

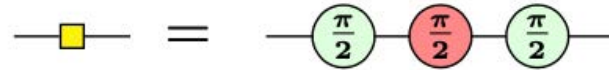
- $ZX_{\frac{\pi}{2}}$  est complet pour le fragment *stabilisable* (ou *Clifford*) [Backens'14]
- $ZX_{\frac{\pi}{4}}$  est complet pour le fragment universel *Clifford+T* [Jeandel, Perdrix, Vilmart'18]

Remarque : Il n'existe pas de résultat de complétude pour un fragment universel des circuits quantiques.

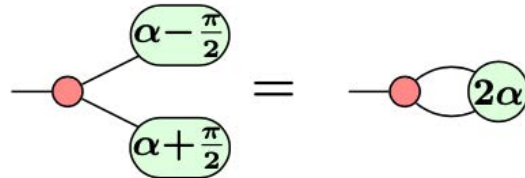
# (In-)Complétude(s)

Incomplétudes :

- Décomposition d'Euler est nécessaire [Duncan, Perdrix'09] :



- Supplémentarité est nécessaire [Perdrix, Wang'15] :

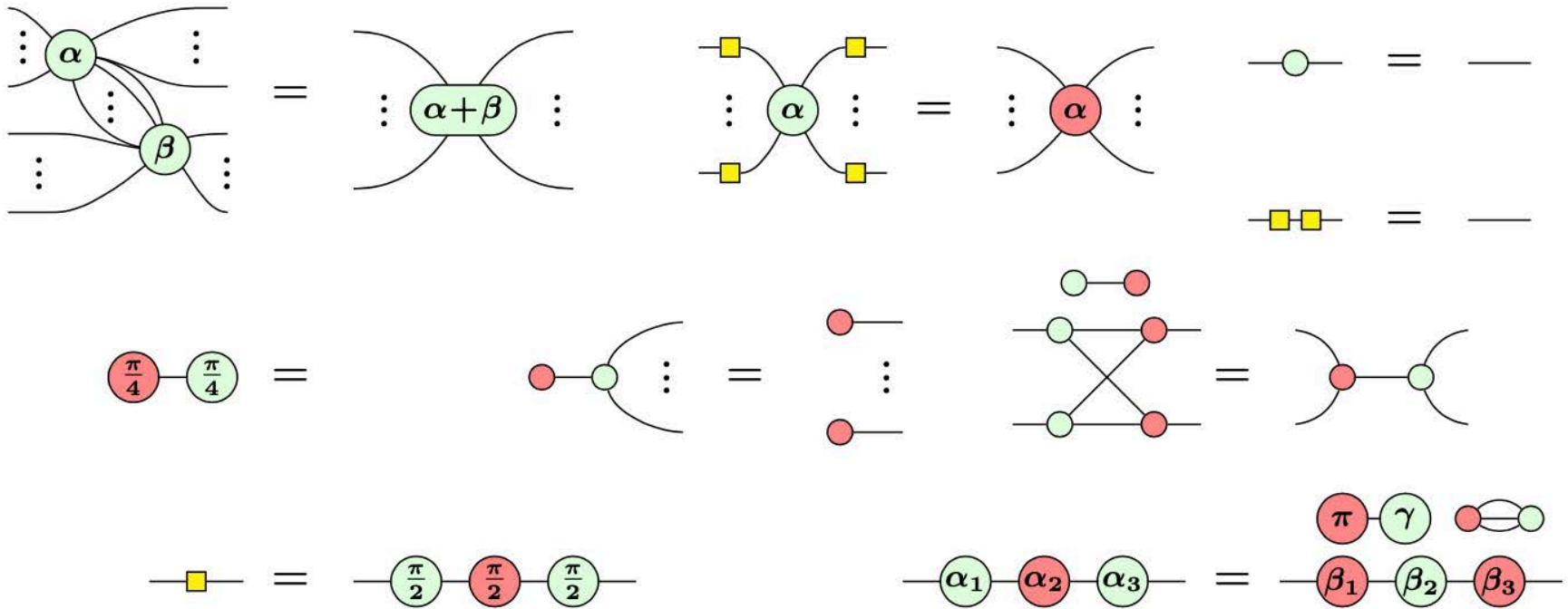


Complétudes :

- $ZX_{\frac{\pi}{2}}$  est complet pour le fragment *stabilisable* (ou *Clifford*) [Backens'14]
- $ZX_{\frac{\pi}{4}}$  est complet pour le fragment universel *Clifford+T* [Jeandel, Perdrix, Vilmart'18]
- $ZX$  est complet [Hadzihasanovic, Ng, Wang'18] [Vilmart'19]

Remarque : Il n'existe pas de résultat de complétude pour un fragment universel des circuits quantiques.

# (In-)Complétude(s)



Théorie équationnelle complète du ZX-calcul [Vilmart'19]

4

- ZX est complet [Hadzihasanovic, Ng, Wang'18] [Vilmart'19]

Remarque : Il n'existe pas de résultat de complétude pour un fragment universel des circuits quantiques.



# Langages graphiques en informatique quantique

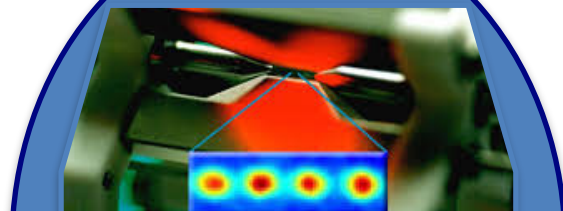
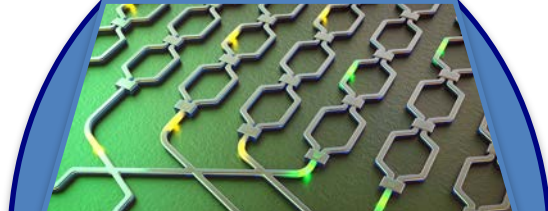
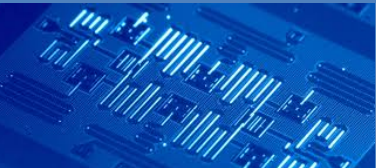
## Plan

