Applied QML

Lecture 6: Quantum Optimization

Christophe Pere

2024-02-08

Table des matières

MaxCut

MaxCut & Modèle d'Ising

MaxCut

Classe de problème d'optimisation combinatoire

Algorithmes d'approximation

MaxCut & Modèle d'Ising

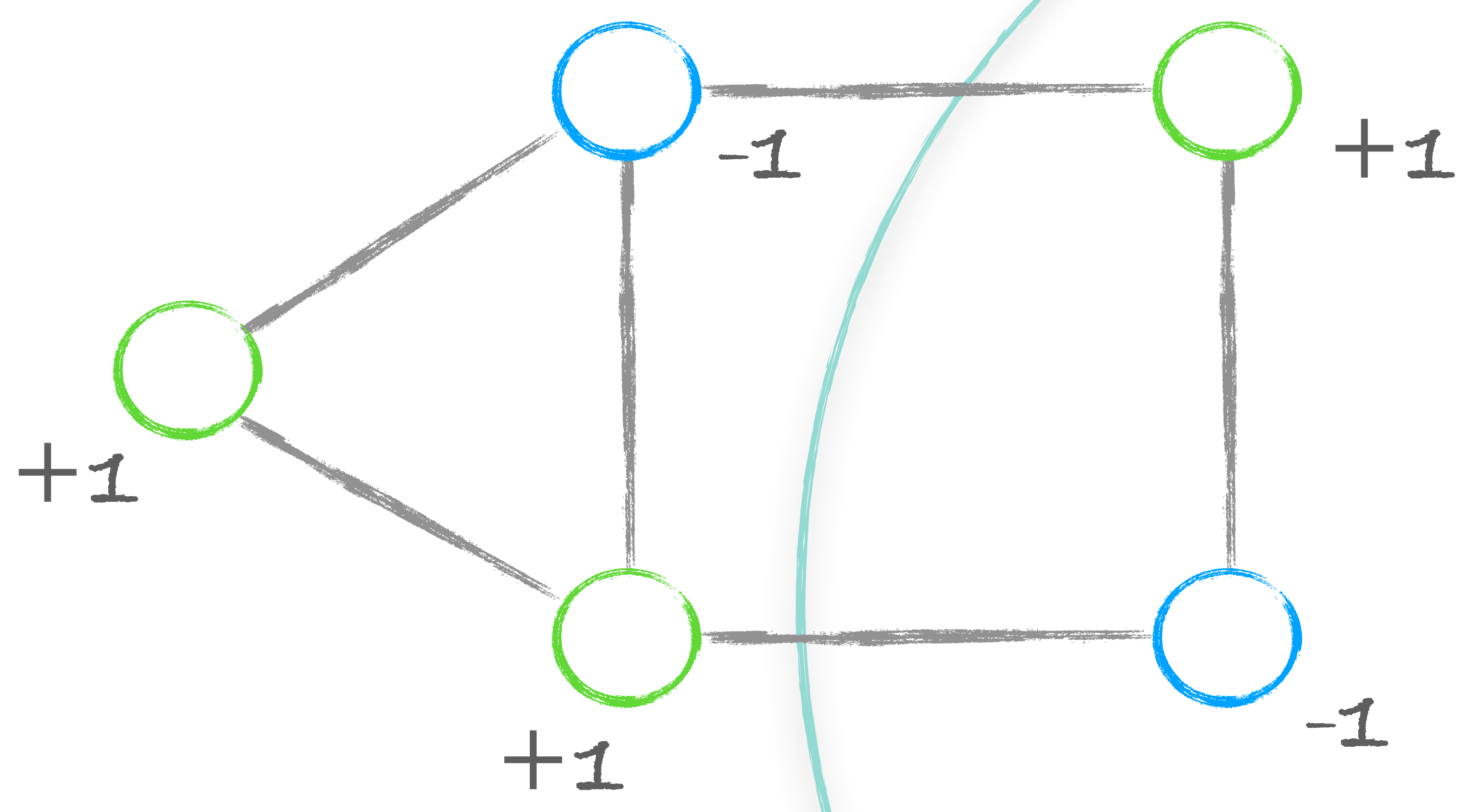
MaxCut

Classe de problème d'optimisation combinatoire

→ Algorithmes d'approximation

- Undirected Graph
- Nodes → Éléments
- Edges → Connexions

  Relations



MaxCut & Modèle d'Ising

MaxCut

Classe de problème d'optimisation combinatoire

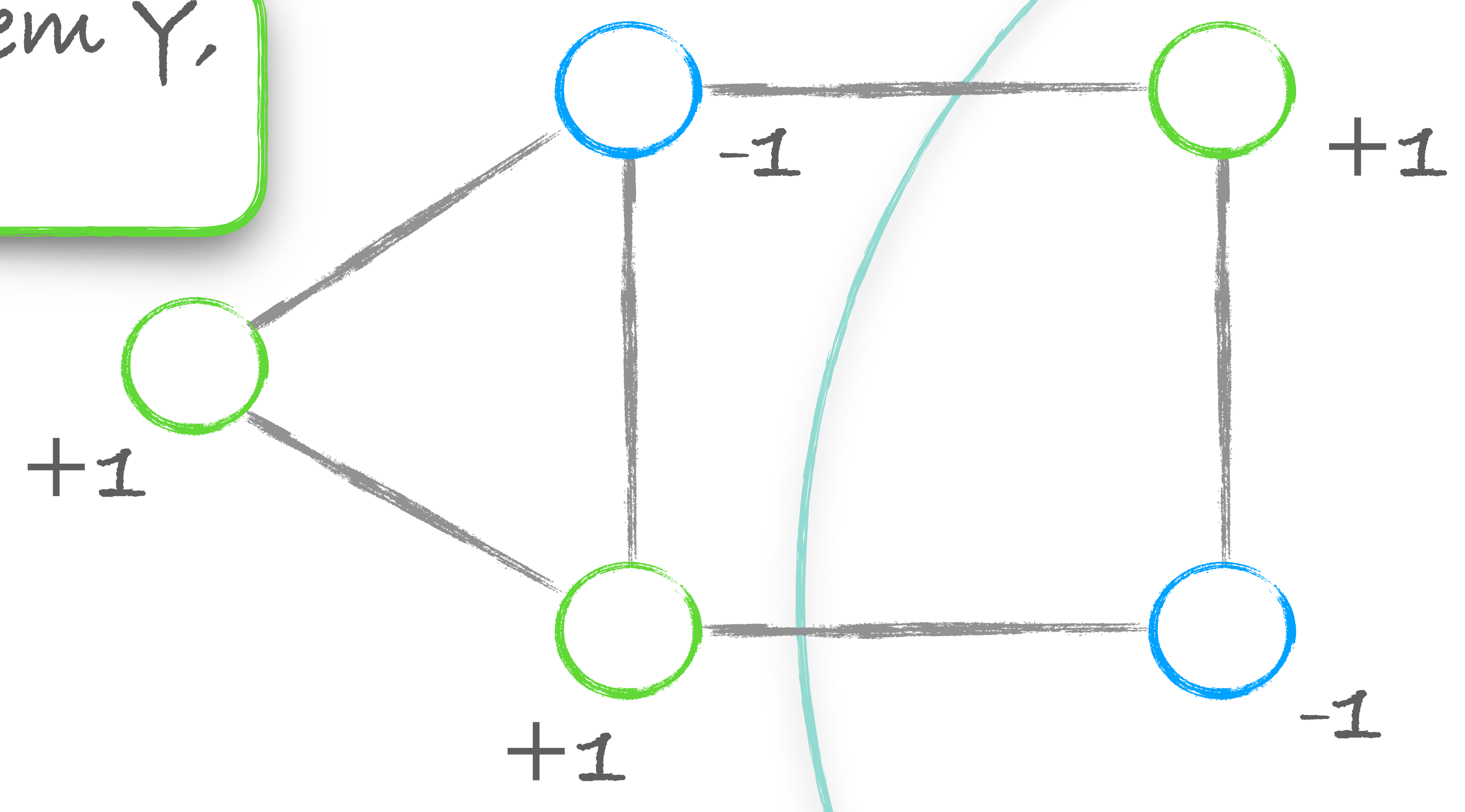
Algorithmes d'approximation

NP-Hard

A problem X is NP-hard, if there is an NP-complete problem Y, such that Y is reducible to X in polynomial time.

Un problème NP-Hard n'a pas besoin d'être dans NP ni d'être un problème de décision

- Undirected Graph
- Nodes → Éléments
- Edges → Connexions / Relations



MaxCut & Modèle d'Ising

MaxCut

Classe de problème d'optimisation combinatoire

Algorithmes d'approximation

- Undirected Graph
- Nodes → Éléments
- Edges → Connexions

  Relations

NP-Hard

A problem X is NP-hard, if there is an NP-complete problem Y, such that Y is reducible to X in polynomial time.

Tout problème NP-complet peut être réduit à tout autre problème NP-complet en temps polynomial, tous les problèmes NP-complets peuvent être réduits à tout problème NP-Hard en temps polynomial. Ainsi, s'il existe une solution à un problème NP-Hard en temps polynomial, il existe une solution à tous les problèmes NP en temps polynomial.

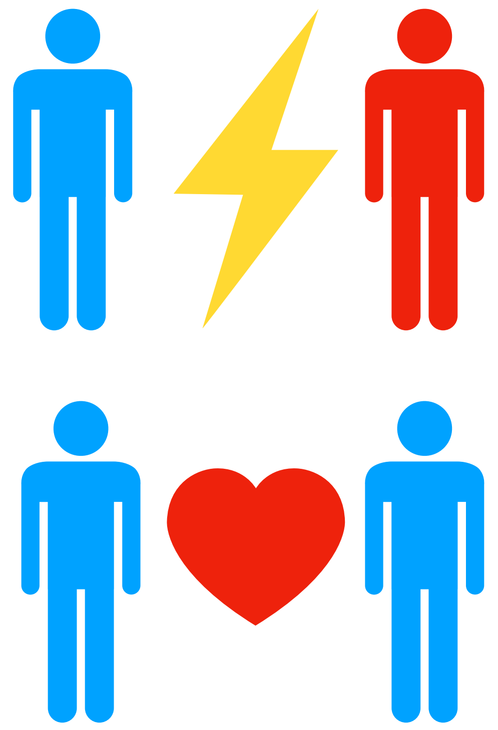
MaxCut & Modèle d'Ising

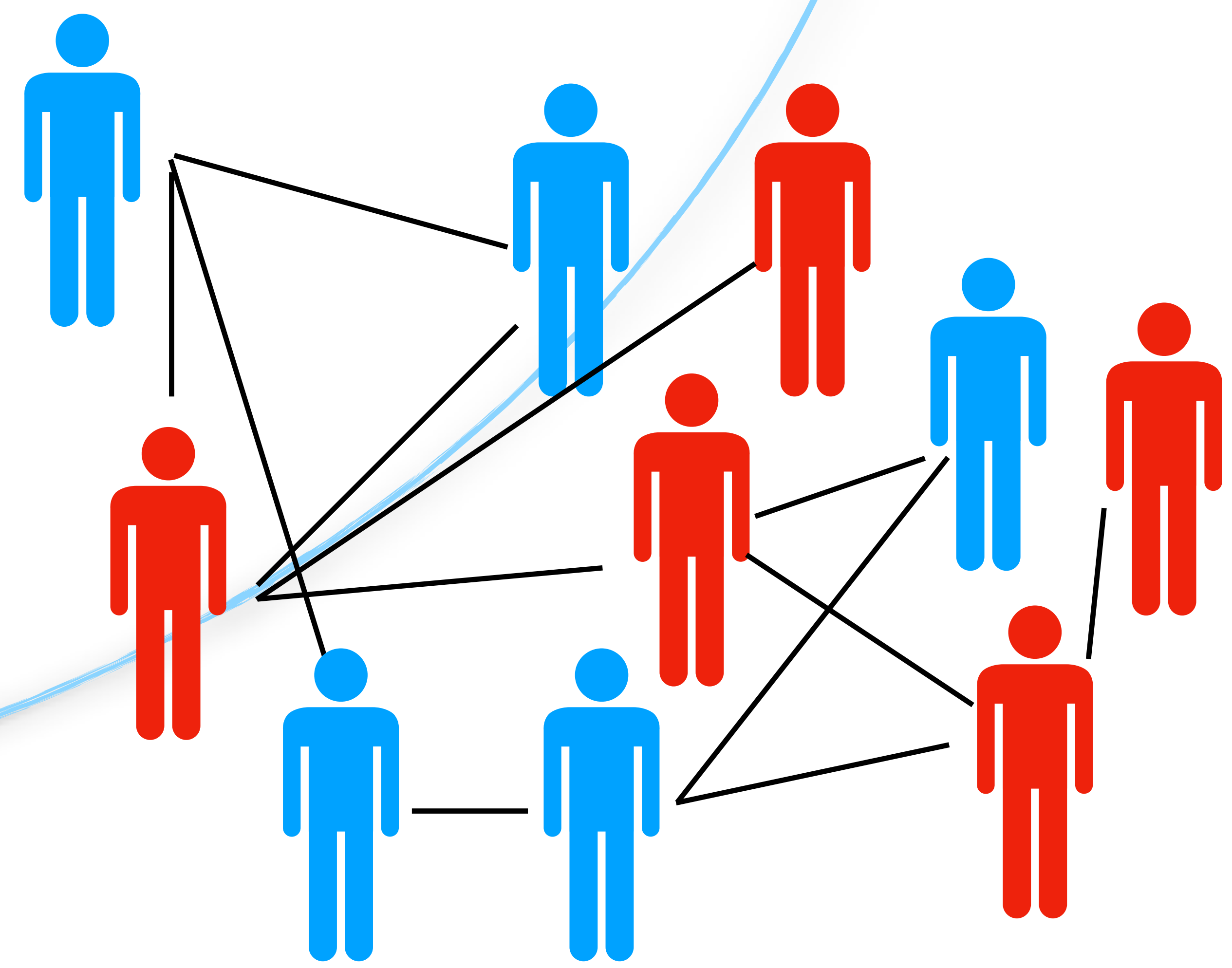
MaxCut

Classe de problème d'optimisation combinatoire

Algorithmes d'approximation

Team





- Undirected Graph
- Nodes → Éléments
- Edges → Connexions

  Relations

MaxCut & Modèle d'Ising

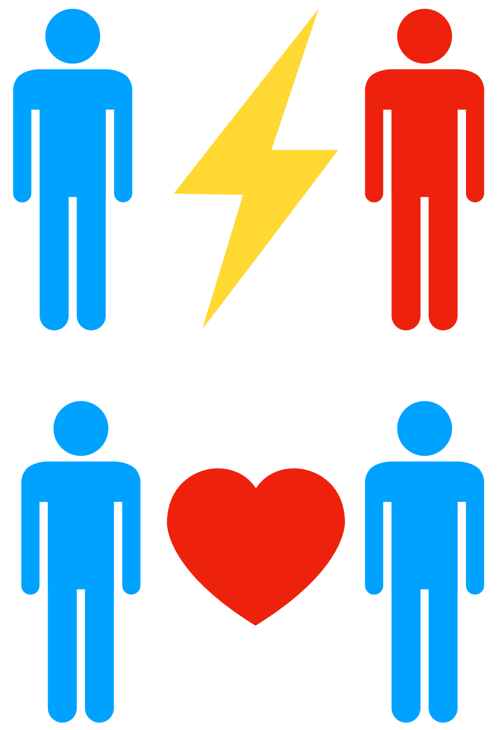
MaxCut

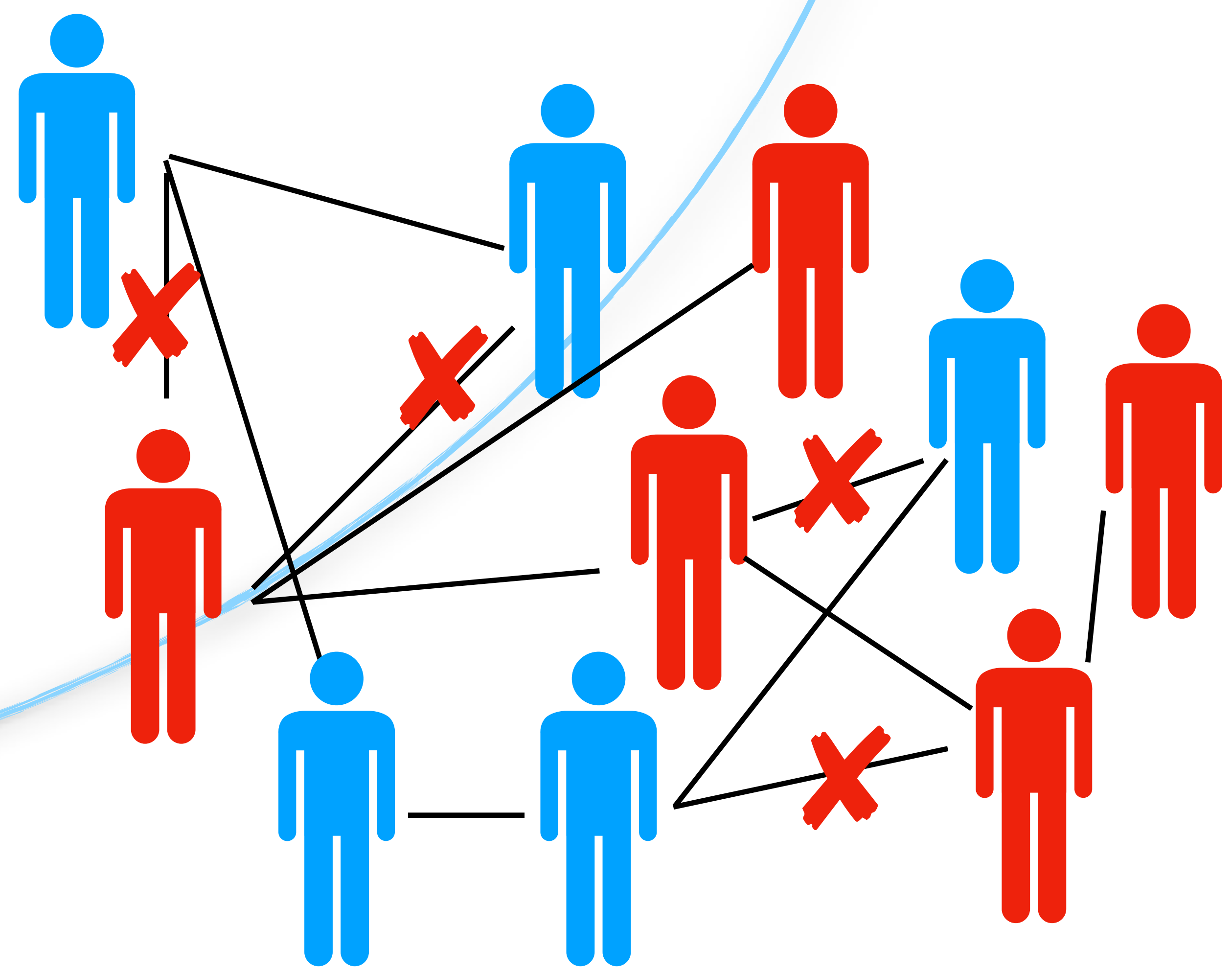
Classe de problème d'optimisation combinatoire

