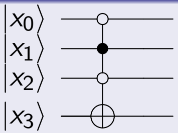
Portes CNOT

Porte de Pauli X contrôlée par n qubits

On note le contrôle par 1 avec $\bullet$ (le qubit de sortie $|x_3\rangle$ vaut 1 quant le qubit de contrôle vaut 1) ;
On note le contrôle par 0 avec $\circ$ (le qubit de sortie $|x_3\rangle$ vaut 1 quant le qubit de contrôle vaut 0).
Le contrôle global est un ET des contrôles individuels.

Exemple



$$|x_0, x_1, x_2, x_3\rangle \mapsto |x_0, x_1, x_2, x_3 \oplus (\neg x_1 \wedge x_2 \wedge \neg x_3)\rangle$$

Construction du circuit

4 étapes de compilation

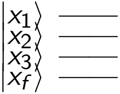
1. Écriture de la table de vérité,
2. Pour chaque sortie donnant 1, former une porte `NOT` controlée. Chaque entrée va servir de contrôle, par 1 si l'entrée est à 1, et par 0 si l'entrée est à 0,
3. Développement du circuit pour n'avoir que des portes `NOT` contrôlées par 0,
4. Simplification du circuit.

Construction du circuit

Exemple

Soit la fonction booléenne
$f(x_1, x_2, x_3) =$
$(x_1 \wedge x_2) \vee (x_3 \wedge \neg x_2) \vee (x_1 \wedge x_3)$.

x_1	x_2	x_3	$F(x_1,\ x_2,\ x_3)$
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1



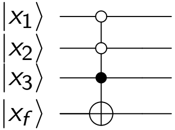
Circuit quantique pour $f(x_1, x_2, x_3)$

Construction du circuit

Exemple

Soit la fonction booléenne $f(x_1, x_2, x_3) = (x_1 \wedge x_2) \vee (x_3 \wedge \neg x_2) \vee (x_1 \wedge x_3)$.

x_1	x_2	x_3	$F(x_1,\ x_2,\ x_3)$
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1



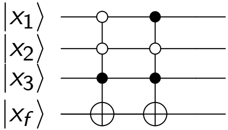
Circuit quantique pour $f(x_1, x_2, x_3)$

Construction du circuit

Exemple

Soit la fonction booléenne $f(x_1, x_2, x_3) = (x_1 \wedge x_2) \vee (x_3 \wedge \neg x_2) \vee (x_1 \wedge x_3)$.

x_1	x_2	x_3	$F(x_1, x_2, x_3)$
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1



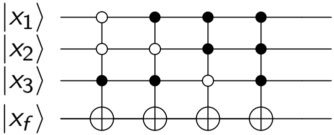
Circuit quantique pour $f(x_1, x_2, x_3)$

Construction du circuit

Exemple

Soit la fonction booléenne $f(x_1, x_2, x_3) = (x_1 \wedge x_2) \vee (x_3 \wedge \neg x_2) \vee (x_1 \wedge x_3)$.

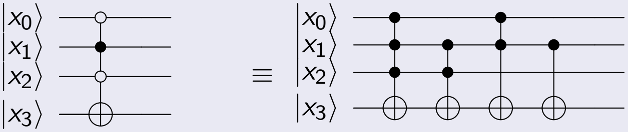
x_1	x_2	x_3	$F(x_1, x_2, x_3)$
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1



Circuit quantique pour $f(x_1, x_2, x_3)$

Développement du circuit

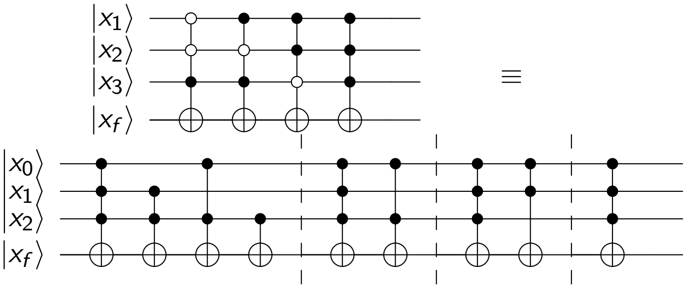
On transforme les contrôles par 0 en combinaison de contrôles par 1 :



Équivalent sans contrôles par 0

Développement du circuit

En reprennant l'exemple développé précédemment $f(x_1, x_2, x_3) = (x_1 \wedge x_2) \vee (x_3 \wedge \neg x_2) \vee (x_1 \wedge x_3)$, on développe pour obtenir le circuit suivant :

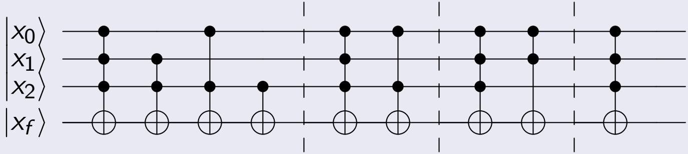


Équivalent sans contrôles par 0

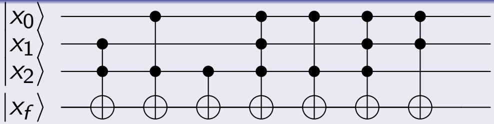
Simplification

On retire les paires de portes identiques qui s'annulent sur le circuit.