- Contexte / historique
- Pourquoi un langage graphique ?
- ZX-calcul
- Applications
- Extensions

# Pile Quantique



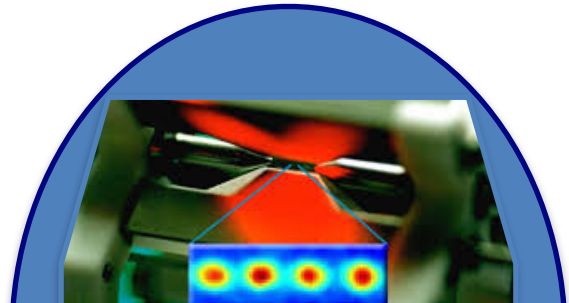
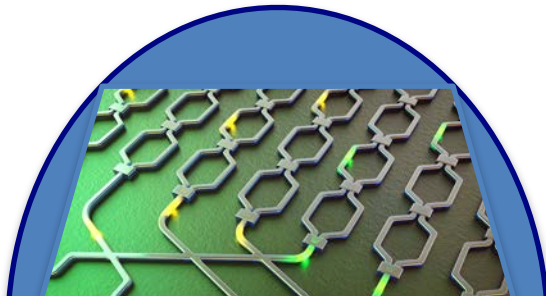
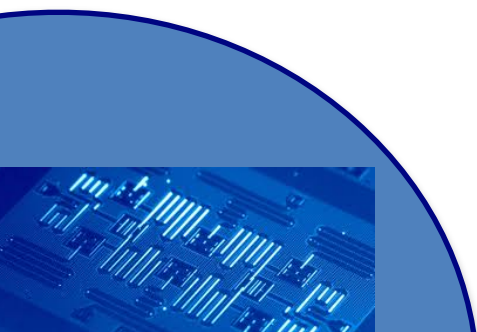
Algos Q



# Pile Quantique



Langages de prog de haut niveau  
Quipper, Qiskit, ...



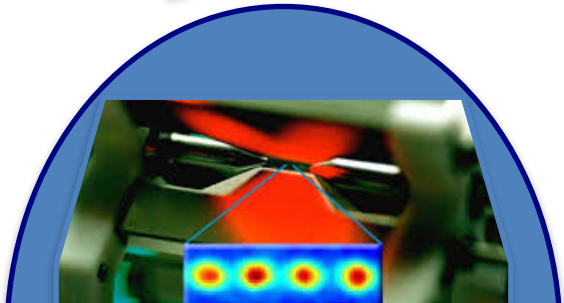
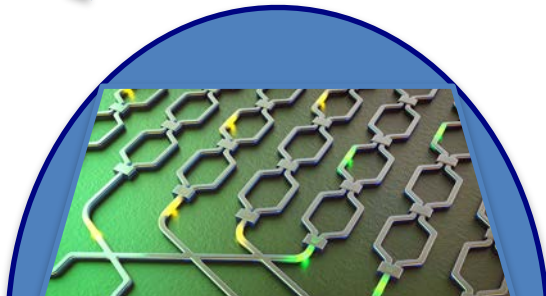
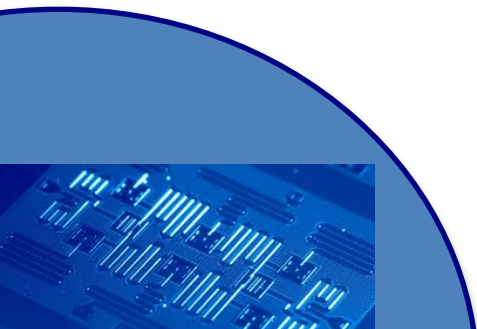
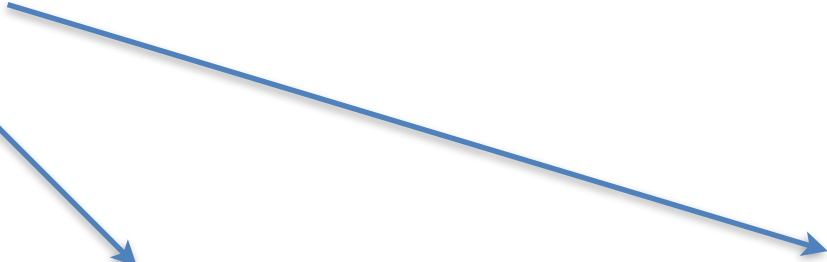
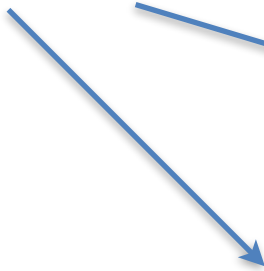
# Pile Quantique



Langages de prog de haut niveau  
Quipper, Qiskit, ...



Langages Intermédiaires  
ZX-calcul, circuits quantiques



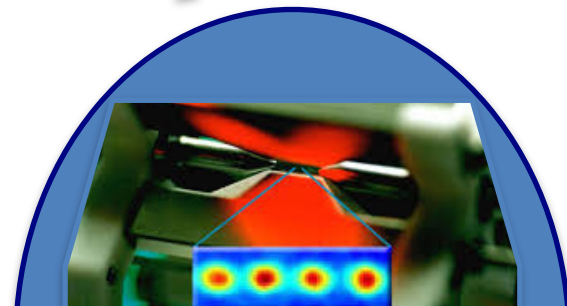
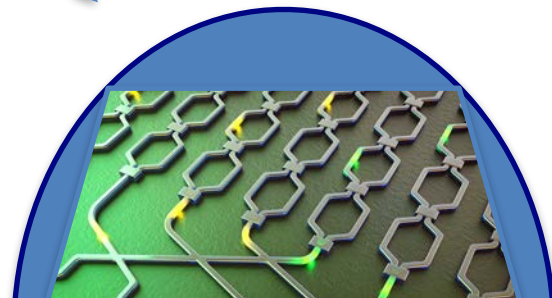
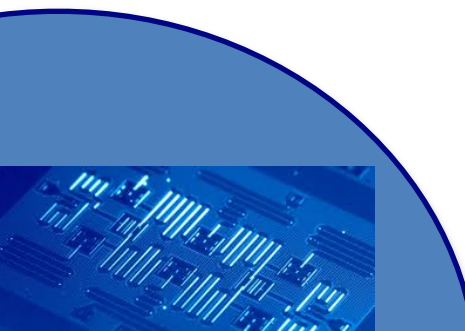
# Pile Quantique