Algorithmes d'approximation

- Undirected Graph
- Nodes → Éléments
- Edges → Connexions

  Relations





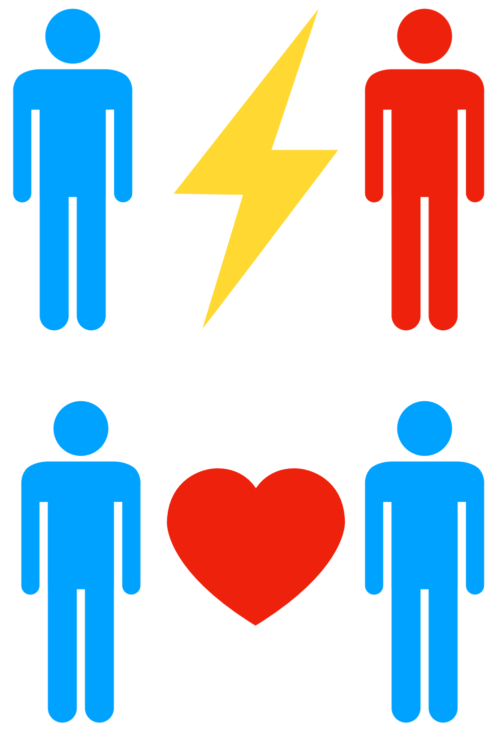
MaxCut & Modèle d'Ising

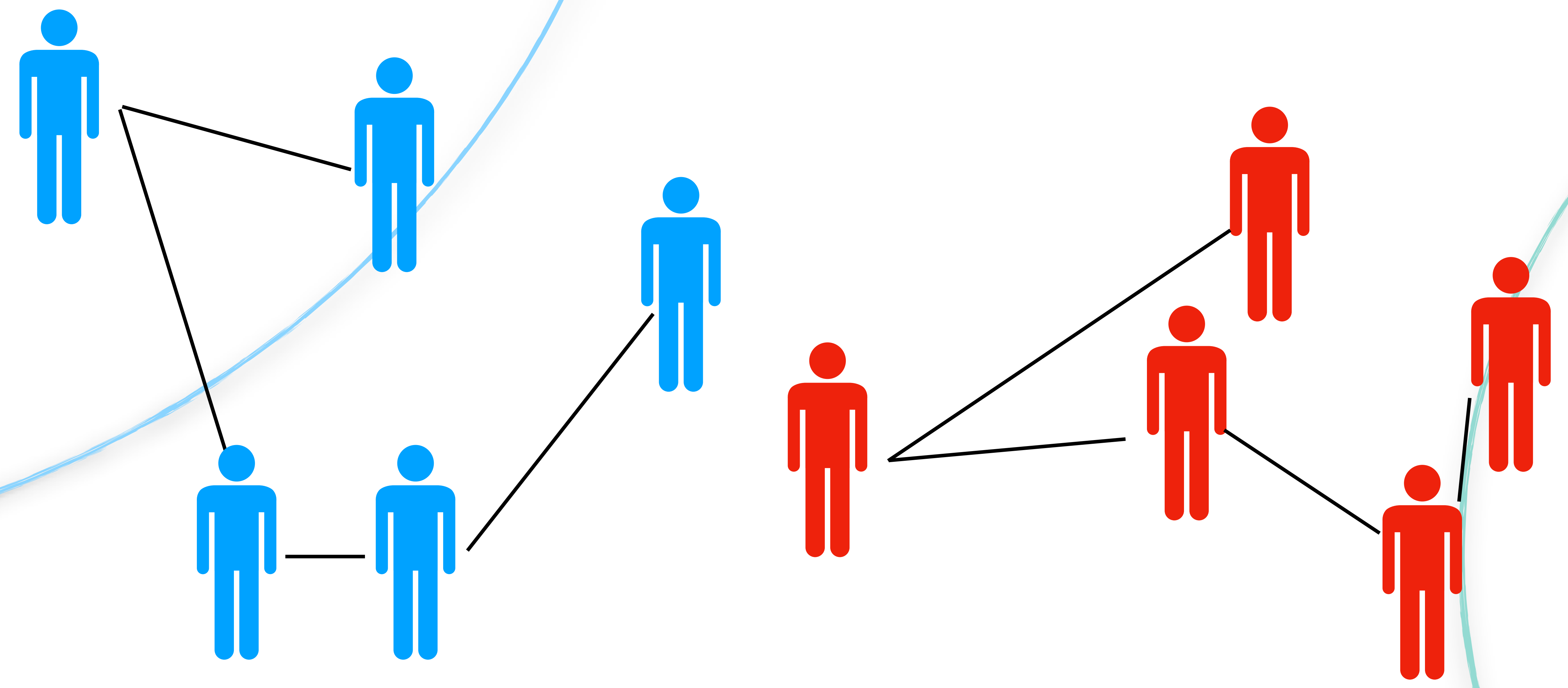
MaxCut

Classe de problème d'optimisation combinatoire

Algorithmes d'approximation

- Undirected Graph
- Nodes → Éléments
- Edges → Connexions
  Relations





Modèle d'Ising

MaxCut & Modèle d'Ising

Ising

Physique

Modèle ferromagnétique qui représente les "spins" des atomes et les interactions avec les atomes voisins. La structure est appelée un lattice.

MaxCut & Modèle d'Ising

Ising

Physique

Modèle ferromagnétique qui représente les "spins" des atomes et les interactions avec les atomes voisins. La structure est appelée un lattice.

Informatique

Modèle de graphe dont les noeuds sont binaires (+1, -1).

MaxCut & Modèle d'Ising

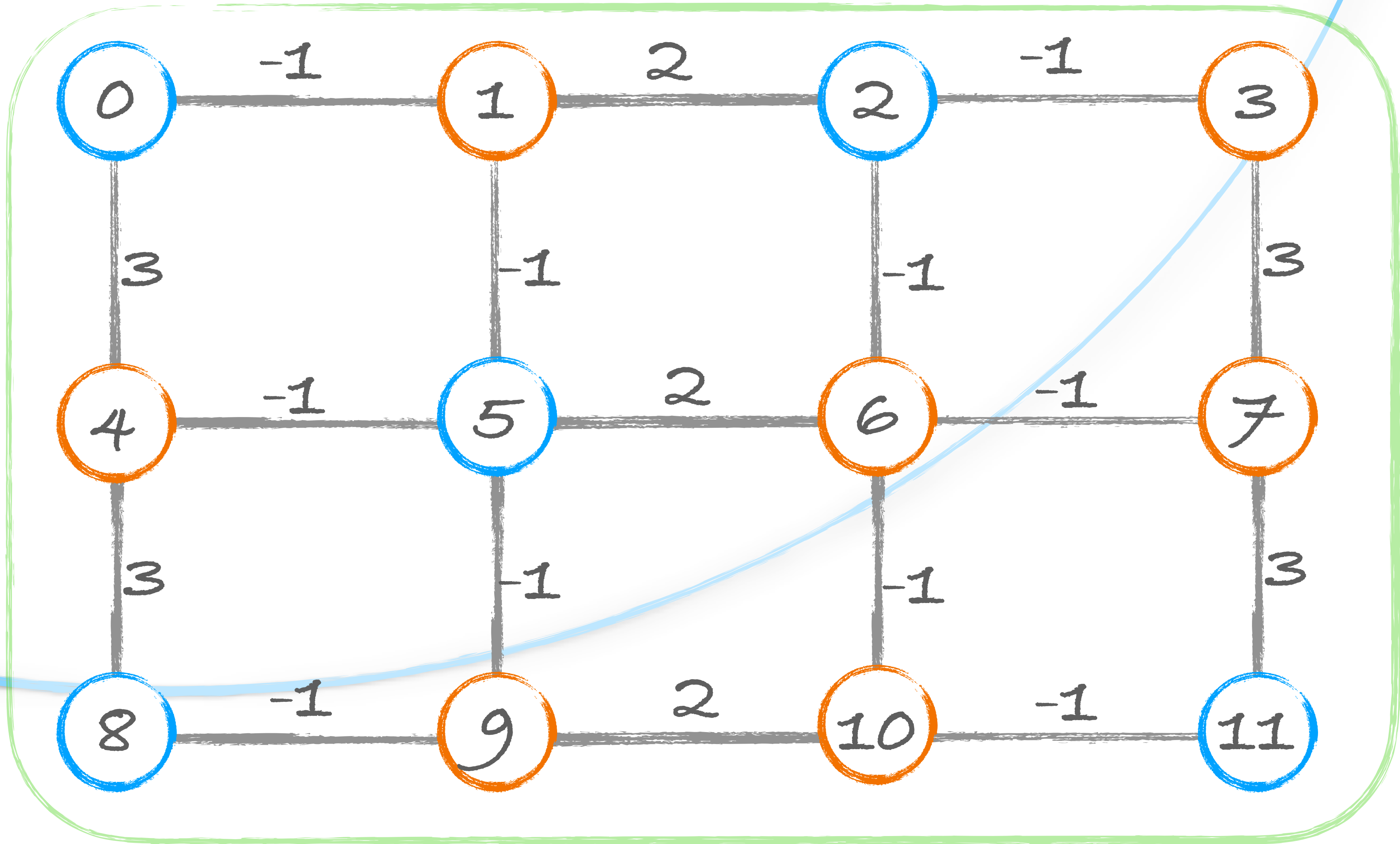
Ising

Physique

Modèle ferromagnétique qui représente les "spins" des atomes et les interactions avec les atomes voisins. La structure est appelée un lattice.

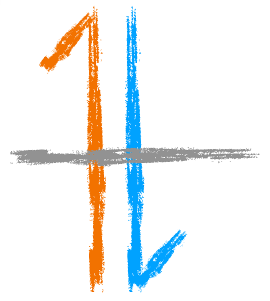
Informatique

Modèle de graphe dont les noeuds sont binaires (+1, -1).



Lattice

Spin up (+1)



Niveau d'énergie

Spin down (-1)

MaxCut & Modèle d'Ising

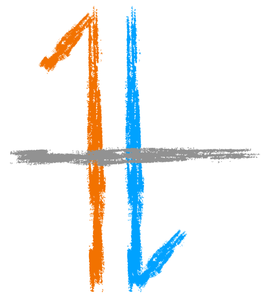
Ising

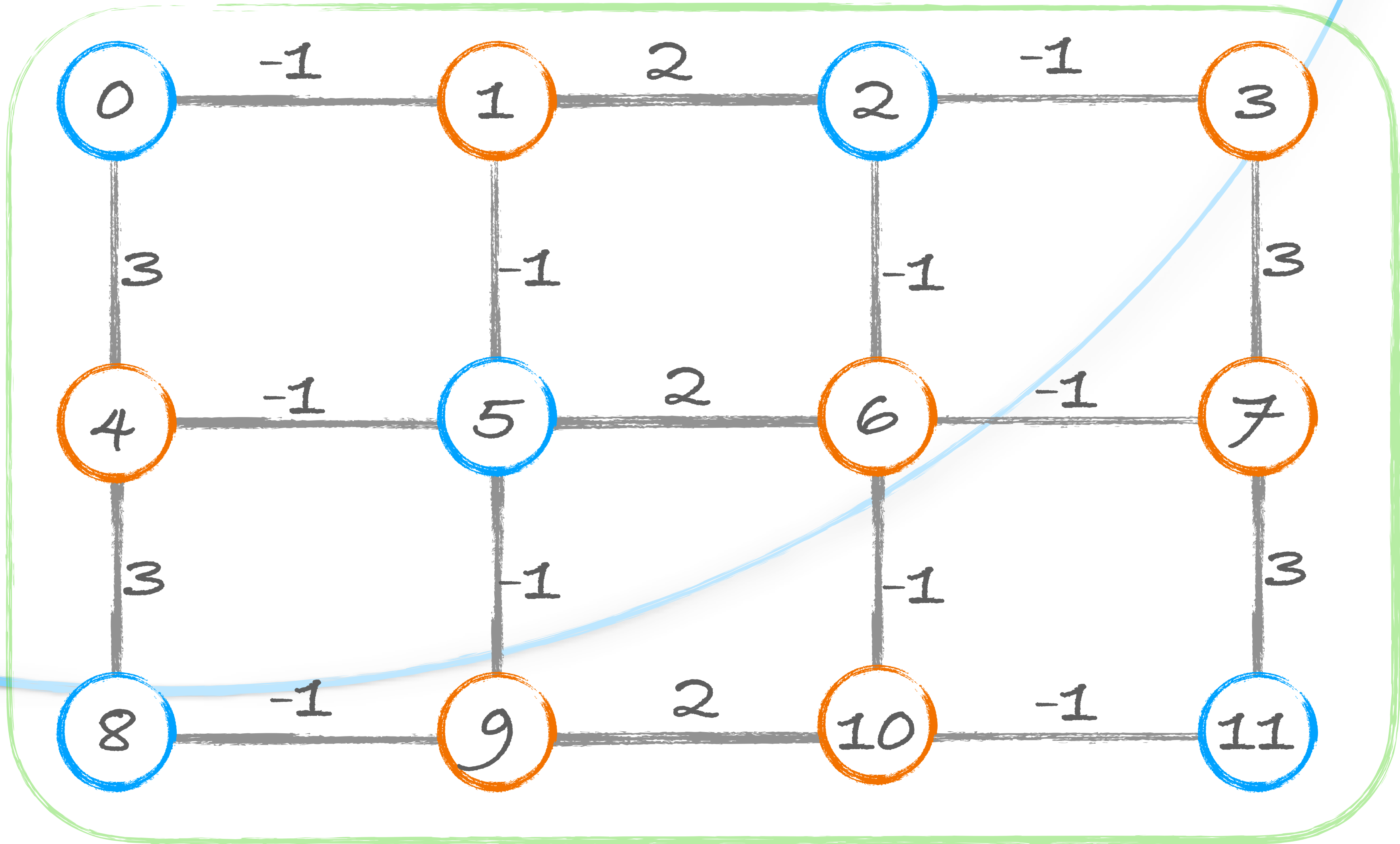
Physique

Modèle ferromagnétique qui représente les "spins" des atomes et les interactions avec les atomes voisins. La structure est appelée un lattice.

Informatique

Modèle de graphe dont les noeuds sont binaires (+1, -1).