Circuit complet



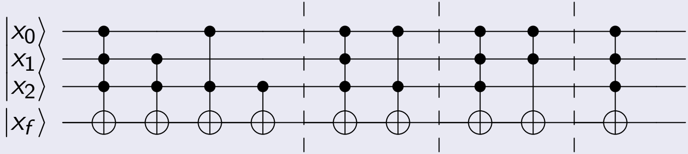
Première simplification



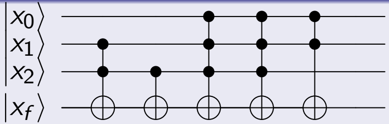
Simplification

On retire les paires de portes identiques qui s'annulent sur le circuit.

Circuit complet



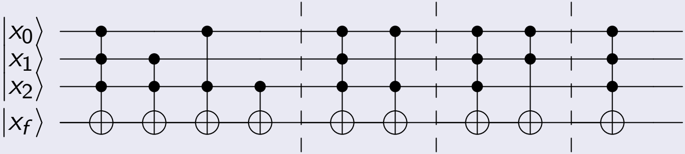
Deuxième simplification



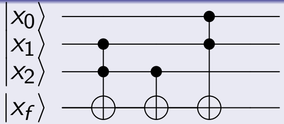
Simplification

On retire les paires de portes identiques qui s'annulent sur le circuit.

Circuit complet

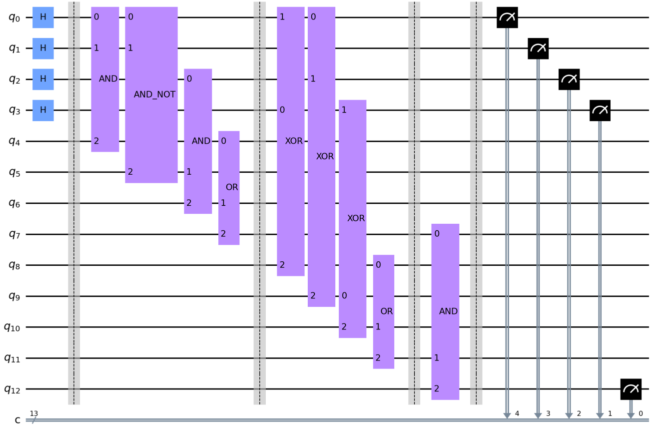


Troisième simplification : circuit final



Construction de circuits composés

On peux combiner des fonctions élémentaires pour former des circuits plus complexes.



$$f(a,b,c,d) = ((a \wedge b) \vee (a \wedge \neg b \wedge c)) \wedge ((d \oplus a) \vee (a \oplus c \oplus d))$$

Modèle d'Ising

Définition

L'énergie d'un système peut être donnée par l'opérateur Hamiltonien. On l'écrit avec le modèle d'Ising de la façon suivante :

$$\mathcal{H} = -\sum_{<i,j>} J_{ij}\sigma_i\sigma_j - \sum_i h_i\sigma_i. \tag{1}$$

On a d'un côté le terme $\sum h_i\sigma_i$ indiquant la contribution de chaque variable au système global, c'est le **biais** du système.

De l'autre côté, on a $\sum J_{ij}\sigma_i\sigma_j$ indiquant les interactions entre chaque paires de variables. C'est le **couplage** du système.

Recuit Quantique

Utilisation d'un opérateur Hamiltonien évoluant au cours du temps :

$$\mathcal{H}(t) = A(t)\mathcal{H}_0 + B(t)\mathcal{H}_1. \tag{2}$$

On fait évoluer $A(t)$ de 1 à 0, et $B(t)$ de 0 à 1.

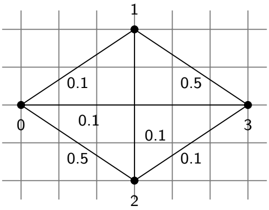
En prenant un Hamiltonien $\mathcal{H}_0$ tel qu'on puisse construire facilement son état minimal, on fait évoluer le système vers $\mathcal{H}_1$, en restant dans un état minimal du système (théorème adiabatique quantique).

Utilisation de D-Wave - 1

Résolution du problème du voyageur de commerce avec les outils fournis par D-Wave :

```python
import dwave_networkx
import networkx
import dimod
g = networkx.Graph()
g.add_weighted_edges_from({(0, 1, .1), (0, 2, .5), (0, 3, .1), (1, 2, .1),(1, 3, .5), (2, 3, .1)})

dwave_networkx.algorithms.traveling_salesperson(g, dimod.ExactSolver(), start=0)
```



Graphe TSP correspondant

Utilisation de D-Wave - 2

Résolution directe de problème Ising et QUBO : utilisation des solvers D-Wave.

```
import dimod
import neal
''' 2 QUBO problems '''
Q1 = {('q1', 'q1'): 0.1, ('q2', 'q2'): 0.1, ('q1', 'q2'): -0.2}
Q2 = {('q1', 'q1'): 0.5, ('q2', 'q2'): 0.5, ('q1', 'q2'): -1}
''' 2 types of solvers: exact, and simulated annealing '''
exact_sampler = dimod.ExactSolver()
sa_sampler = neal.SimulatedAnnealingSampler()
''' Solve using exact solver '''
sample_set = exact_sampler.sample_qubo(Q1)
print(sample_set)
''' Solve using simulated annealing'''
sample_set_1 = exact_sampler.sample_qubo(Q2)
print(sample_set_1)
```