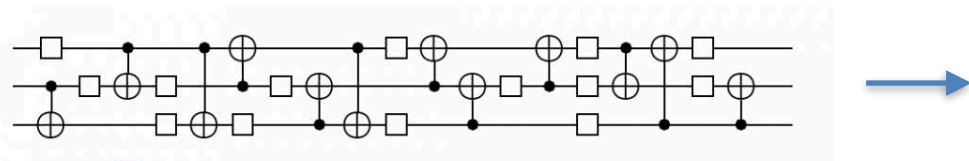
Langages de prog de haut niveau  
Quipper, Qiskit, ...

Langages Intermédiaires  
ZX-calcul, circuits quantiques

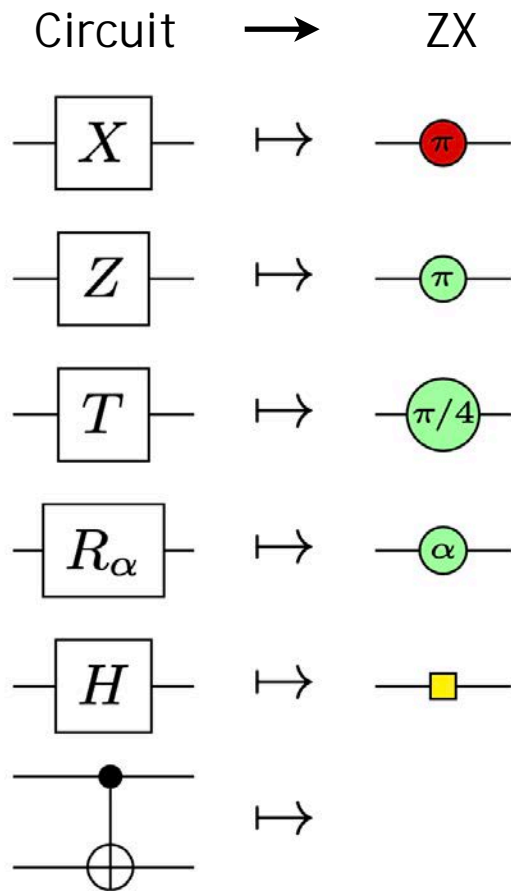
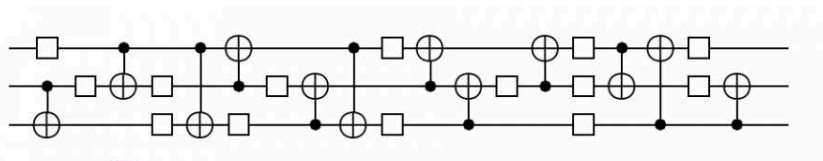
- Optimisation
- Adaptation aux contraintes : architecture, primitives, hybride, simulation
- Tolérance aux fautes



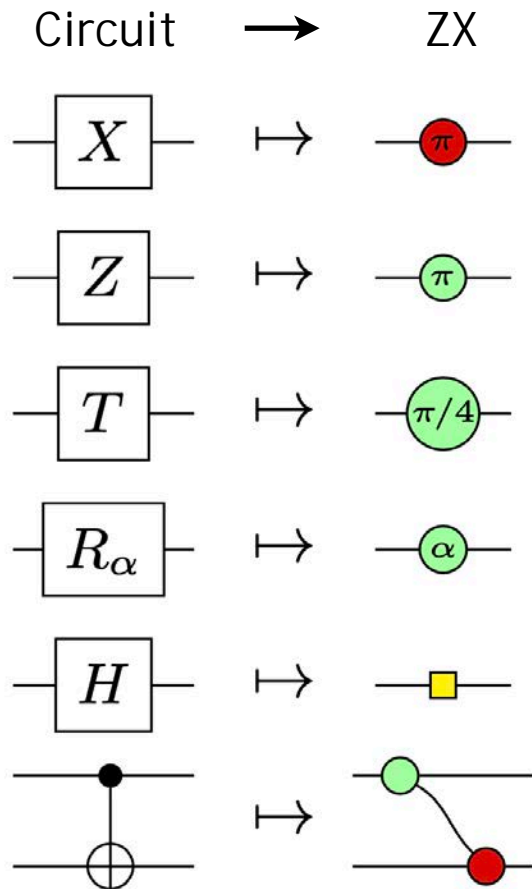
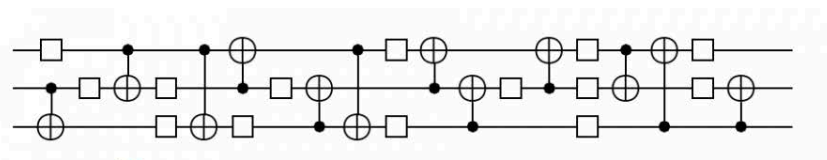
# Application - Optimisation de circuits