Lattice

Hamiltonien

$$H = -\sum_{j,k} J_{jk} z_j z_k - \sum_j h_j z_j$$

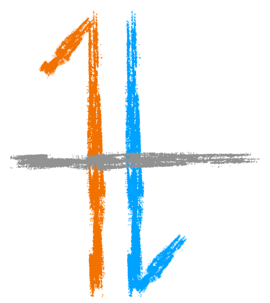
J_{jk} Interaction particule j et k

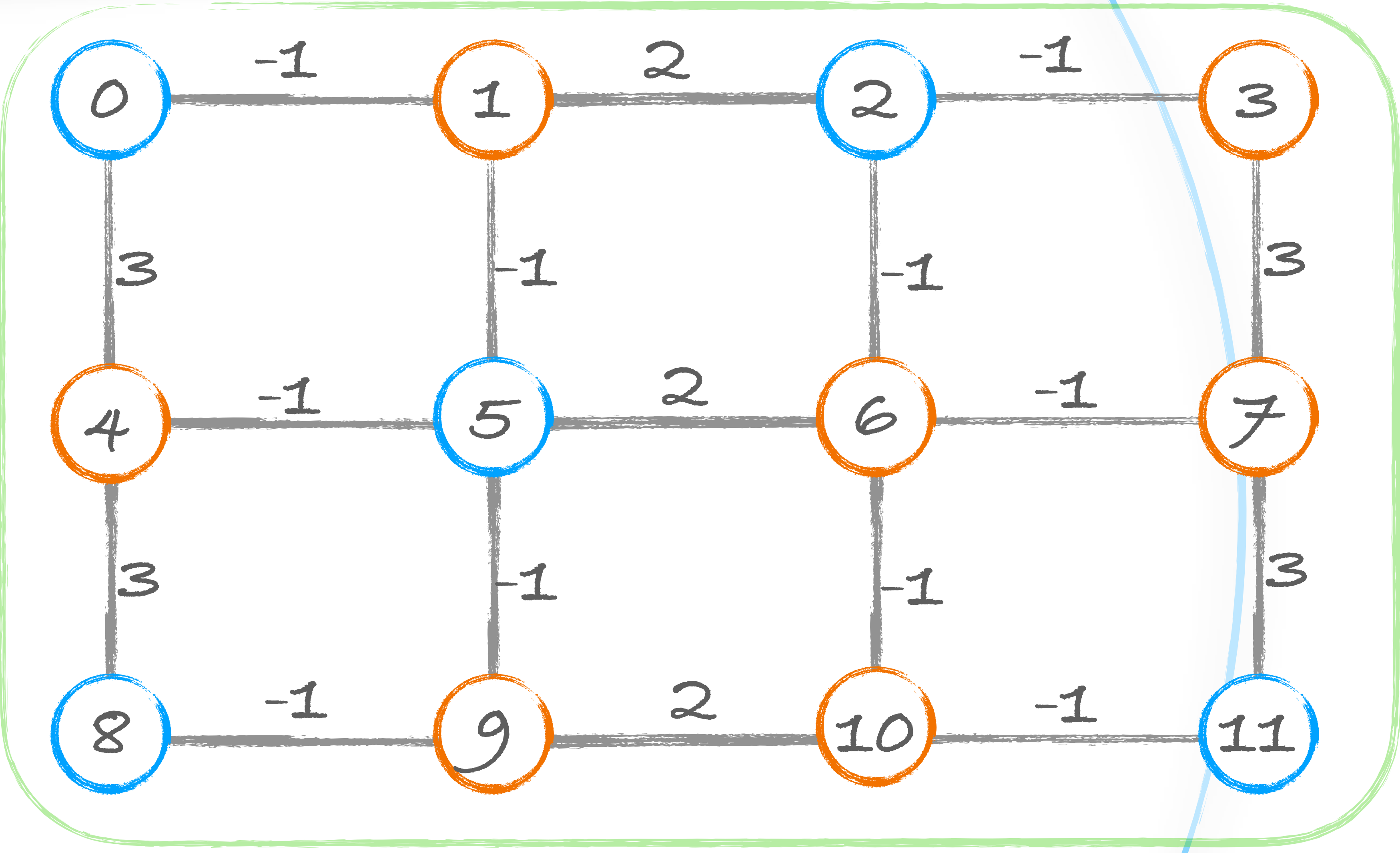
h_j Champ magnétique externe

z Particule

MaxCut & Modèle d'Ising

Ising





Lattice

Hamiltonien

$$H = -\sum_{j,k} J_{jk} z_j z_k - \sum_j h_j z_j$$

J_{jk} Interaction particule j et k

h_j Champ magnétique externe

z Particule

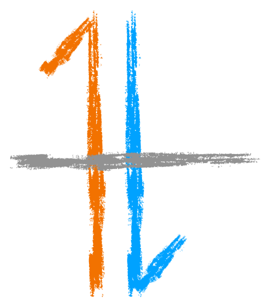
Comment formuler l'hamiltonien avec:

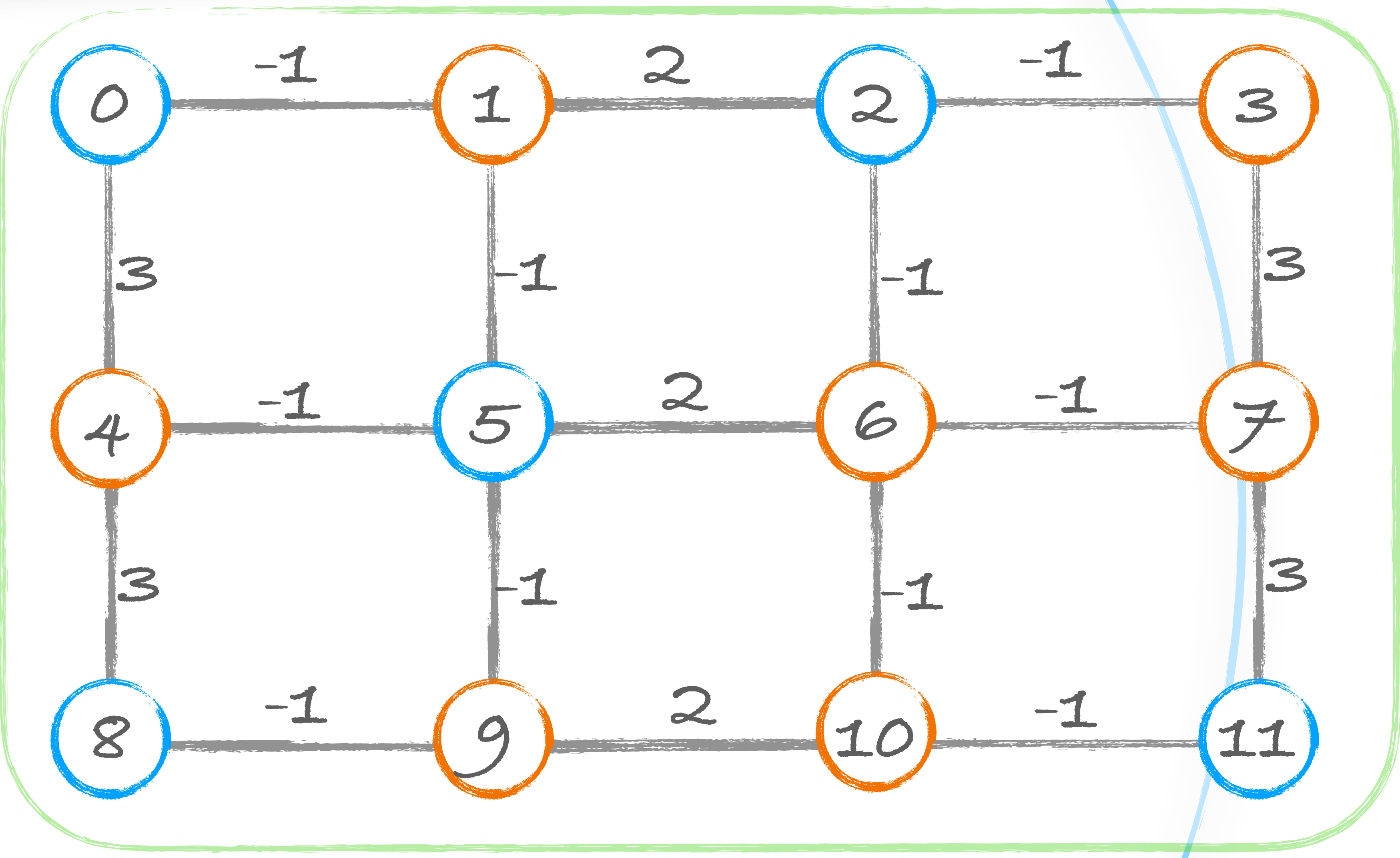
$$J_{jk} = 1$$

$$h_j = 1$$

MaxCut & Modèle d'Ising

Ising





Lattice

Hamiltonien

$$H = -\sum_{j,k} J_{jk} z_j z_k - \sum_j h_j z_j$$

J_{jk} Interaction particule j et k

h_j Champ magnétique externe

z Particule

Comment formuler l'hamiltonien avec:

$$J_{jk} = 1$$

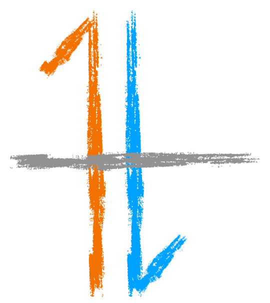
$$h_j = 1$$

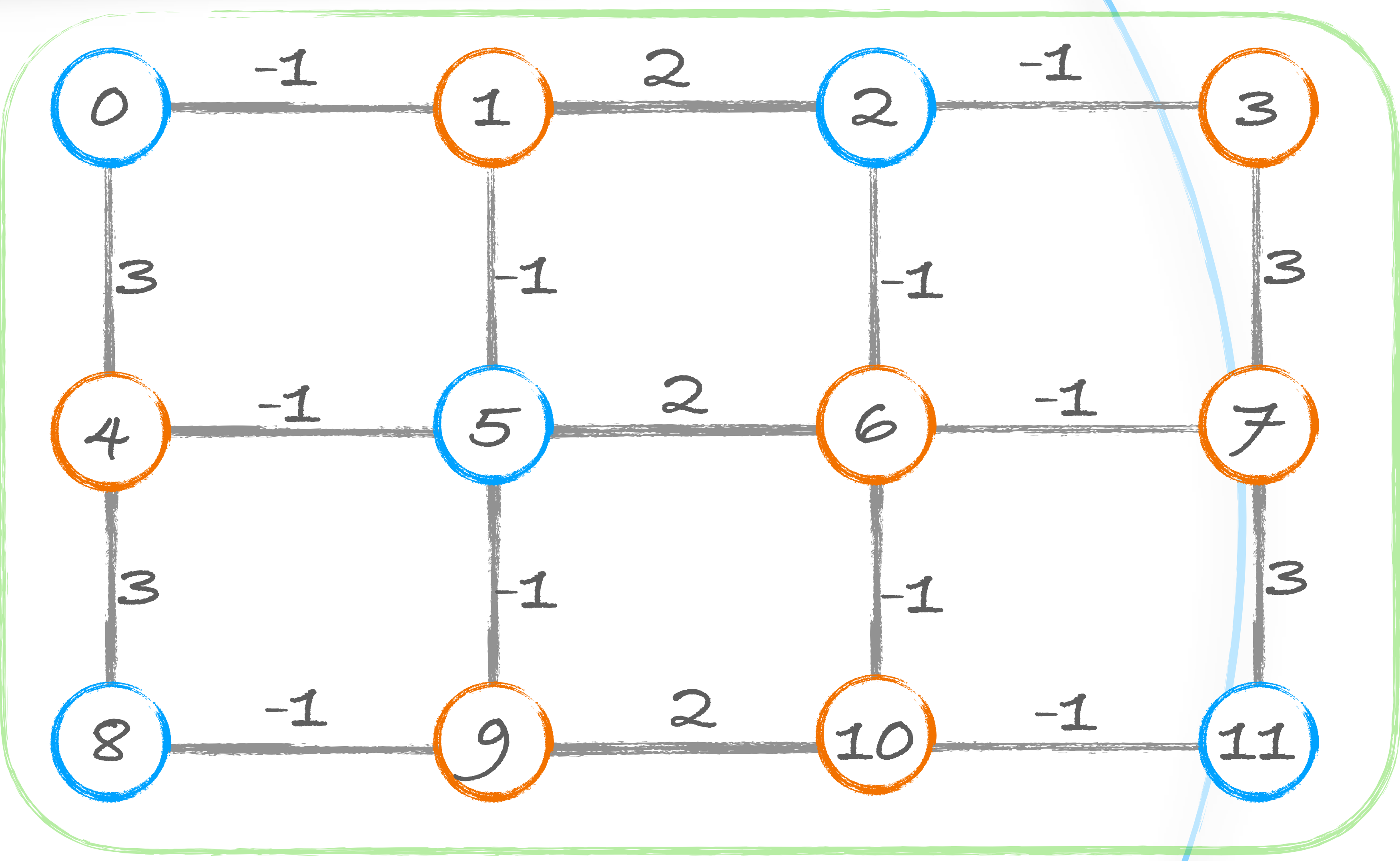
$$-\sum_{j,k} J_{jk} z_j z_k$$

$$z_0z_1 - 2z_1z_2 + z_2z_3 - 3z_0z_4 + z_1z_5 + z_2z_6 - 3z_3z_7$$
$$+z_4z_5 - 2z_5z_6 + z_6z_7 - 3z_4z_8 + z_5z_9 + z_6z_10 - 3z_7z_{11}$$
$$+z_8z_9 - 2z_9z_{10} + z_{10}z_{11}$$

MaxCut & Modèle d'Ising

Ising





Lattice

Hamiltonien

$$H = -\sum_{j,k} J_{jk} z_j z_k - \sum_j h_j z_j$$

J_{jk} Interaction particule j et k

h_j Champ magnétique externe

z Particule

Comment formuler l'hamiltonien avec:

$$J_{jk} = 1$$

$$h_j = 1$$

$$-\sum_{j,k} J_{jk} z_j z_k$$

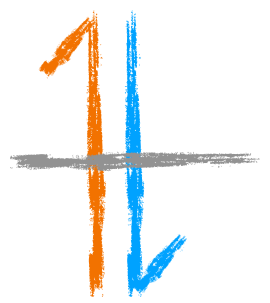
$$z_0z_1 - 2z_1z_2 + z_2z_3 - 3z_0z_4 + z_1z_5 + z_2z_6 - 3z_3z_7$$
$$+z_4z_5 - 2z_5z_6 + z_6z_7 - 3z_4z_8 + z_5z_9 + z_6z_10 - 3z_7z_{11}$$
$$+z_8z_9 - 2z_9z_{10} + z_{10}z_{11}$$

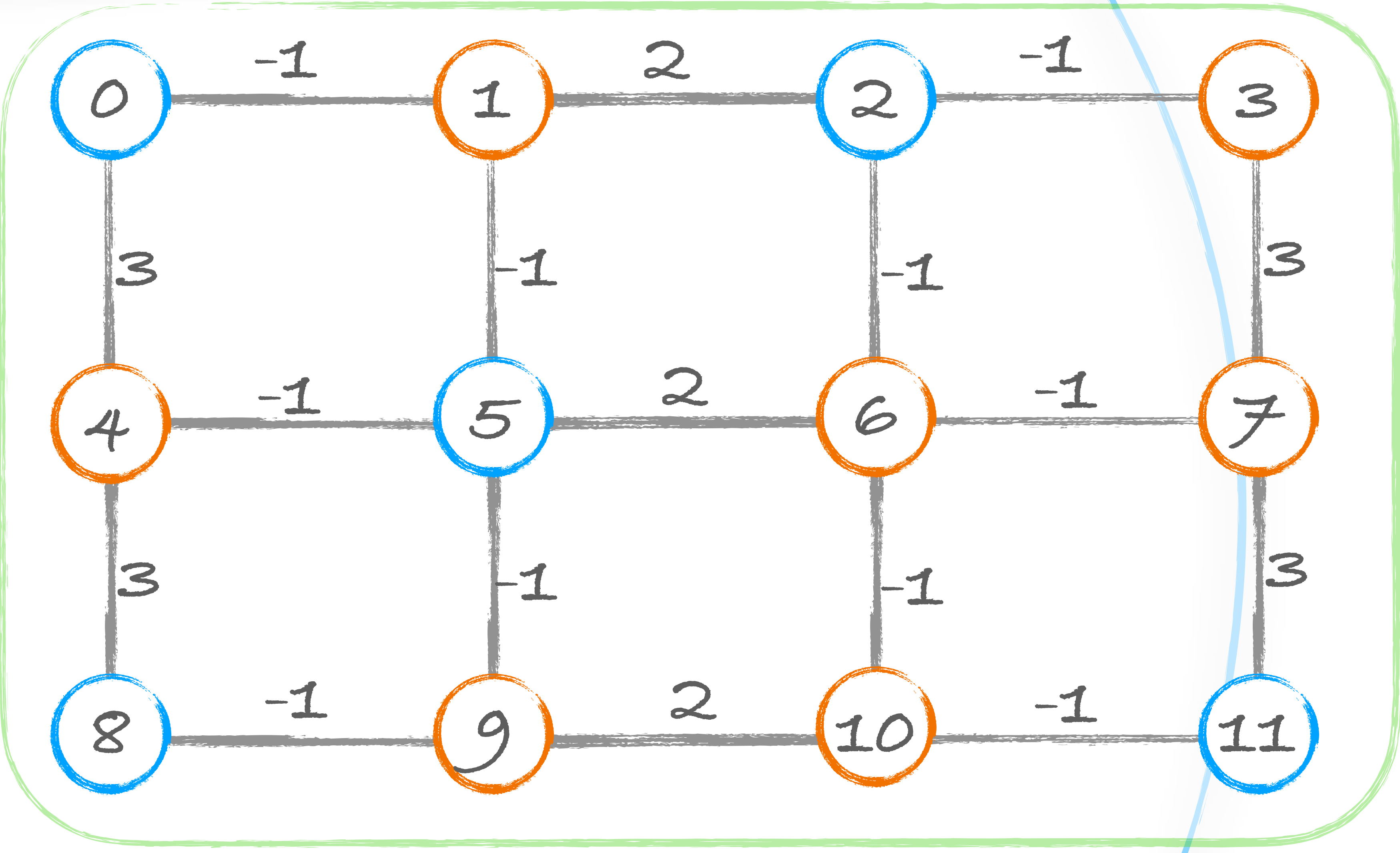
$$-\sum_j h_j z_j$$

$$-z_0 - z_1 - z_2 - z_3 - z_4 - z_5 - z_6 - z_7 - z_8 - z_9$$
$$-z_{10} - z_{11}$$

MaxCut & Modèle d'Ising

Ising





Lattice

Hamiltonien

$$H = -\sum_{j,k} J_{jk} z_j z_k - \sum_j h_j z_j$$

J_{jk} Interaction particule j et k

h_j Champ magnétique externe

z Particule

Comment formuler l'hamiltonien avec:

$$J_{jk} = 1$$

$$h_j = 1$$

Minimiser

$$-\sum_{j,k} J_{jk} z_j z_k$$

$$z_0z_1 - 2z_1z_2 + z_2z_3 - 3z_0z_4 + z_1z_5 + z_2z_6 - 3z_3z_7$$
$$+z_4z_5 - 2z_5z_6 + z_6z_7 - 3z_4z_8 + z_5z_9 + z_6z_10 - 3z_7z_{11}$$
$$+z_8z_9 - 2z_9z_{10} + z_{10}z_{11}$$

$$-\sum_j h_j z_j$$

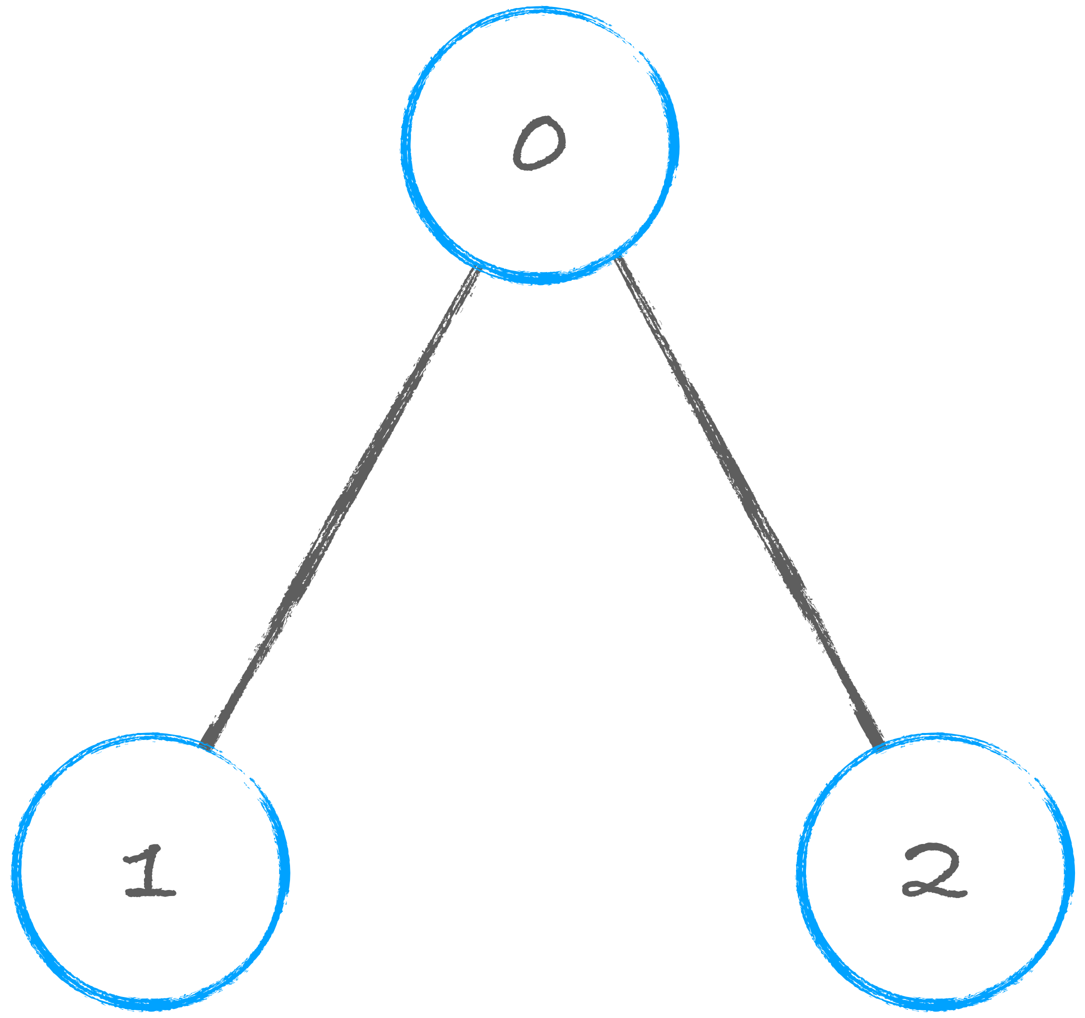
$$-z_0 - z_1 - z_2 - z_3 - z_4 - z_5 - z_6 - z_7 - z_8 - z_9$$
$$-z_{10} - z_{11}$$

$$z_j = \{1, -1\} \quad j = 0,\ldots,11$$

Formuler un problème de la bonne façon

Formuler un problème de la bonne façon

Problème simple



Hamiltonien

$$H = -\sum_{j,k} J_{jk} z_j z_k - \cancel{\sum_j z_j}$$

Minimiser $\longrightarrow$ $\boxed{\langle \psi | H | \psi \rangle}$ Valeur attendue

$$z_0 z_1 + z_1 z_2$$

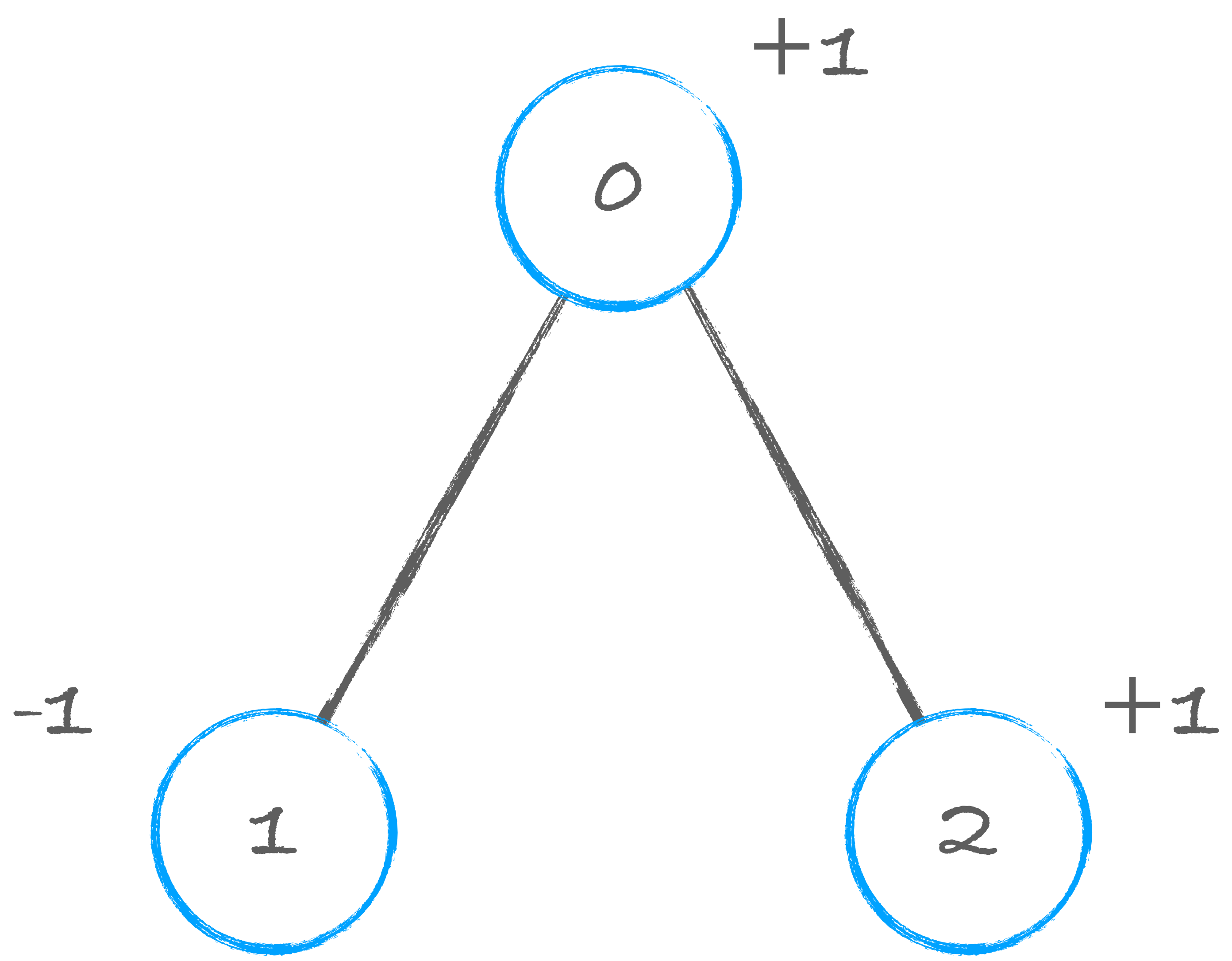
"Ground state"

Pour

$\vert\psi\rangle = \vert 0\rangle$ $H = Z$	$\vert\psi\rangle = \vert 1\rangle$ $H = Z$
$\langle 0 \vert Z \vert 0 \rangle = (1 \quad 0)\begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}\begin{pmatrix} 1 \\ 0 \end{pmatrix} = 1$	$\langle 1 \vert Z \vert 1 \rangle = (0 \quad 1)\begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}\begin{pmatrix} 0 \\ 1 \end{pmatrix} = -1$

Formuler un problème de la bonne façon

Problème simple



Prenons l'état

$$|\psi\rangle = |010\rangle$$

0 pour représenter +1
1 pour représenter -1

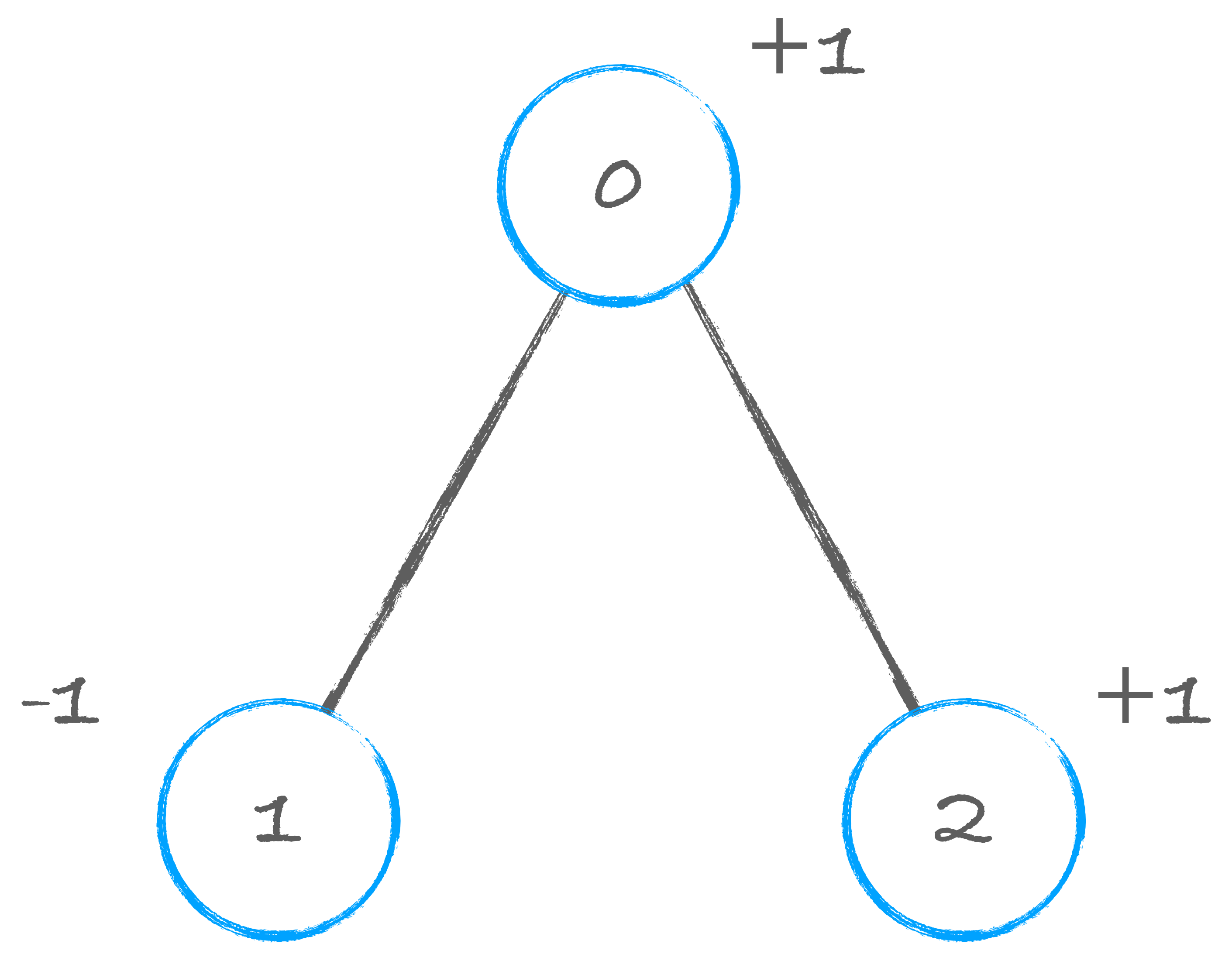
$$H = ZZI$$

$$Z \otimes Z \otimes I = Z_0 Z_1$$

Z pour évaluer 0
Z pour évaluer 1
I pour évaluer 2

Formuler un problème de la bonne façon

Problème simple



Prenons l'état

$$|\psi\rangle = |010\rangle$$

0 pour représenter +1

1 pour représenter -1

$$H = ZZI$$

$$Z \otimes Z \otimes I = Z_0 Z_1$$

Z pour évaluer 0

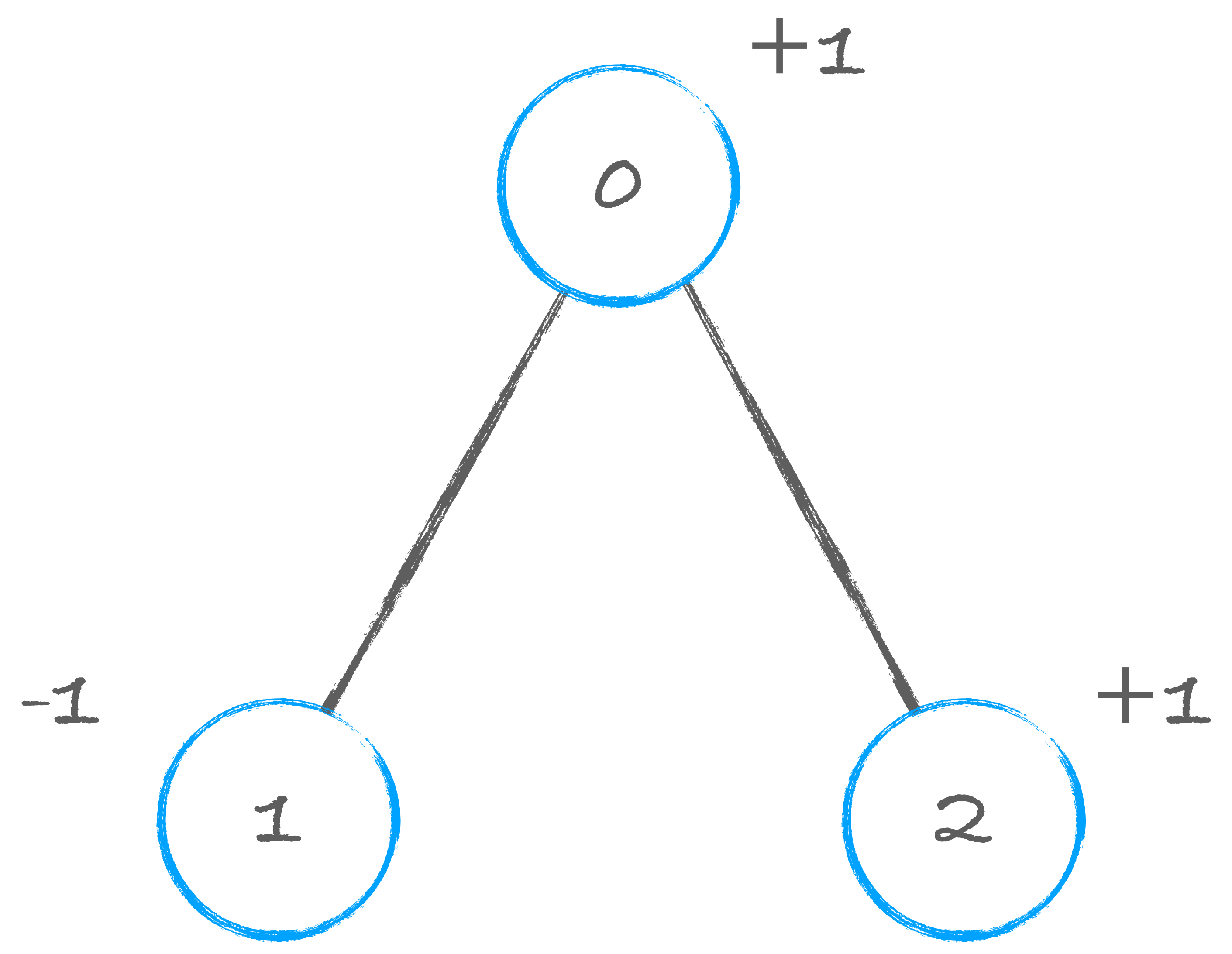
Z pour évaluer 1

I pour évaluer 2

$$\begin{aligned} \langle 010 | Z \otimes Z \otimes I | 010 \rangle &= \langle 010 | \left(Z|0\rangle \otimes Z|1\rangle \otimes I|0\rangle \right) \\ &= \langle 0|Z|0\rangle \langle 1|Z|1\rangle \langle 0|I|0\rangle \\ &= 1 \cdot (-1) \cdot 1 = -1 \end{aligned}$$