# Application - Optimisation de circuits

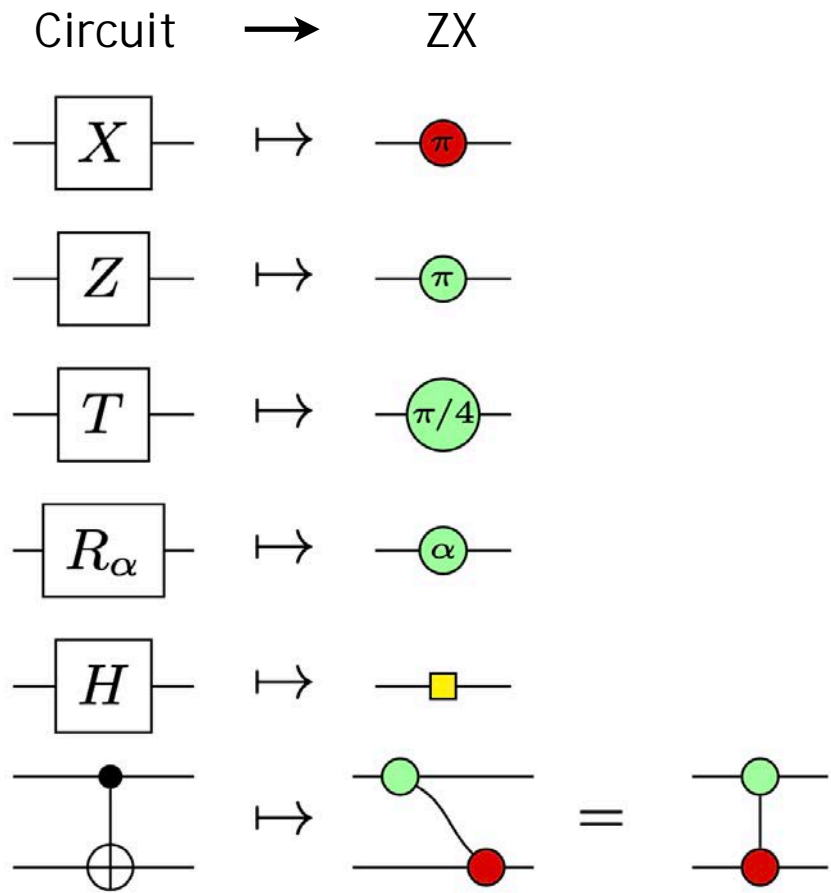
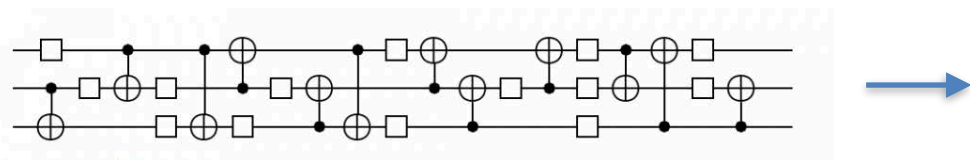


# Application - Optimisation de circuits

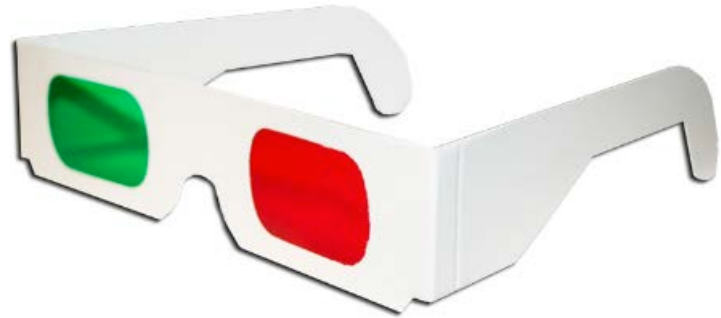
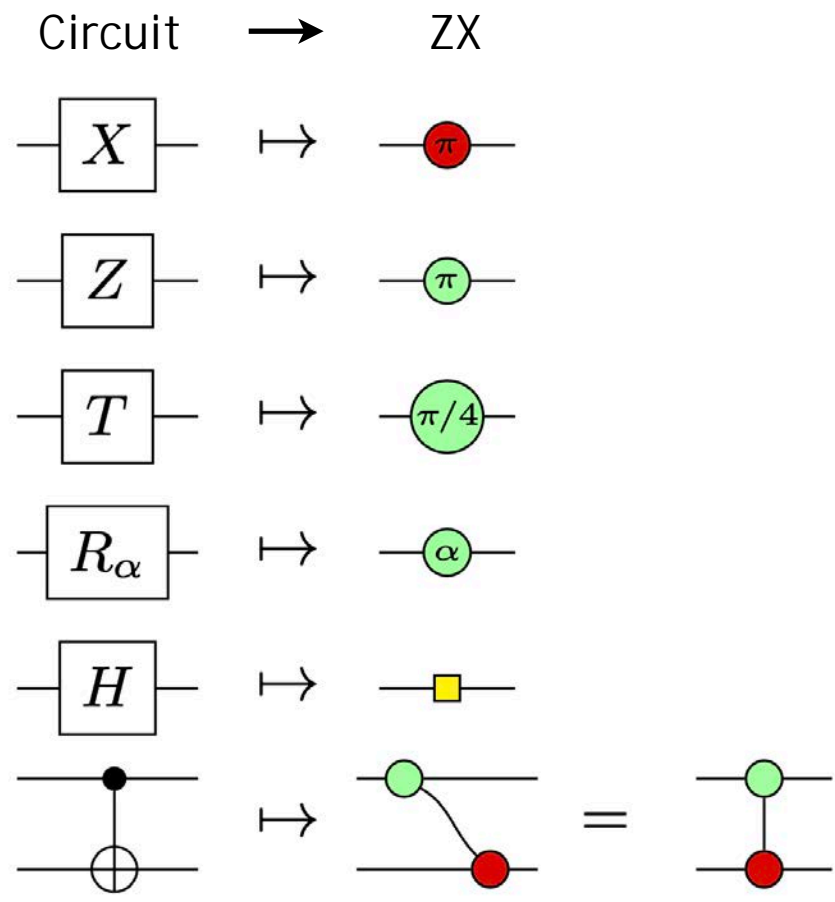
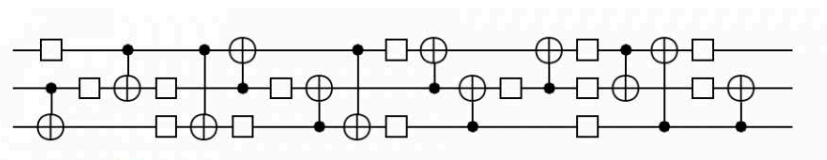