Formuler un problème de la bonne façon

Problème simple



Prenons l'état

$$|\psi\rangle = |010\rangle$$

0 pour représenter +1
1 pour représenter -1

$$H = ZZI$$

$$Z \otimes Z \otimes I = Z_0 Z_1$$

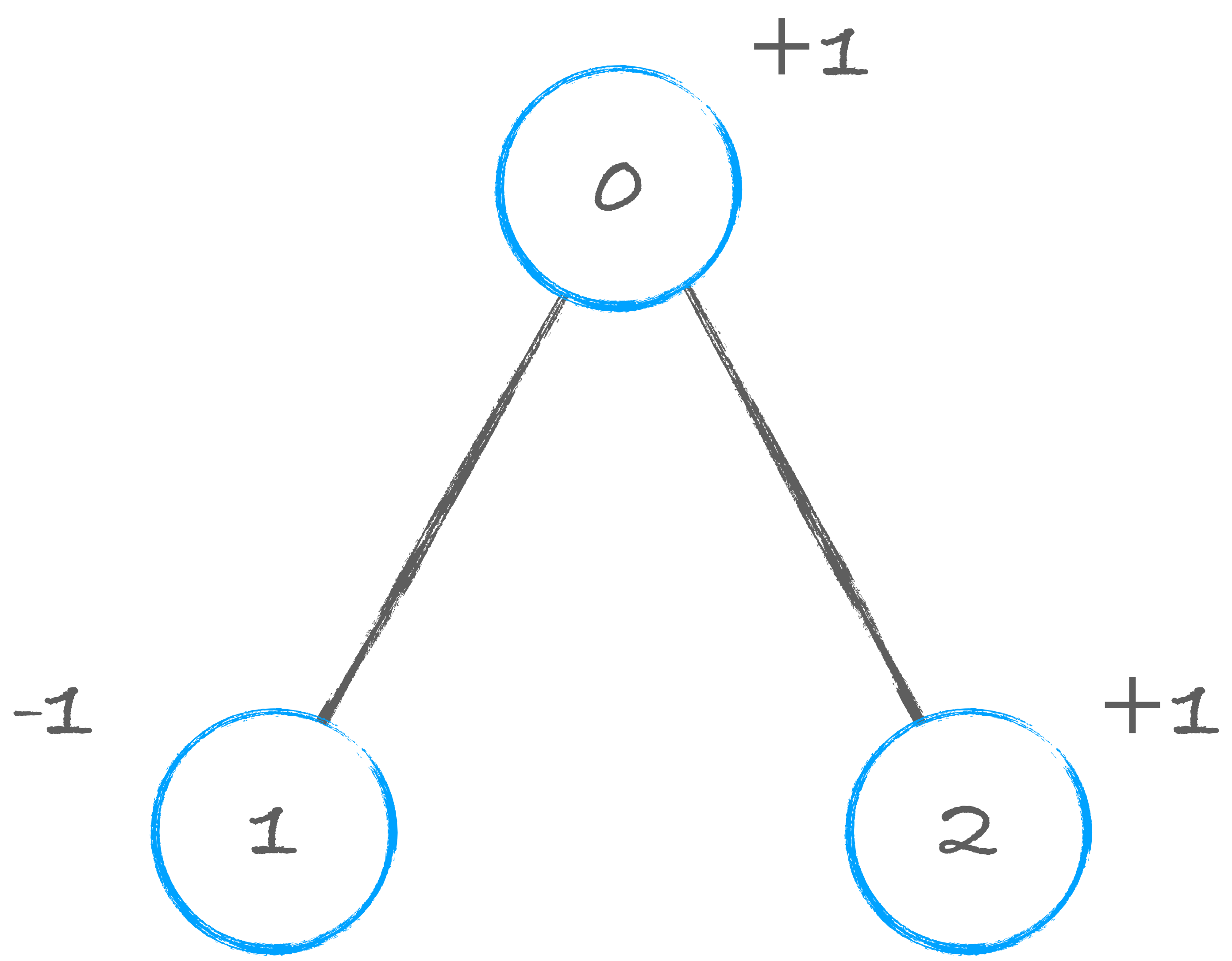
Z pour évaluer 0
Z pour évaluer 1
I pour évaluer 2

$$\begin{aligned}\langle 010|Z \otimes Z \otimes I|010\rangle &= \langle 010|\big(Z|0\rangle \otimes Z|1\rangle \otimes I|0\rangle\big) \\ &= \langle 0|Z|0\rangle\langle 1|Z|1\rangle\langle 0|I|0\rangle \\ &= 1 \cdot (-1) \cdot 1 = -1\end{aligned}$$

→ Comme 0 et 1 sont différents l'évaluation donne -1 et devient un "cut"

Formuler un problème de la bonne façon

Problème simple



Prenons l'état

$$|\psi\rangle = |010\rangle$$

0 pour représenter +1

1 pour représenter -1

$$H = ZZI$$

$$Z \otimes Z \otimes I = Z_0 Z_1$$

Z pour évaluer 0

Z pour évaluer 1

I pour évaluer 2

$$\begin{aligned}\langle 010|Z\otimes Z\otimes I|010\rangle &= \langle 010|\big(Z|0\rangle\otimes Z|1\rangle\otimes I|0\rangle\big)\\ &= \langle 0|Z|0\rangle\langle 1|Z|1\rangle\langle 0|I|0\rangle\\ &= 1\cdot(-1)\cdot 1 = -1\end{aligned}$$

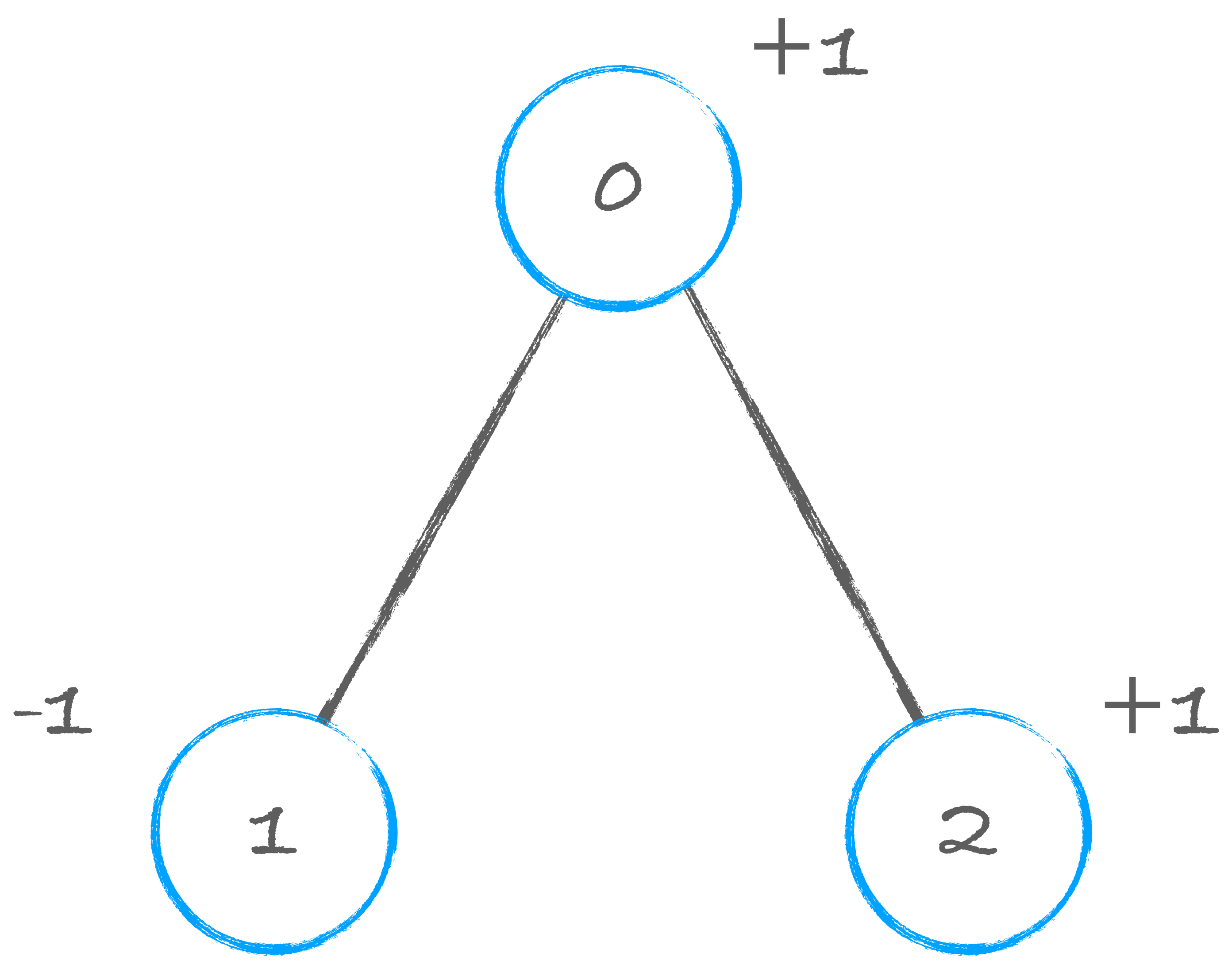
Comme 0 et 1 sont différents l'évaluation donne -1 et devient un "cut"

Pour Z_0Z_2

$$\begin{aligned}\langle 010|Z\otimes I\otimes Z|010\rangle &= \langle 010|\big(Z|0\rangle\otimes I|1\rangle\otimes Z|0\rangle\big)\\ &= \langle 0|Z|0\rangle\langle 1|I|1\rangle\langle 0|Z|0\rangle\\ &= 1\cdot 1\cdot 1 = 1\end{aligned}$$

Formuler un problème de la bonne façon

Problème simple



Prenons l'état

$$|\psi\rangle = |010\rangle$$

0 pour représenter +1

1 pour représenter -1

$$H = ZZI$$

$$Z \otimes Z \otimes I = Z_0 Z_1$$

Z pour évaluer 0

Z pour évaluer 1

I pour évaluer 2

$$\begin{aligned}\langle 010 | Z \otimes Z \otimes I | 010 \rangle &= \langle 010 | \big(Z|0\rangle \otimes Z|1\rangle \otimes I|0\rangle \big) \\ &= \langle 0|Z|0\rangle\langle 1|Z|1\rangle\langle 0|I|0\rangle \\ &= 1 \cdot (-1) \cdot 1 = -1\end{aligned}$$

Comme 0 et 1 sont différents l'évaluation donne -1 et devient un "cut"

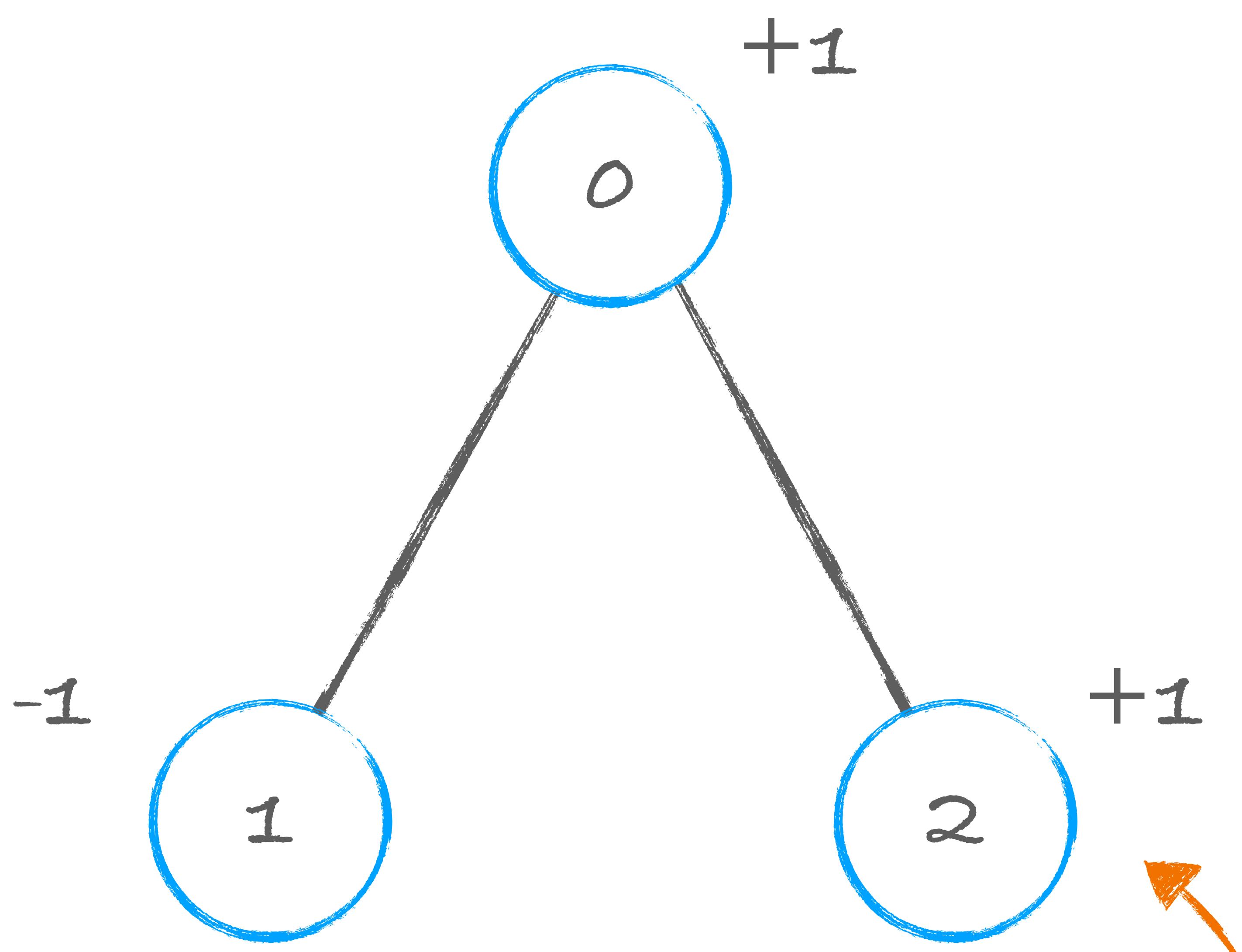
Pour $Z_0 Z_2$

$$\begin{aligned}\langle 010 | Z \otimes I \otimes Z | 010 \rangle &= \langle 010 | \big(Z|0\rangle \otimes I|1\rangle \otimes Z|0\rangle \big) \\ &= \langle 0|Z|0\rangle\langle 1|I|1\rangle\langle 0|Z|0\rangle \\ &= 1 \cdot 1 \cdot 1 = 1\end{aligned}$$

Comme 0 et 2 sont identiques l'évaluation donne 1 et ne devient pas un "cut"

Formuler un problème de la bonne façon

Problème simple



Exemple

Minimiser $\sum_{(j,k)\in E} \langle\psi|Z_j Z_k|\psi\rangle = \langle\psi|Z_0 Z_1 + Z_0 Z_2|\psi\rangle$

n qubits
min

Prenons l'état

$$|\psi\rangle = |010\rangle$$

0 pour représenter +1
1 pour représenter -1

$$H = ZZI$$

$$Z \otimes Z \otimes I = Z_0 Z_1$$

Z pour évaluer 0
Z pour évaluer 1
I pour évaluer 2

$$Z_j Z_k |\psi\rangle$$

Eigenvector

Pour des eigenvalues de 1 et -1

L'objectif est de trouver la valeur minimum pour maximiser les "cuts"

Cela reviens à dire trouver le bitstring dont l'entier est le plus bas

Formuler un problème de la bonne façon

Problème simple

Ising

Rappel $H = -\sum_{j,k} J_{jk} z_j z_k - \sum_j h_j z_j$

Hamiltonien
ou
Self-adjoint matrix

Minimiser $-\sum_{(j,k)\in E} J_{jk}\langle\psi|Z_j Z_k|\psi\rangle - \sum_j h_j\langle\psi|Z_j|\psi\rangle$

n qubits
min

Les problèmes combinatoires peuvent être réécrits comme des problèmes de détermination du "ground state"

D'un Ising à un QUBO

D'un Ising à un QUBO

Subset sum pb

1 set de données
1 cible (target)

Est-ce que la somme de plusieurs éléments peut être égale a la cible?

NP-Complet

D'un Ising à un QUBO

Subset sum pb

1 set de données
1 cible (target)

Est-ce que la somme de plusieurs éléments peut être égale a la cible?

Exemple 1:

$S = \{1, 3, 4, 7, -4\}$
$T = 6$

NP-Complet

D'un Ising à un QUBO

Subset sum pb

1 set de données
1 cible (target)

Est-ce que la somme de plusieurs éléments peut être égale a la cible?

Exemple 1:

$S = \{1, 3, 4, 7, -4\}$
$T = 6$

$S' = \{3, 7, -4\}$
$3 + 7 - 4 = 6 = T$

Il y a une solution

NP-Complet

D'un Ising à un QUBO

Subset sum pb

1 set de données
1 cible (target)

Est-ce que la somme de plusieurs éléments peut être égale a la cible?

Exemple 1:

$S = \{1, 3, 4, 7, -4\}$
$T = 6$

$S' = \{3, 7, -4\}$
$3 + 7 - 4 = 6 = T$

Il y a une solution

Exemple 2:

$S = \{2, -2, 4, 8, -12\}$
$T = 1$

NP-Complet

D'un Ising à un QUBO

Subset sum pb

1 set de données
1 cible (target)

Est-ce que la somme de plusieurs éléments peut être égale a la cible?

Exemple 1:

$S = \{1, 3, 4, 7, -4\}$
$T = 6$

$S' = \{3, 7, -4\}$
$3 + 7 - 4 = 6 = T$

Il y a une solution

Exemple 2:

$S = \{2, -2, 4, 8, -12\}$
$T = 1$

Il n'y a une solution
Tous les éléments de S sont pairs, la target est impaire

NP-Complet

D'un Ising à un QUBO

Subset sum pb

Réécriture selon un Ising ground state

Optimisation combinatoire

1 set de données
1 cible (target)

Est-ce que la somme de plusieurs éléments peut être égale a la cible?

Problème de minimisation avec des variables binaires

$S = \{a_0, \ldots, a_m\}$

Variables binaires $x_j \quad j = 0, \ldots, m$

$$c(x_0, x_1, \ldots, x_m) = \underbrace{(a_0 x_0 + a_1 x_1 + \ldots + a_m x_m - T)^2}_{\text{Équation quadratique}}$$

D'un Ising à un QUBO

Subset sum pb

Réécriture selon un Ising ground state

Optimisation combinatoire

1 set de données
1 cible (target)

Est-ce que la somme de plusieurs éléments peut être égale a la cible?