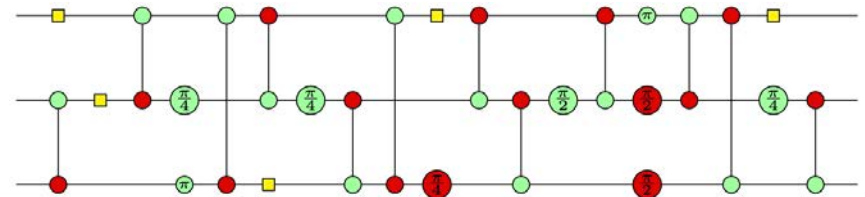
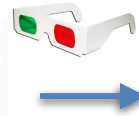
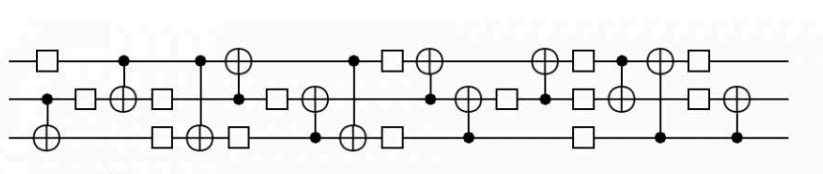
# Application - Optimisation de circuits



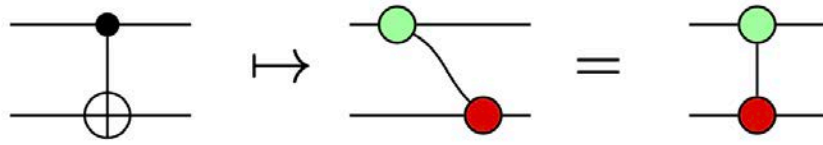
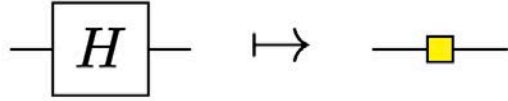
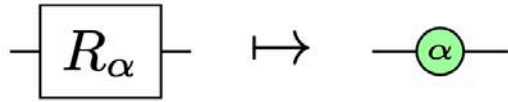
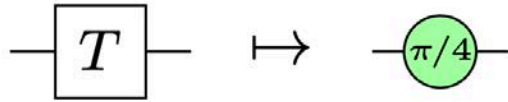
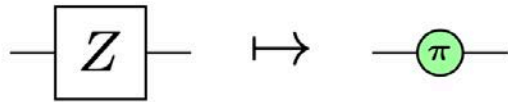
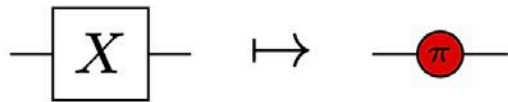
# Application - Optimisation de circuits



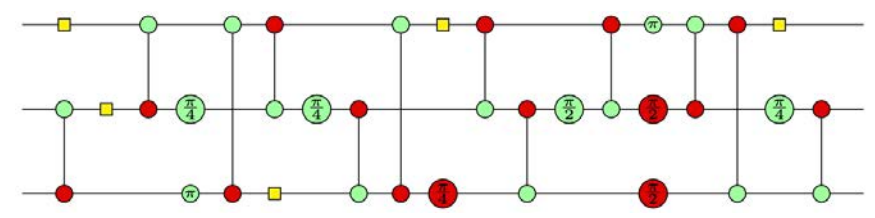
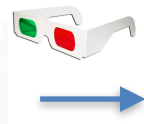
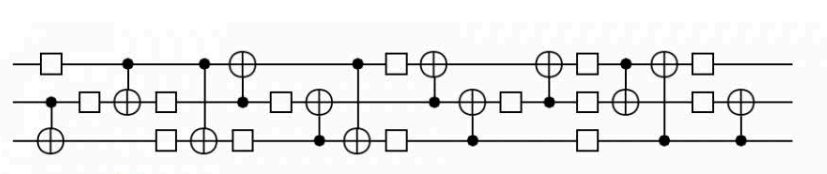
# Application - Optimisation de circuits



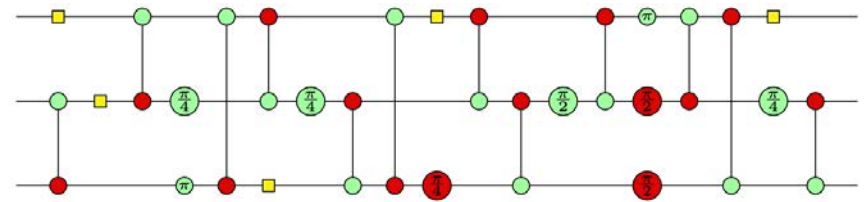
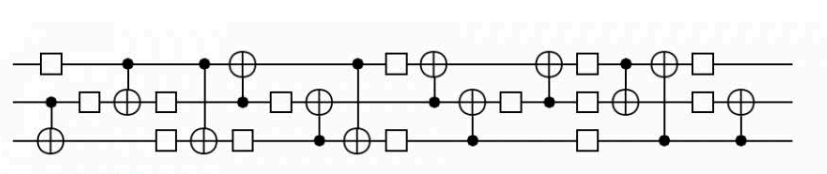
Circuit  $\rightarrow$  ZX



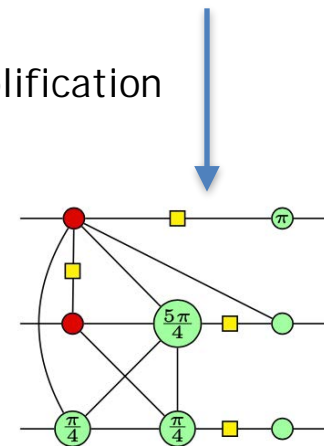
# Application - Optimisation de circuits



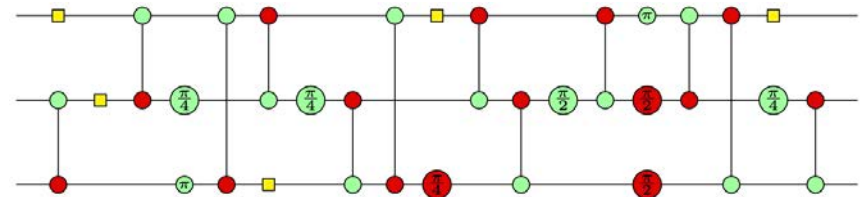
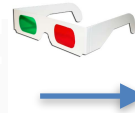
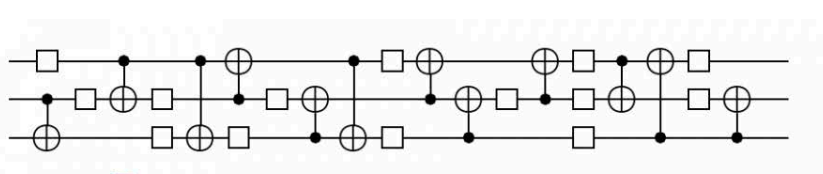
# Application - Optimisation de circuits



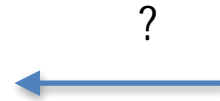
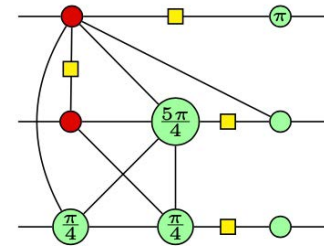
Simplification



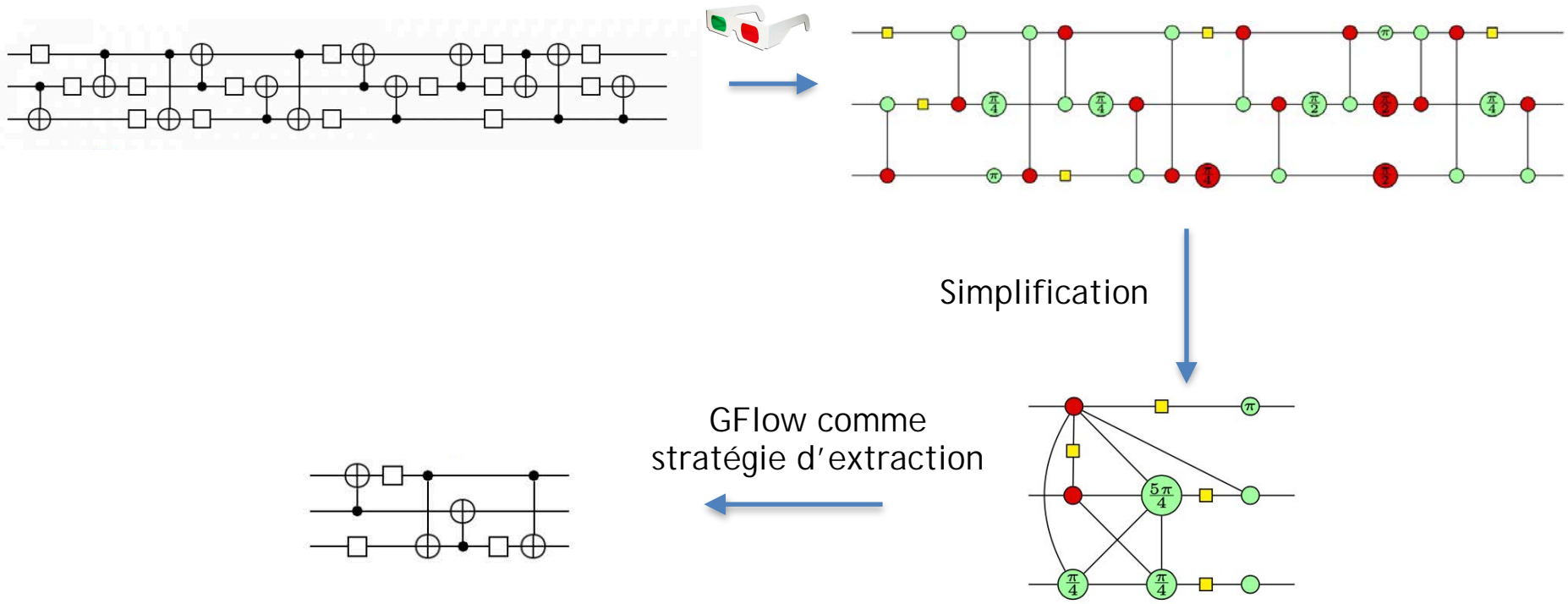
# Application - Optimisation de circuits