Problème de minimisation avec des variables binaires

$S = \{a_0, \ldots, a_m\}$

Variables binaires $x_j \quad j = 0, \ldots, m$

$$c(x_0, x_1, \ldots, x_m) = \underbrace{(a_0x_0 + a_1x_1 + \ldots + a_mx_m - T)^2}_{\text{Équation quadratique}}$$

Pour obtenir une valeur positive → $c(x_0, x_1, \ldots, x_m) = 0$ → Toujours non-négatif

$x_0, x_1, \ldots, x_m$

Chaine binaire qui permet
de sélectionner les valeurs du set
quand 1 est présent dans x_j

D'un Ising à un QUBO

Subset sum pb

Réécriture selon un Ising ground state

Optimisation combinatoire

1 set de données
1 cible (target)

Est-ce que la somme de plusieurs éléments peut être égale a la cible?

Problème de minimisation avec des variables binaires

$S = \{a_0, \ldots, a_m\}$

Variables binaires $x_j \quad j = 0, \ldots, m$

$$c(x_0, x_1, \ldots, x_m) = \underbrace{(a_0 x_0 + a_1 x_1 + \ldots + a_m x_m - T)^2}_{\text{Équation quadratique}}$$

Le problème revient à minimiser $c(x_0, x_1, \ldots, x_m)$

Si la minimisation est égale à 0 il y une solution sinon il n'y a pas de solution

D'un Ising à un QUBO

Subset sum pb

Réécriture selon un Ising ground state

$$c(x_0, x_1, \ldots, x_m) = (a_0x_0 + a_1x_1 + \ldots + a_mx_m - T)^2$$

Exemple:

$$S = \{1,4,-2\}$$
$$T = 2$$

D'un Ising à un QUBO

Subset sum pb

Réécriture selon un Ising ground state

$$c(x_0, x_1, \ldots, x_m) = (a_0x_0 + a_1x_1 + \ldots + a_mx_m - T)^2$$

Exemple:

$S = \{1,4,-2\}$
$T = 2$
$\longrightarrow c(x_0, x_1, x_2) = (x_0 + 4x_1 - 2x_2 - 2)^2$

D'un Ising à un QUBO

Subset sum pb

Réécriture selon un Ising ground state

$$c(x_0, x_1, \ldots, x_m) = (a_0x_0 + a_1x_1 + \ldots + a_mx_m - T)^2$$

Exemple:

$S = \{1,4,-2\}$
$T = 2$
$\longrightarrow$ $c(x_0, x_1, x_2) = (x_0 + 4x_1 - 2x_2 - 2)^2$

Revient à minimiser $x_0^2 + 8x_0x_1 - 4x_0x_2 - 4x_0 + 16x_1^2 - 16x_1x_2 - 16x_1 + 4x_2^2 + 8x_2 + 4$

$$x_j \in \{0,1\}, \quad j = 0,\ldots,m$$

D'un Ising à un QUBO

Subset sum pb

Réécriture selon un Ising ground state

$$c(x_0, x_1, \ldots, x_m) = (a_0x_0 + a_1x_1 + \ldots + a_mx_m - T)^2$$

Exemple:

$$\begin{aligned} S &= \{1,4,-2\} \\ T &= 2 \end{aligned} \longrightarrow c(x_0, x_1, x_2) = (x_0 + 4x_1 - 2x_2 - 2)^2$$

Revient à minimiser $x_0^2 + 8x_0x_1 - 4x_0x_2 - 4x_0 + 16x_1^2 - 16x_1x_2 - 16x_1 + 4x_2^2 + 8x_2 + 4$

$x_j \in \{0,1\}, \quad j = 0,\ldots,m$

Variables binaires $x_j^2 = x_j$

D'un Ising à un QUBO

Subset sum pb

Réécriture selon un Ising ground state

$$c(x_0, x_1, \ldots, x_m) = (a_0x_0 + a_1x_1 + \ldots + a_mx_m - T)^2$$

Exemple:

$$S = \{1,4,-2\} \longrightarrow c(x_0, x_1, x_2) = (x_0 + 4x_1 - 2x_2 - 2)^2$$
$$T = 2$$

Revient à minimiser $x_0^2 + 8x_0x_1 - 4x_0x_2 - 4x_0 + 16x_1^2 - 16x_1x_2 - 16x_1 + 4x_2^2 + 8x_2 + 4$

$x_j \in \{0,1\}, \quad j = 0,\ldots,m$

Variables binaires $x_j^2 = x_j$

Solution $x_0 = 0, \quad x_1 = x_2 = 1$

D'un Ising à un QUBO

Subset sum pb

Réécriture selon un Ising ground state

$$c(x_0, x_1, \ldots, x_m) = (a_0x_0 + a_1x_1 + \ldots + a_mx_m - T)^2$$

Toujours de degré 2 → Quadratic Unconstrained Binary Optimization (QUBO) problem

NP-Hard

D'un Ising à un QUBO

Subset sum pb

Réécriture selon un Ising ground state

$$c(x_0, x_1, \ldots, x_m) = (a_0x_0 + a_1x_1 + \ldots + a_mx_m - T)^2$$

Toujours de degré 2 → Quadratic Unconstrained Binary Optimization (QUBO) problem

Minimiser $q(x_0, \ldots, x_m)$

Avec $x_j \in \{0,1\}, \quad j = 0,\ldots,m$

QUBO car on cherche à minimiser des expressions quadratiques avec des variables binaires sans restrictions car toutes les combinaisons de 0 et 1 sont permises.

NP-Hard

D'un Ising à un QUBO

Subset sum pb

Réécriture selon un Ising ground state

$$c(x_0, x_1, \ldots, x_m) = (a_0 x_0 + a_1 x_1 + \ldots + a_m x_m - T)^2$$

Toujours de degré 2 → Quadratic Unconstrained Binary Optimization (QUBO) problem

Minimiser $q(x_0, \ldots, x_m)$

Avec $x_j \in \{0,1\}, \quad j = 0, \ldots, m$

Formulation pour un Ising?

$$H = -\sum_{j,k} J_{jk} z_j z_k - \sum_j h_j z_j$$

D'un Ising à un QUBO

Subset sum pb

Réécriture selon un Ising ground state

$$c(x_0, x_1, \ldots, x_m) = (a_0x_0 + a_1x_1 + \ldots + a_mx_m - T)^2$$

Toujours de degré 2 → Quadratic Unconstrained Binary Optimization (QUBO) problem

Minimiser $q(x_0, \ldots, x_m)$

Avec $x_j \in \{0,1\}, \quad j = 0, \ldots, m$

Formulation pour un Ising?

$$H = -\sum_{j,k} J_{jk} z_j z_k - \sum_j h_j z_j$$

Nouvelle variable

QUBO à Ising	Ising à QUBO
$x_j = \frac{(1-z_j)}{2}$	$\boxed{z_j = 1 - 2x_j}$
$x_j = 0 \rightarrow z_j = 1$ $x_j = 1 \rightarrow z_j = -1$	$x_j = 0 \rightarrow z_j = 1$ $x_j = 1 \rightarrow z_j = -1$

Polynôme quadratique qui représente exactement la fonction d'énergie

Ising à QUBO $z_j = 2x_j - 1$

D'un Ising à un QUBO

Exemple

Ising énergie $-\frac{z_0 z_1}{2} + z_2$ QUBO?

D'un Ising à un QUBO

Exemple

Ising énergie $-\frac{z_0 z_1}{2} + z_2$ QUBO? $z_j = 1 - 2x_j$

$$-\frac{(1-2x_0)(1-2x_1)}{2} + (1-2x_2) = -2x_0x_1 - x_0 + x_1 - 2x_2 + \frac{1}{2}$$

D'un Ising à un QUBO

Exemple

Ising énergie $-\frac{z_0 z_1}{2} + z_2$ QUBO? $z_j = 1 - 2x_j$

$$-\frac{(1-2x_0)(1-2x_1)}{2} + (1-2x_2) = -2x_0x_1 - x_0 + x_1 - 2x_2 + \frac{1}{2}$$

QUBO Formulation

Minimiser $-2x_0x_1 - x_0 + x_1 - 2x_2 + \frac{1}{2}$

Avec $x_j \in \{0,1\}, \quad j = 0,1,2$

D'un Ising à un QUBO

Exemple

QUBO $x_0^2 + 2x_0x_1 - 3$

Ising? $x_j = \dfrac{1 - z_j}{2}$

D'un Ising à un QUBO

Exemple

QUBO $x_0^2 + 2x_0x_1 - 3$ Ising? $x_j = \dfrac{1 - z_j}{2}$

$$\frac{z_0^2}{4} + \frac{z_0z_1}{2} - z_0 - \frac{z_1}{2} - \frac{9}{4}$$

D'un Ising à un QUBO

Exemple

QUBO $x_0^2 + 2x_0x_1 - 3$

Ising? $x_j = \dfrac{1 - z_j}{2}$

$$\frac{z_0^2}{4} + \frac{z_0 z_1}{2} - z_0 - \frac{z_1}{2} - \frac{9}{4}$$

Plusieurs problèmes

- Valeurs au carré
- Termes indépendants

D'un Ising à un QUBO

Exemple

QUBO $x_0^2 + 2x_0x_1 - 3$

Ising? $x_j = \dfrac{1 - z_j}{2}$

$$\frac{z_0^2}{4} + \frac{z_0 z_1}{2} - z_0 - \frac{z_1}{2} - \frac{9}{4}$$

Plusieurs problèmes

- Valeurs au carré
- Termes indépendants

Solutions

- $z_j = \{1; -1\}; \quad z_j^2 = 1 \longrightarrow \dfrac{1}{4} + \dfrac{z_0 z_1}{2} - z_0 - \dfrac{z_1}{2} - \dfrac{9}{4}$

D'un Ising à un QUBO

Exemple

QUBO $x_0^2 + 2x_0x_1 - 3$

Ising? $x_j = \dfrac{1 - z_j}{2}$

$$\frac{z_0^2}{4} + \frac{z_0z_1}{2} - z_0 - \frac{z_1}{2} - \frac{9}{4}$$

Plusieurs problèmes

- Valeurs au carré
- Termes indépendants

Solutions

- $z_j = \{1; -1\}; \quad z_j^2 = 1 \rightarrow \dfrac{1}{4} + \dfrac{z_0z_1}{2} - z_0 - \dfrac{z_1}{2} - \dfrac{9}{4}$
- Minimisation $\rightarrow \dfrac{z_0z_1}{2} - z_0 - \dfrac{z_1}{2}$

D'un Ising à un QUBO

Exemple

QUBO $x_0^2 + 2x_0x_1 - 3$

Ising? $x_j = \dfrac{1 - z_j}{2}$

$$\frac{z_0^2}{4} + \frac{z_0z_1}{2} - z_0 - \frac{z_1}{2} - \frac{9}{4}$$

Plusieurs problèmes

- Valeurs au carré
- Termes indépendants

Solutions

- $z_j = \{1; -1\}; \quad z_j^2 = 1 \longrightarrow \dfrac{1}{4} + \dfrac{z_0z_1}{2} - z_0 - \dfrac{z_1}{2} - \dfrac{9}{4}$
- Minimisation $\longrightarrow \dfrac{z_0z_1}{2} - z_0 - \dfrac{z_1}{2}$

Ising Formulation

Minimiser $\dfrac{z_0z_1}{2} - z_0 - \dfrac{z_1}{2}$

Avec $z_j \in \{1, -1\}, \quad j = 0,1,2$

D'un Ising à un QUBO

Exemple

A vous de jouer!!

$S = \{1, -2, 3, -4\}; \quad T = 0$

QUBO?

Ising?

D'un Ising à un QUBO

Exemple

$S = \{1, -2,3, -4\}; \quad T = 0$

QUBO?

QUBO Formulation

Minimiser $x_o^2 - 4x_0x_1 + 6x_0x_2 - 8x_0x_3 + 4x_1^2 - 12x_1x_2 + 16x_1x_3 + 9x_2^2 - 24x_2x_3 + 16x_3^2$

Avec $x_j \in \{0,1\}, \quad j = 0,1,2,3$

Ising?

D'un Ising à un QUBO

Exemple

$S = \{1, -2, 3, -4\}; \quad T = 0$

QUBO?

QUBO Formulation

Minimiser $x_o^2 - 4x_0x_1 + 6x_0x_2 - 8x_0x_3 + 4x_1^2 - 12x_1x_2 + 16x_1x_3 + 9x_2^2 - 24x_2x_3 + 16x_3^2$

Avec $x_j \in \{0,1\}, \quad j = 0,1,2,3$

Ising?

Ising Formulation

Minimiser $-z_0z_1 + \dfrac{3z_0z_2}{2} - 2z_0z_3 + z_0 - 3z_1z_2 + 4z_1z_3 - 2z + 1 - 6z_2z_3 + 3z_2 - 4z_3$

Avec $z_j \in \{1, -1\}, \quad j = 0,1,2,3$

Pour les problèmes combinatoires, l'approche Ising est généralement préférée

QUBO pour les problèmes d'optimisation

QUBO pour les problèmes d'optimisation

Binary Linear programming

Le Principe est d'optimiser une fonction linéaire avec des variables binaires en posant des contraintes linéaires.

Forme générale

Minimiser	$c_0x_0 + c_1x_1 + \ldots + c_mx_m$
Avec	$Ax \leq b$
	$x_j \in \{0,1\}, \quad j = 0,\ldots,m$

$c_0, c_1, \ldots, c_m$	Coefficients entiers
$x_0, x_1, \ldots, x_m$	Valeurs binaires
A	Matrice d'entiers
$x = (x_0, \ldots, x_m)^\dagger$	Transposée
b	Vecteur colonne d'entiers

QUBO pour les problèmes d'optimisation

Binary Linear programming

Formulation

Forme générale

Minimiser $c_0x_0 + c_1x_1 + \ldots + c_mx_m$

Avec $Ax \leq b$

$x_j \in \{0,1\}, \quad j = 0,\ldots,m$

Minimiser $-5x_0 + 3x_1 - 3x_2$

Avec $x_0 + x_2 \leq 1$

$3x_0 - x_1 + 3x_2 \leq 4$

$x_j \in \{0,1\}, \quad j = 0,1,2$

On ne peut avoir que x_0 ou x_1 car $x_j \in \{0,1\}$

Les configurations qui respectent les contraintes sont appelées "Faisable"

Karp, 1972, Reducibility among combinatorial problems

QUBO pour les problèmes d'optimisation

Binary Linear programming

Formulation

Forme générale

Minimiser $c_0x_0 + c_1x_1 + \ldots + c_mx_m$

Avec $Ax \leq b$

$x_j \in \{0,1\}, \quad j = 0,\ldots,m$

Minimiser $-5x_0 + 3x_1 - 3x_2$

Avec
$$\begin{cases} x_0 + x_2 \leq 1 \\ 3x_0 - x_1 + 3x_2 \leq 4 \end{cases}$$
$$x_j \in \{0,1\}, \quad j = 0,1,2$$

On ne peut avoir que x_0 ou x_1 car $x_j \in \{0,1\}$

Le problème est que les inégalités ne sont pas "tolérées", les contraintes doivent être modifiées pour intégrer des variables supplémentaires appelées "slack variables"

Karp, 1972, Reducibility among combinatorial problems

QUBO pour les problèmes d'optimisation

Binary Linear programming

Formulation

Forme générale

Minimiser $c_0x_0 + c_1x_1 + \ldots + c_mx_m$

Avec $Ax \le b$

$x_j \in \{0,1\}, \quad j = 0,\ldots,m$

Minimiser $-5x_0 + 3x_1 - 3x_2$

Avec
$$\begin{cases} x_0 + x_2 \le 1 \\ 3x_0 - x_1 + 3x_2 \le 4 \end{cases}$$
$$x_j \in \{0,1\}, \quad j = 0,1,2$$

On ne peut avoir que x_0 ou x_1 car $x_j \in \{0,1\}$

$$x_0 + x_2 + y_0 = 1$$

$$\begin{cases} x_0 = x_2 = 0\ ; \quad y_0 = 1 \\ x_0 = 0; x_2 = 1\ ; \quad y_0 = 0 \quad \text{ou} \quad x_0 = 1; x_2 = 0\ ; \quad y_0 = 0 \end{cases}$$

Le set de solutions "faisables" n'est pas changé

Note importante
Une règle pour la minimisation des expressions linéaires est de mettre 0 pour les coefficients positifs et 1 pour les négatifs

Karp, 1972, Reducibility among combinatorial problems

QUBO pour les problèmes d'optimisation

Binary Linear programming

Formulation

Forme générale

Minimiser $c_0x_0 + c_1x_1 + \ldots + c_mx_m$

Avec $Ax \leq b$

$x_j \in \{0,1\}, \quad j = 0,\ldots,m$

Minimiser $-5x_0 + 3x_1 - 3x_2$

Avec
$$\begin{cases} x_0 + x_2 \leq 1 \\ 3x_0 - x_1 + 3x_2 \leq 4 \end{cases}$$
$x_j \in \{0,1\}, \quad j = 0,1,2$

$x_0 + x_2 \leq 1 \rightarrow$ On ne peut avoir que x_0 ou x_1 car $x_j \in \{0,1\}$

$$x_0 + x_2 + y_0 = 1 \rightarrow \begin{cases} x_0 = x_2 = 0\ ; \quad y_0 = 1 \\ x_0 = 0; x_2 = 1\ ; \quad y_0 = 0 \quad \text{ou} \quad x_0 = 1; x_2 = 0\ ; \quad y_0 = 0 \end{cases}$$

Le set de solutions "faisables" n'est pas changé

Valeur minimale de $3x_0 - x_1 + 3x_2$ est -1 la valeur a ajouté pour la contrainte est donc de 5

5 en binaire s'écrit 101 il faut donc ajouter 3 variables binaires supplémentaires

Karp, 1972, Reducibility among combinatorial problems

QUBO pour les problèmes d'optimisation

Binary Linear programming

Formulation

Forme générale

Minimiser $c_0x_0 + c_1x_1 + \ldots + c_mx_m$

Avec $Ax \leq b$

$x_j \in \{0,1\}, \quad j = 0,\ldots,m$

Minimiser $-5x_0 + 3x_1 - 3x_2$

Avec
$$\left\{\begin{array}{l} x_0 + x_2 \leq 1 \\ 3x_0 - x_1 + 3x_2 \leq 4 \end{array}\right.$$
$$x_j \in \{0,1\}, \quad j = 0,1,2$$

On ne peut avoir que x_0 ou x_1 car $x_j \in \{0,1\}$

$$x_0 + x_2 + y_0 = 1$$

$$\left\{\begin{array}{l} x_0 = x_2 = 0\ ; \quad y_0 = 1 \\ x_0 = 0; x_2 = 1\ ; \quad y_0 = 0 \quad \text{ou} \quad x_0 = 1; x_2 = 0\ ; \quad y_0 = 0 \end{array}\right.$$

Le set de solutions "faisables" n'est pas changé

$$3x_0 - x_1 + 3x_2 + y_1 + 2y_2 + 2y_3 = 4$$

$$y_1 = 1\ ;\ 2y_2 = 2\ ;\ 2y_3 = 2\ ;\ \sum_j y_j = 5$$

Reformulation

Karp, 1972, Reducibility among combinatorial problems

QUBO pour les problèmes d'optimisation

Binary Linear programming

Formulation

Forme générale

$$\begin{aligned} &\text{Minimiser} && c_0x_0 + c_1x_1 + \ldots + c_mx_m \\ &\text{Avec} && Ax \leq b \\ & && x_j \in \{0,1\}, \quad j = 0,\ldots,m \end{aligned}$$

$$\begin{aligned} &\text{Minimiser} && -5x_0 + 3x_1 - 3x_2 \\ &\text{Avec} && x_0 + x_2 + y_0 = 1 \\ & && 3x_0 - x_1 + 3x_2 + y_1 + 2y_2 + 2y_3 = 4 \\ & && x_j \in \{0,1\}, \quad j = 0,1,2 \\ & && y_j \in \{0,1\}, \quad j = 0,1,2,3 \end{aligned}$$

Forme d'un QUBO ?