Simplification

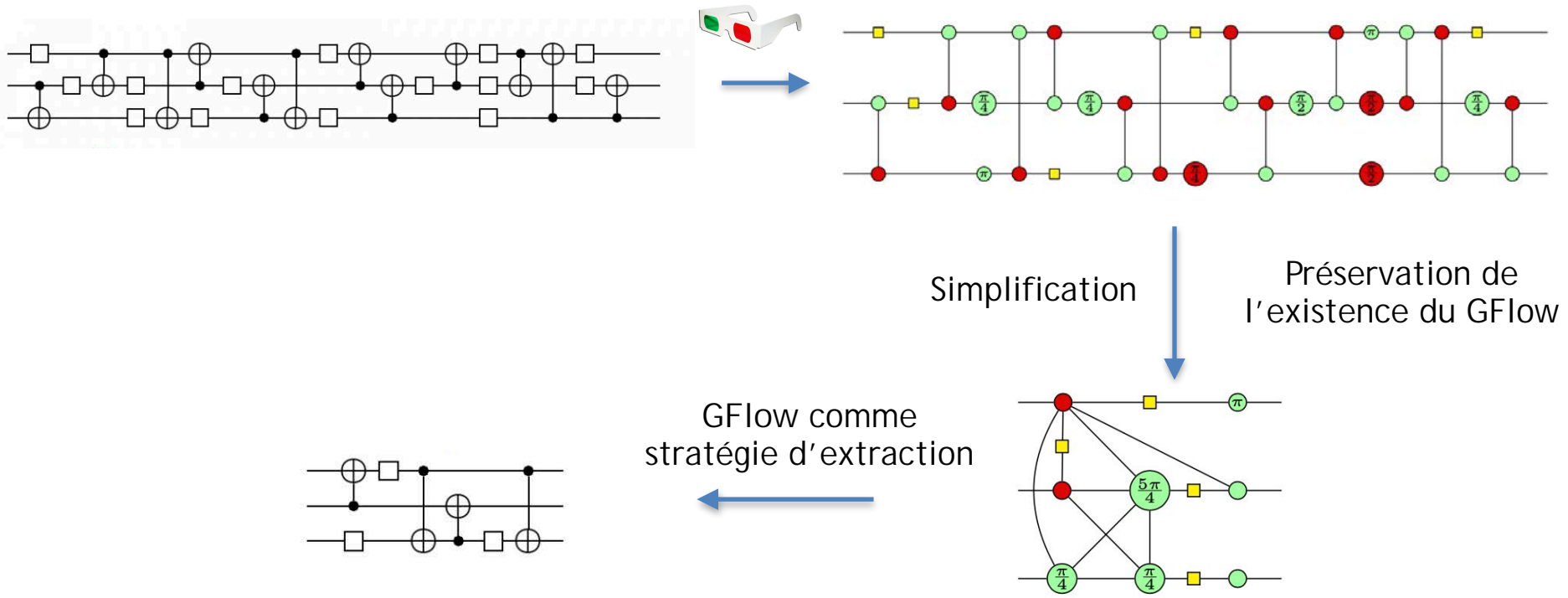


# Application - Optimisation de circuits



Graph-theoretic Simplification of Quantum Circuits with the ZX-calculus.  
 [Duncan, Kissinger, Perdrix, Van de Wetering. Quantum' 20]

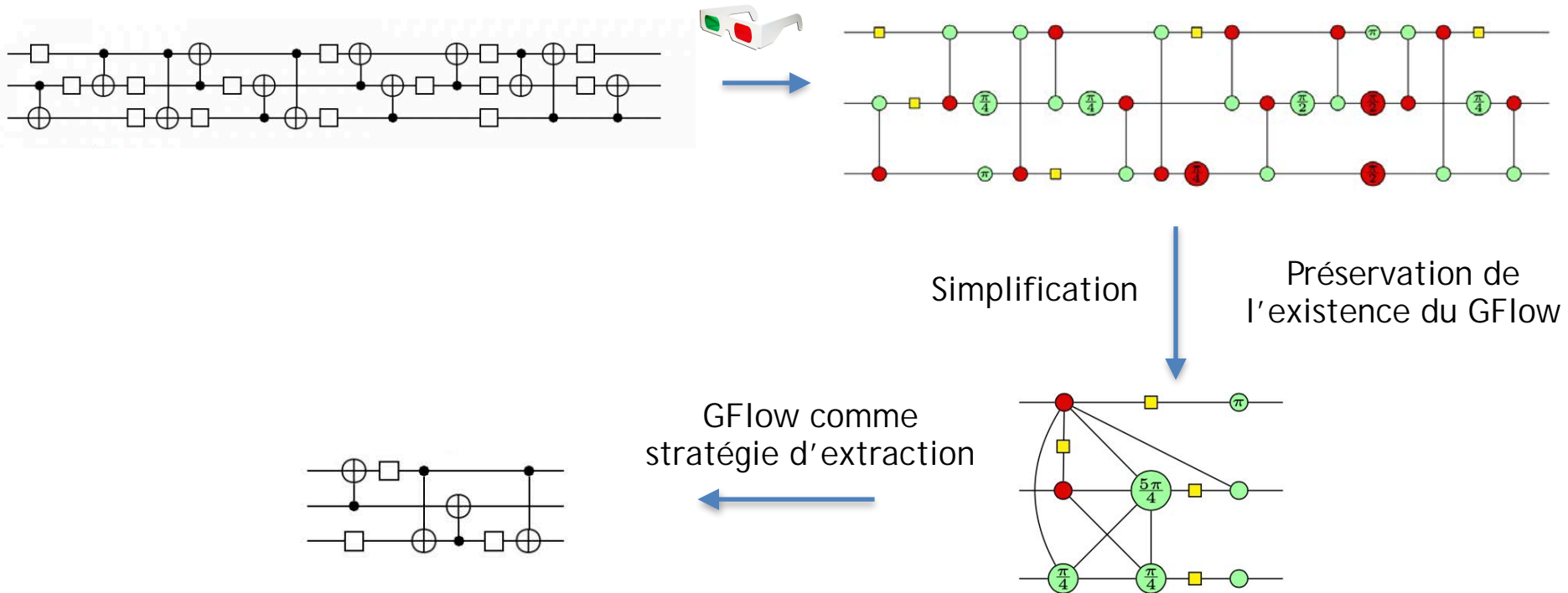
# Application - Optimisation de circuits



Graph-theoretic Simplification of Quantum Circuits with the ZX-calculus. [Duncan, Kissinger, Perdrix, Van de Wetering. Quantum' 20]



# Application - Optimisation de circuits



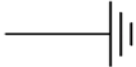
Graph-theoretic Simplification of Quantum Circuits with the ZX-calculus.  
[Duncan, Kissinger, Perdrix, Van de Wetering. Quantum'20]

Implementation et validation expérimentale sur benchmark  
[pyZX. Kissinger, Van de Wetering] (<https://pyzx.readthedocs.io/>)

# Langages graphiques en informatique quantique

- Contexte / historique
- Pourquoi un langage graphique ?
- ZX-calcul
- Applications
- Extensions

# Extensions

- Calcul hybride classique / quantique<sup>1,2,3</sup> : 


---

1- Coecke, Perdrix. Environment and classical channels in categorical quantum mechanics. CSL'10

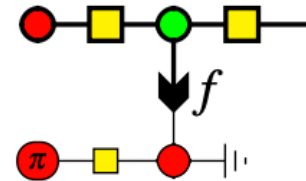
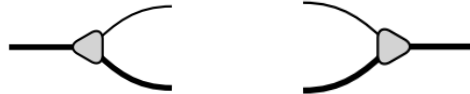
2- Selinger. Dagger compact closed categories and completely positive maps. QPL'07

3- Carette, Jeandel, Perdrix, Vilmart. Completeness of Graphical Languages for Mixed States Quantum Mechanics. ICALP'19

# Extensions

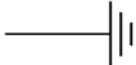
- Calcul hybride classique / quantique<sup>1,2,3</sup> : 

- Passage à l'échelle<sup>4,5</sup> :

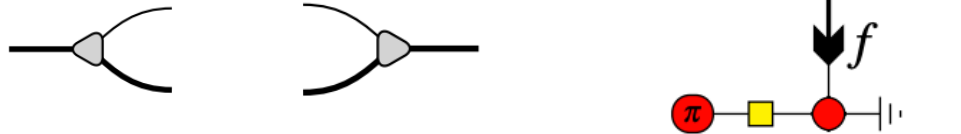


- 
- 1- Coecke, Perdrix. Environment and classical channels in categorical quantum mechanics. CSL'10
  - 2- Selinger. Dagger compact closed categories and completely positive maps. QPL'07
  - 3- Carette, Jeandel, Perdrix, Vilmart. Completeness of Graphical Languages for Mixed States Quantum Mechanics. ICALP'19
  - 4- Carette, Horsman, Perdrix. SZX-calculus: Scalable Graphical Quantum Reasoning. MFCS'19
  - 5- Carette, D'Anello, Perdrix. Quantum Algorithms and Oracles with the Scalable ZX-calculus. arXiv:2104.01043

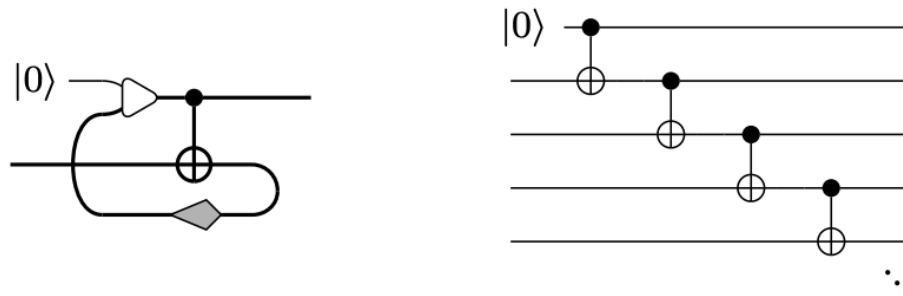
# Extensions

- Calcul hybride classique / quantique<sup>1,2,3</sup> : 

- Passage à l'échelle<sup>4,5</sup> :



- Stream de qubits<sup>6</sup> :



1- Coecke, Perdrix. Environment and classical channels in categorical quantum mechanics. CSL'10  
 2- Selinger. Dagger compact closed categories and completely positive maps. QPL'07  
 3- Carette, Jeandel, Perdrix, Vilmart. Completeness of Graphical Languages for Mixed States Quantum Mechanics. ICALP'19  
 4- Carette, Horsman, Perdrix. SZX-calculus: Scalable Graphical Quantum Reasoning. MFCS'19  
 5- Carette, D'Anello, Perdrix. Quantum Algorithms and Oracles with the Scalable ZX-calculus. arXiv:2104.01043  
 6- Carette, de Visme, Perdrix. Graphical Language with Delayed Trace: Picturing Quantum Computing with Finite Memory. LICS'21

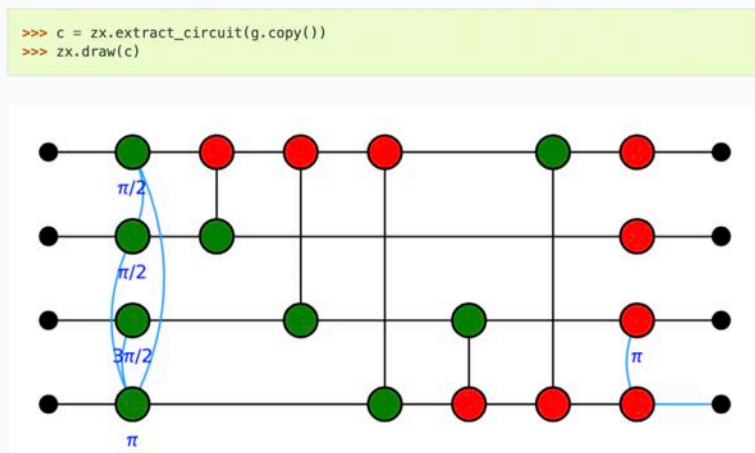
# Références

*Picturing Quantum Processes*. B. Coecke, A. Kissinger. Cambridge University Press.

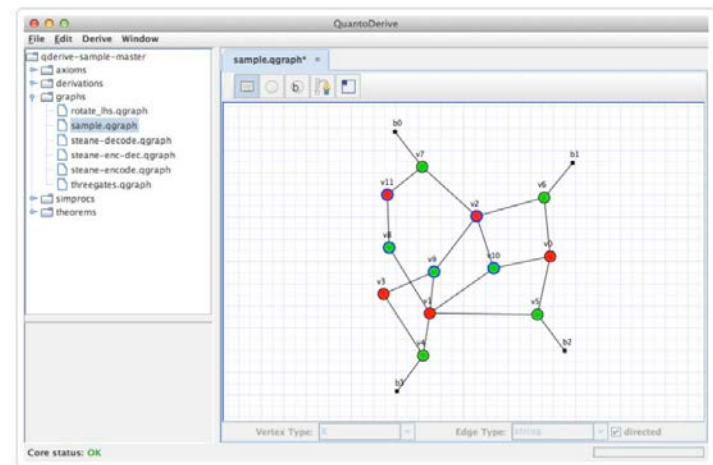
*ZX-calculus for the working quantum computer scientist*. J. van de Wetering. arXiv:2012.13966

*A survey of graphical languages for monoidal categories*. P. Selinger. NSP 2010.

*Approches Graphiques en Informatique Quantique*. (HDR) S. Perdrix.



<https://pyzx.readthedocs.io/>



<https://quantomatic.github.io/>

<https://zxcalculus.com/>