Karp, 1972, Reducibility among combinatorial problems

QUBO pour les problèmes d'optimisation

Binary Linear programming

Formulation

Forme générale

Minimiser $c_0x_0 + c_1x_1 + \ldots + c_mx_m$

Avec $Ax \leq b$

$x_j \in \{0,1\}, \quad j = 0,\ldots,m$

Minimiser $-5x_0 + 3x_1 - 3x_2$

Avec

$$\boxed{\begin{aligned} &x_0 + x_2 + y_0 = 1 \\ &3x_0 - x_1 + 3x_2 + y_1 + 2y_2 + 2y_3 = 4 \end{aligned}}$$

$x_j \in \{0,1\}, \quad j = 0,1,2$

$y_j \in \{0,1\}, \quad j = 0,1,2,3$

Ajout en termes de pénalité pour la minimisation

Karp, 1972, Reducibility among combinatorial problems

QUBO pour les problèmes d'optimisation

Binary Linear programming

Formulation

Forme générale

Minimiser	$c_0x_0 + c_1x_1 + \ldots + c_mx_m$
Avec	$Ax \leq b$
	$x_j \in \{0,1\}, \quad j = 0,\ldots,m$

Minimiser $-5x_0 + 3x_1 - 3x_2 + B(x_0 + x_2 + y_0 - 1)^2 + B(3x_0 - x_1 + 3x_2 + y_1 + 2y_2 + 2y_3 - 4)^2$

Avec $x_j \in \{0,1\}, \quad j = 0,1,2$

$y_j \in \{0,1\}, \quad j = 0,1,2,3$

Karp, 1972, Reducibility among combinatorial problems

QUBO pour les problèmes d'optimisation

Binary Linear programming

Formulation

Forme générale

Minimiser $c_0x_0 + c_1x_1 + \ldots + c_mx_m$

Avec $Ax \leq b$

$x_j \in \{0,1\}, \quad j = 0,\ldots,m$

Minimiser $-5x_0 + 3x_1 - 3x_2 + B(x_0 + x_2 + y_0 - 1)^2 + B(3x_0 - x_1 + 3x_2 + y_1 + 2y_2 + 2y_3 - 4)^2$

Avec $x_j \in \{0,1\}, \quad j = 0,1,2$

$y_j \in \{0,1\}, \quad j = 0,1,2,3$

B doit être grand pour ne pas que les conditions soient violées

Ici si B = 11 on a une forme de QUBO dont les contraintes ne peuvent pas être violées

Minimiser $-5x_0 + 3x_1 - 3x_2 + 11(x_0 + x_2 + y_0 - 1)^2 + 11(3x_0 - x_1 + 3x_2 + y_1 + 2y_2 + 2y_3 - 4)^2$

Avec $x_j \in \{0,1\}, \quad j = 0,1,2$

$y_j \in \{0,1\}, \quad j = 0,1,2,3$

Karp, 1972, Reducibility among combinatorial problems

QUBO pour les problèmes d'optimisation

Binary Linear programming

Formulation

Forme générale

Minimiser $c_0x_0 + c_1x_1 + \ldots + c_mx_m$

Avec $Ax \leq b$

$x_j \in \{0,1\}, \quad j = 0,\ldots,m$

Minimiser $-5x_0 + 3x_1 - 3x_2 + B(x_0 + x_2 + y_0 - 1)^2 + B(3x_0 - x_1 + 3x_2 + y_1 + 2y_2 + 2y_3 - 4)^2$

Avec $x_j \in \{0,1\}, \quad j = 0,1,2$

$y_j \in \{0,1\},$

B doit être ... soient violées

Avec cette approche vous savez transformer tout problème linéaire en QUBO

Slack variables

Pénalités

Ici si B = 11 on a une forme de c... ntraintes ne peuvent pas être violées

Minimiser $-5x_0 + 3x_1 - 3x_2 + 11(x_0 + x_2 + y_0 - 1)^2 + 11(3x_0 - x_1 + 3x_2 + y_1 + 2y_2 + 2y_3 - 4)^2$

Avec $x_j \in \{0,1\}, \quad j = 0,1,2$

$y_j \in \{0,1\}, \quad j = 0,1,2,3$

Karp, 1972, Reducibility among combinatorial problems

QUBO pour les problèmes d'optimisation

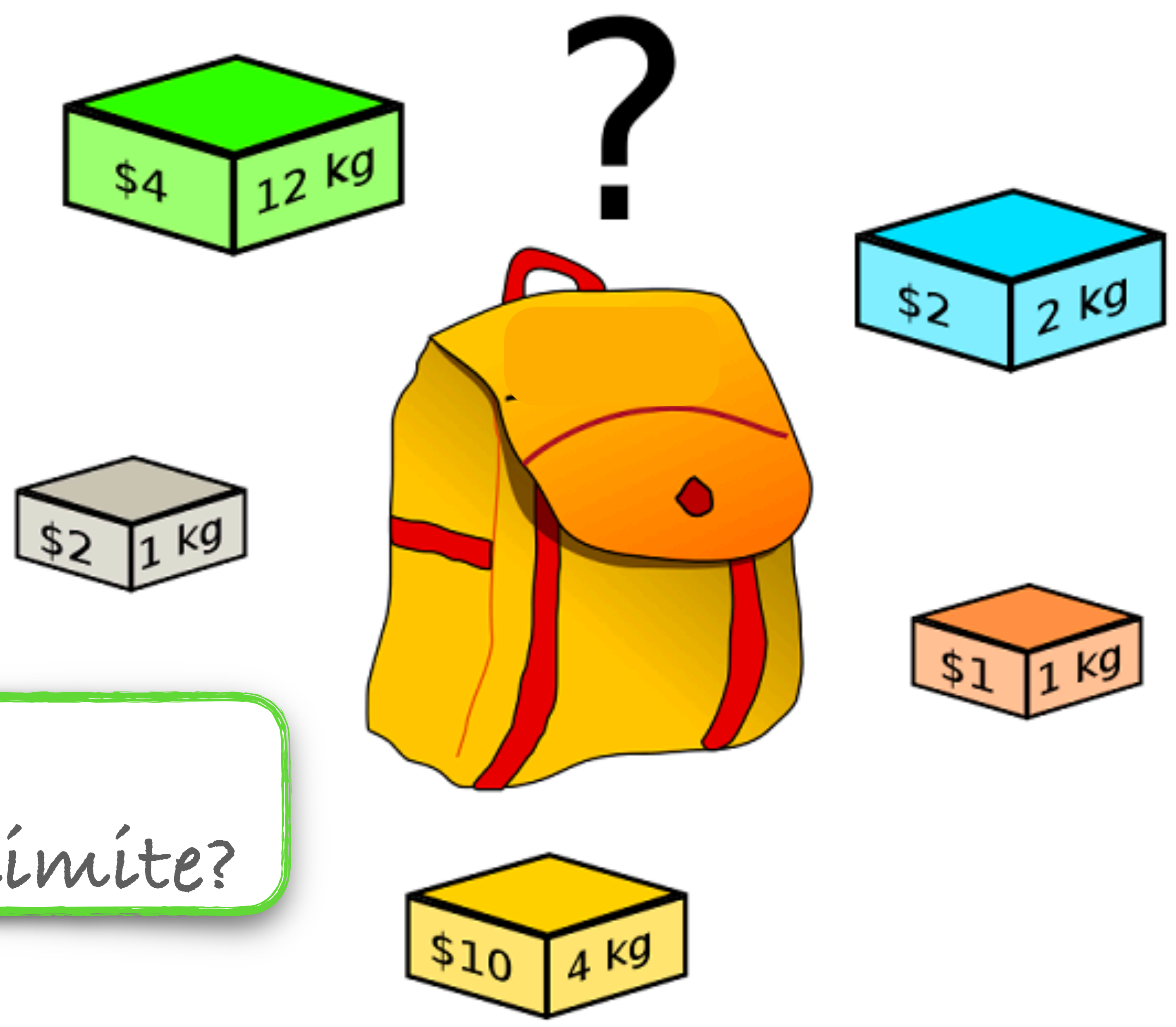
Knapsack Problem

Liste d'objets $j = 0, \ldots, m$

Chaque objet a un poids w_j
et une valeur c_j

On a un poids maximum W

Comment obtenir la valeur maximum sans dépasser le poids limite?



$$w_0 = 12 \;;\; c_0 = 4$$
$$w_1 = 2 \;;\; c_1 = 2$$
$$w_2 = 1 \;;\; c_2 = 2$$
$$w_3 = 1 \;;\; c_3 = 1$$
$$w_4 = 4 \;;\; c_4 = 10$$
$$W = 17$$

NP-Hard

https://www.hackerearth.com/practice/notes/the-knapsack-problem/

QUBO pour les problèmes d'optimisation

Knapsack Problem

Liste d'objets $j = 0,\ldots,m$

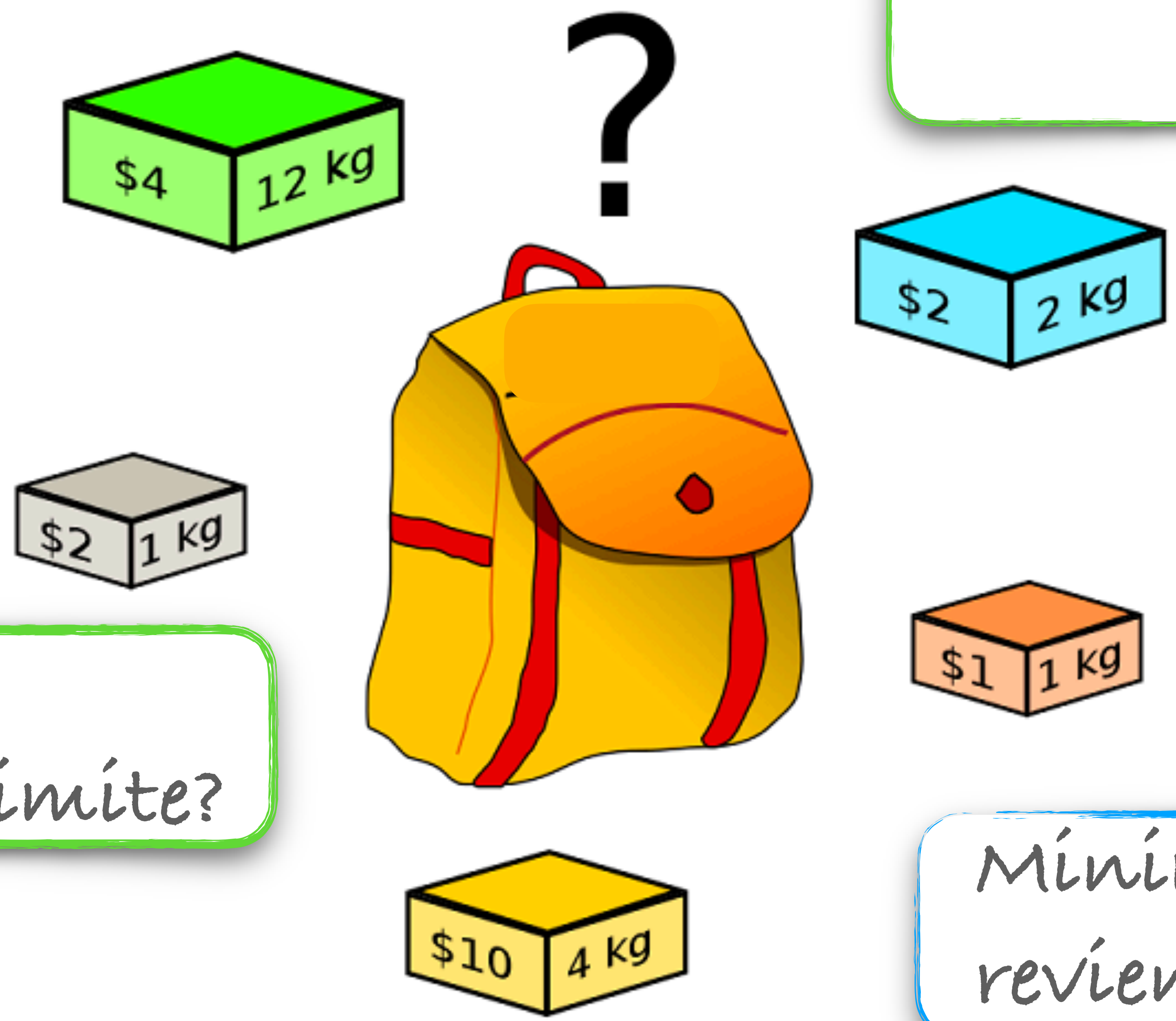
Chaque objet a un poids w_j
et une valeur c_j

On a un poids maximum W

Minimiser $\quad -c_0x_0 - c_1x_1 - \ldots - c_mx_m$

Avec $\quad w_0x_0 + w_1x_1 + \ldots + w_mx_m \leq W$

$$x_j \in \{0,1\}, \quad j = 0,\ldots,m$$



Comment obtenir la valeur maximum sans dépasser le poids limite?

$w_0 = 12\ ;\ c_0 = 4$

$w_1 = 2\ ;\ c_1 = 2$

$w_2 = 1\ ;\ c_2 = 2$

$w_3 = 1\ ;\ c_3 = 1$

$w_4 = 4\ ;\ c_4 = 10$

$W = 17$

Minimiser une expression négative revient à maximiser le problème

https://www.hackerearth.com/practice/notes/the-knapsack-problem/

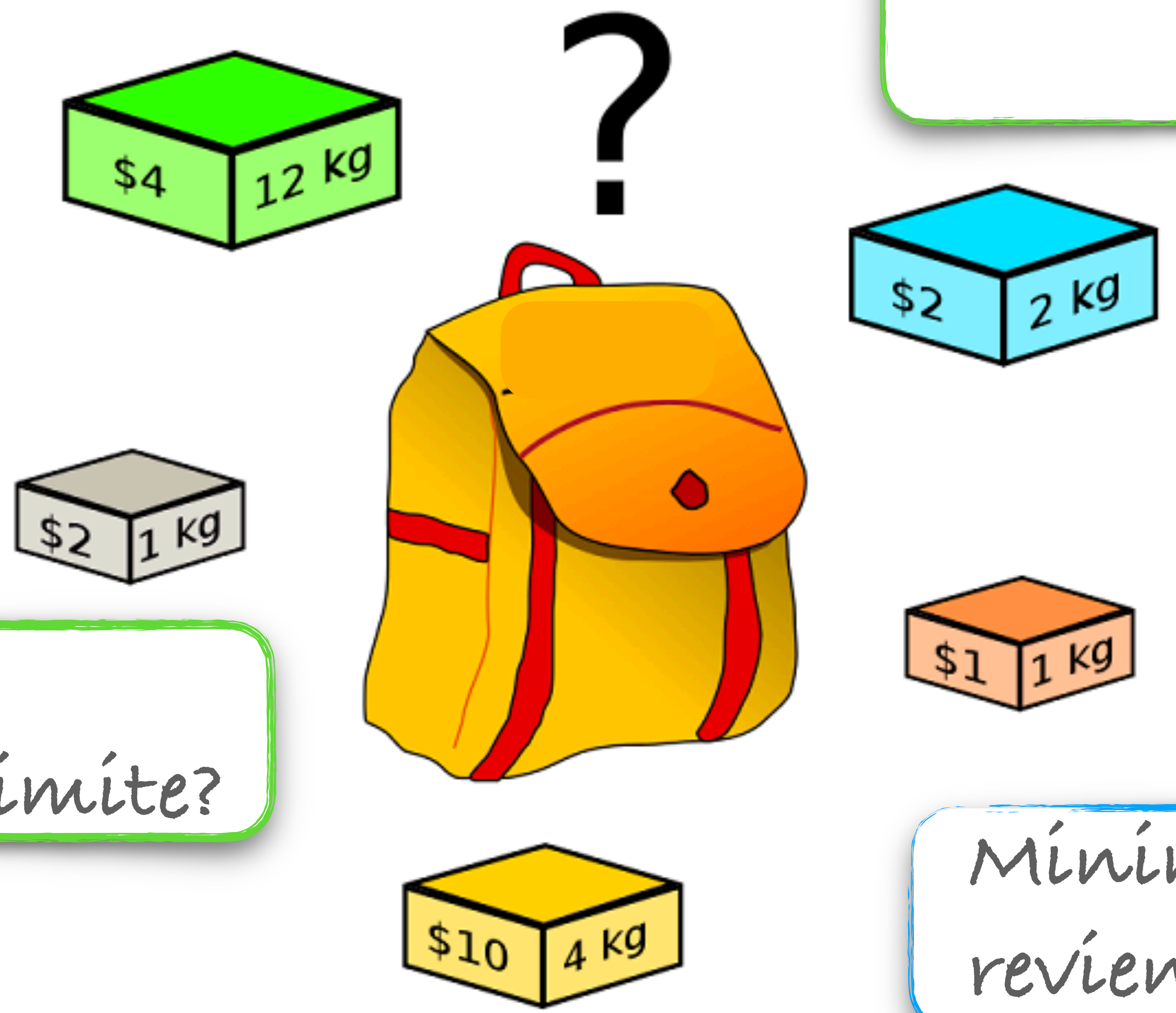
QUBO pour les problèmes d'optimisation

Knapsack Problem

Liste d'objets $j = 0, \ldots, m$

Chaque objet a un poids w_j
et une valeur c_j

On a un poids maximum W



Minimiser	$-c_0x_0 - c_1x_1 - \ldots - c_mx_m$
Avec	$w_0x_0 + w_1x_1 + \ldots + w_mx_m \leq W$
	$x_j \in \{0,1\}, \quad j = 0,\ldots,m$

Comment obtenir la valeur maximum sans dépasser le poids limite?

$w_0 = 12 \ ; \ c_0 = 4$
$w_1 = 2 \ ; \ c_1 = 2$
$w_2 = 1 \ ; \ c_2 = 2$
$w_3 = 1 \ ; \ c_3 = 1$
$w_4 = 4 \ ; \ c_4 = 10$
$W = 17$

Minimiser une expression négative revient à maximiser le problème

Minimiser	$-4x_0 - 2x_1 - 2x_2 - 1x_3 - 10x_4$
Avec	$12x_0 + 2x_1 + 1x_2 + 1x_3 + 4x_4 \leq 17$
	$x_j \in \{0,1\}, \quad j = 0,1,2,3,4$

NP-Hard

https://www.hackerearth.com/practice/notes/the-knapsack-problem/

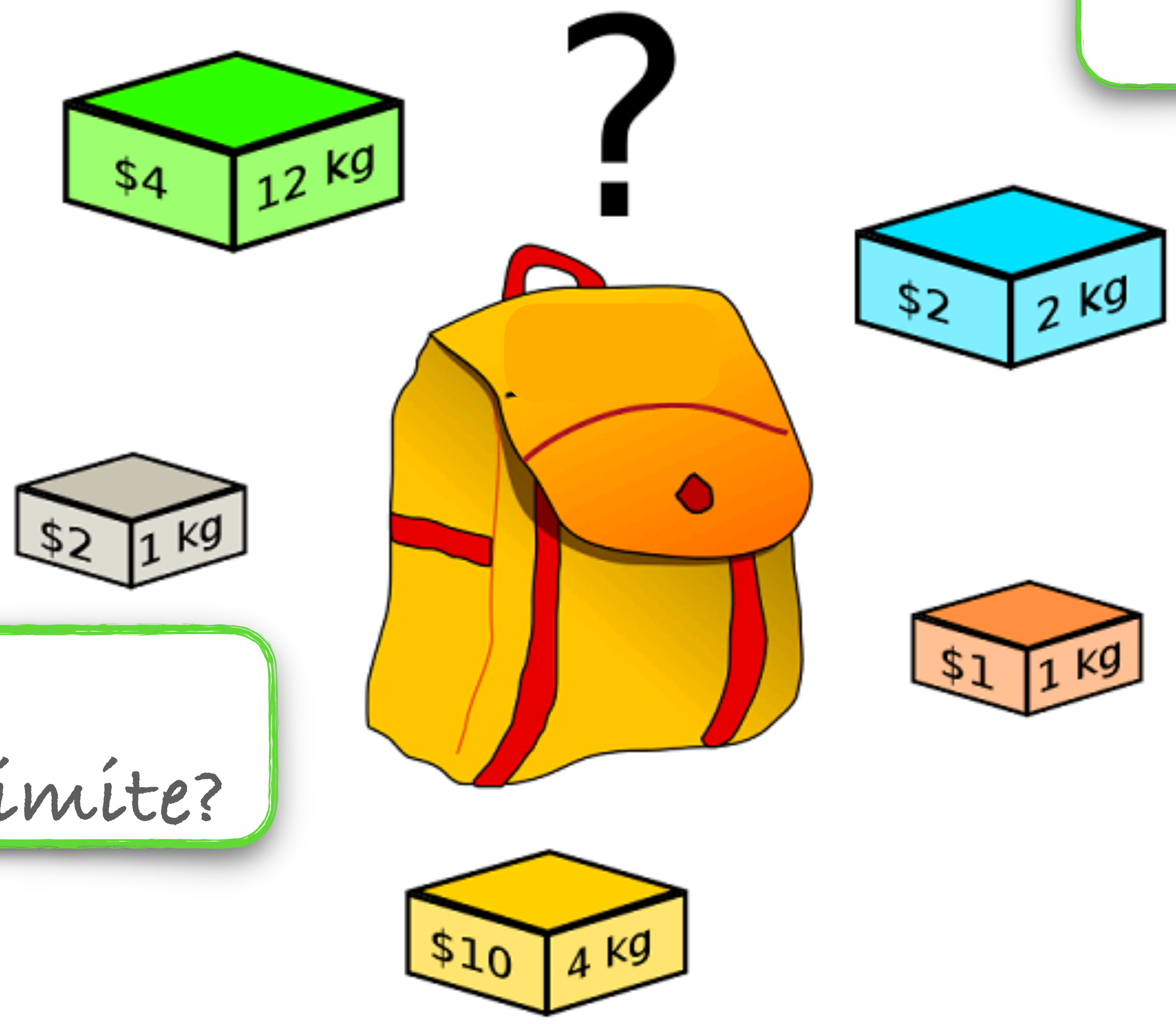
QUBO pour les problèmes d'optimisation

Knapsack Problem

Liste d'objets $j = 0,\ldots,m$

Chaque objet a un poids w_j
et une valeur c_j

On a un poids maximum W



Minimiser $\quad -4x_0 - 2x_1 - 2x_2 - 1x_3 - 10x_4$

Avec $\quad 12x_0 + 2x_1 + 1x_2 + 1x_3 + 4x_4 \leq 17$

$$x_j \in \{0,1\}, \quad j = 0,1,2,3,4$$

QUBO?

Comment obtenir la valeur maximum sans dépasser le poids limite?

$$w_0 = 12\ ;\ c_0 = 4$$
$$w_1 = 2\ ;\ c_1 = 2$$
$$w_2 = 1\ ;\ c_2 = 2$$
$$w_3 = 1\ ;\ c_3 = 1$$
$$w_4 = 4\ ;\ c_4 = 10$$
$$W = 17$$

https://www.hackerearth.com/practice/notes/the-knapsack-problem/

QUBO pour les problèmes d'optimisation

Knapsack Problem

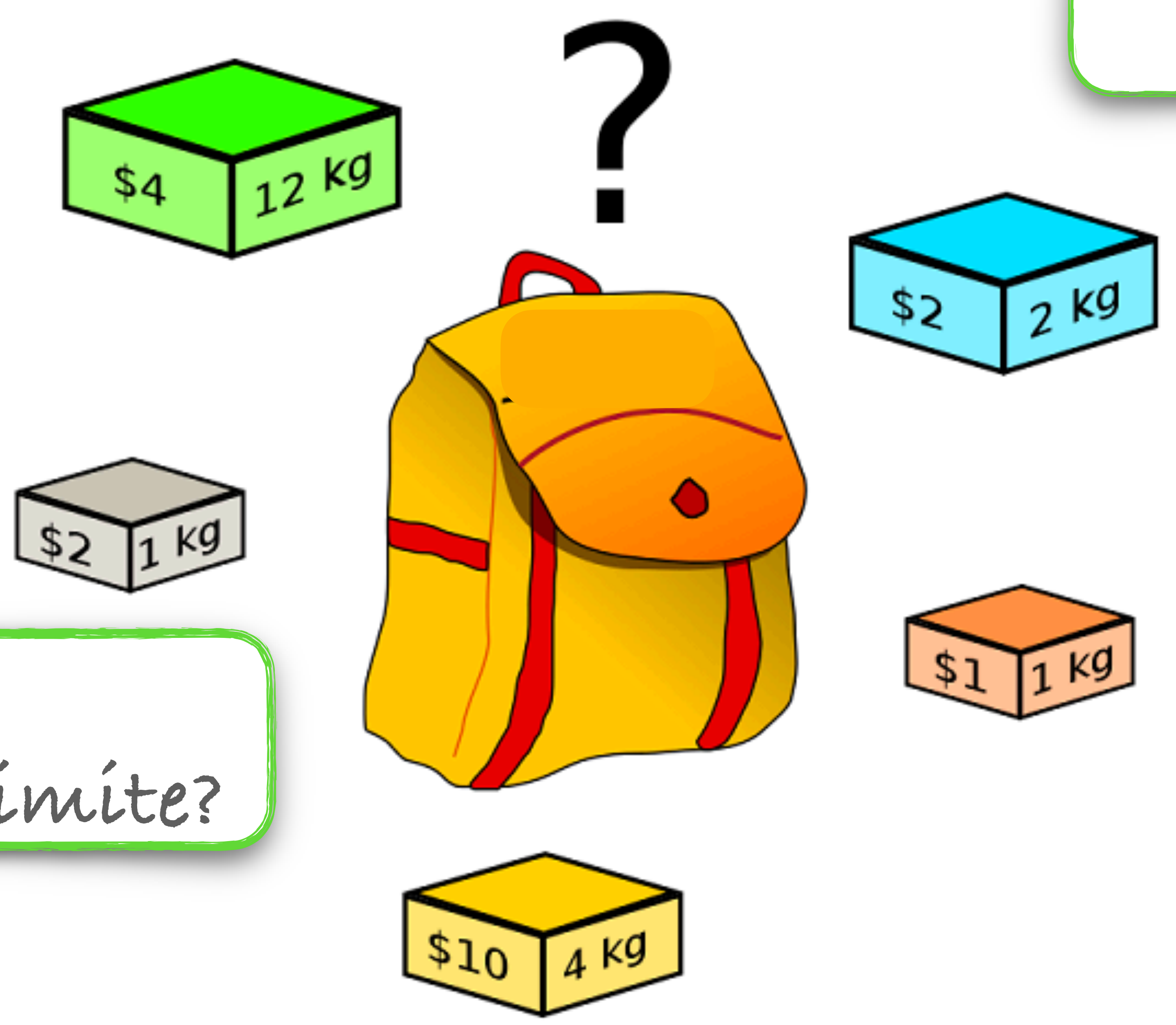
Liste d'objets $j = 0,\ldots,m$

Chaque objet a un poids w_j
et une valeur c_j

On a un poids maximum W

Comment obtenir la valeur maximum sans dépasser le poids limite?

$$w_0 = 12\ ;\ c_0 = 4$$
$$w_1 = 2\ ;\ c_1 = 2$$
$$w_2 = 1\ ;\ c_2 = 2$$
$$w_3 = 1\ ;\ c_3 = 1$$
$$w_4 = 4\ ;\ c_4 = 10$$
$$W = 17$$



Minimiser $-4x_0 - 2x_1 - 2x_2 - 1x_3 - 10x_4$

Avec $12x_0 + 2x_1 + 1x_2 + 1x_3 + 4x_4 \le 17$

$x_j \in \{0,1\}, \quad j = 0,1,2,3,4$

QUBO?

$17 = 10001$

https://www.hackerearth.com/practice/notes/the-knapsack-problem/

QUBO pour les problèmes d'optimisation

Knapsack Problem

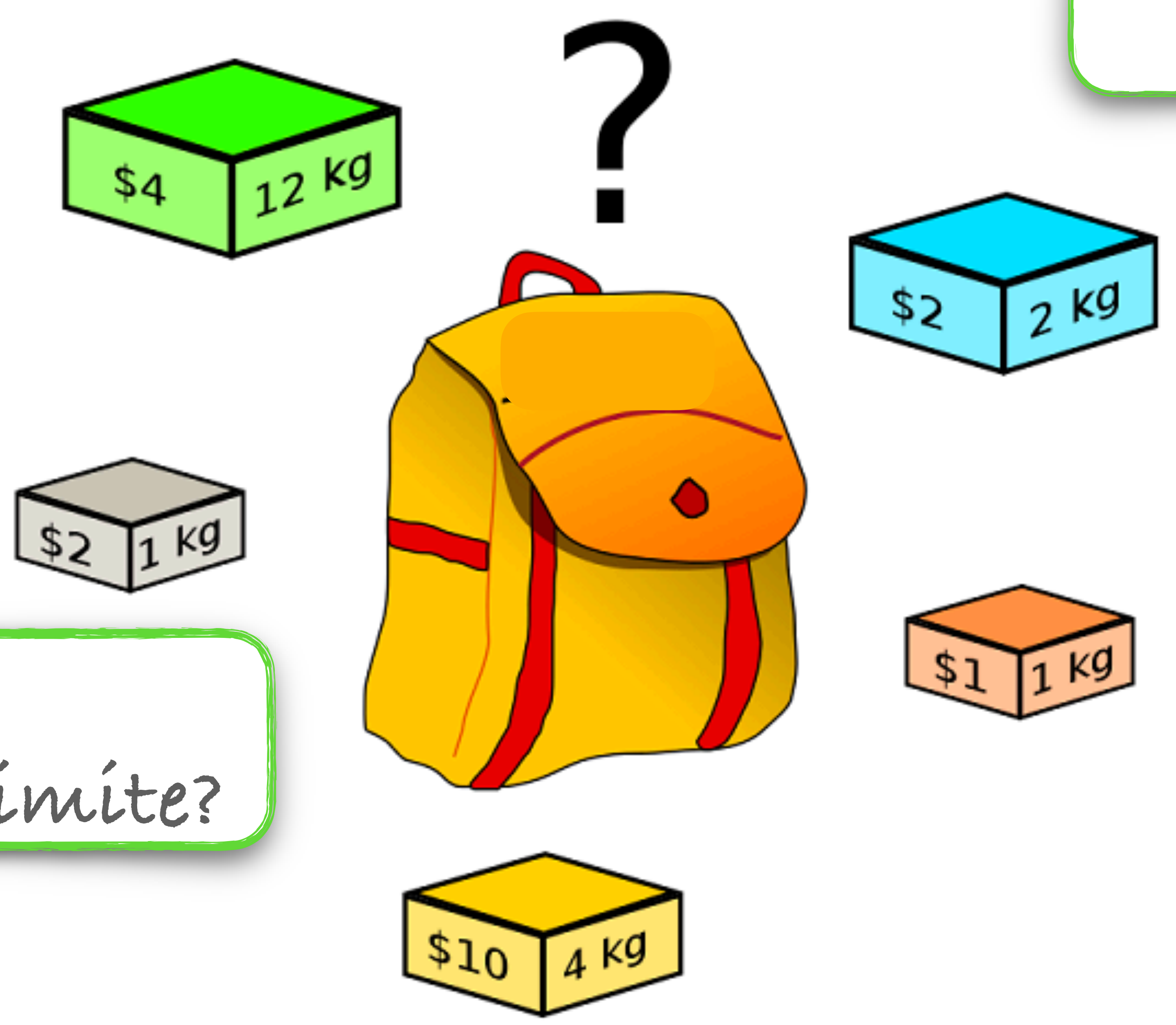
Liste d'objets $j = 0,\ldots,m$

Chaque objet a un poids w_j
et une valeur c_j

On a un poids maximum W

Comment obtenir la valeur maximum sans dépasser le poids limite?

$$w_0 = 12\ ;\ c_0 = 4$$
$$w_1 = 2\ ;\ c_1 = 2$$
$$w_2 = 1\ ;\ c_2 = 2$$
$$w_3 = 1\ ;\ c_3 = 1$$
$$w_4 = 4\ ;\ c_4 = 10$$
$$W = 17$$



Minimiser $\quad -4x_0 - 2x_1 - 2x_2 - 1x_3 - 10x_4$

Avec $\quad 12x_0 + 2x_1 + 1x_2 + 1x_3 + 4x_4 \leq 17$

$$x_j \in \{0,1\}, \quad j = 0,1,2,3,4$$

QUBO?

17 = 10001

$$12x_0 + 2x_1 + 1x_2 + 1x_3 + 4x_4$$
$$+3y_0 + 3y_1 + 3y_2 + 4y_3 + 4y_4 = 17$$

https://www.hackerearth.com/practice/notes/the-knapsack-problem/

QUBO pour les problèmes d'optimisation

Knapsack Problem

Liste d'objets $j = 0, \dots, m$

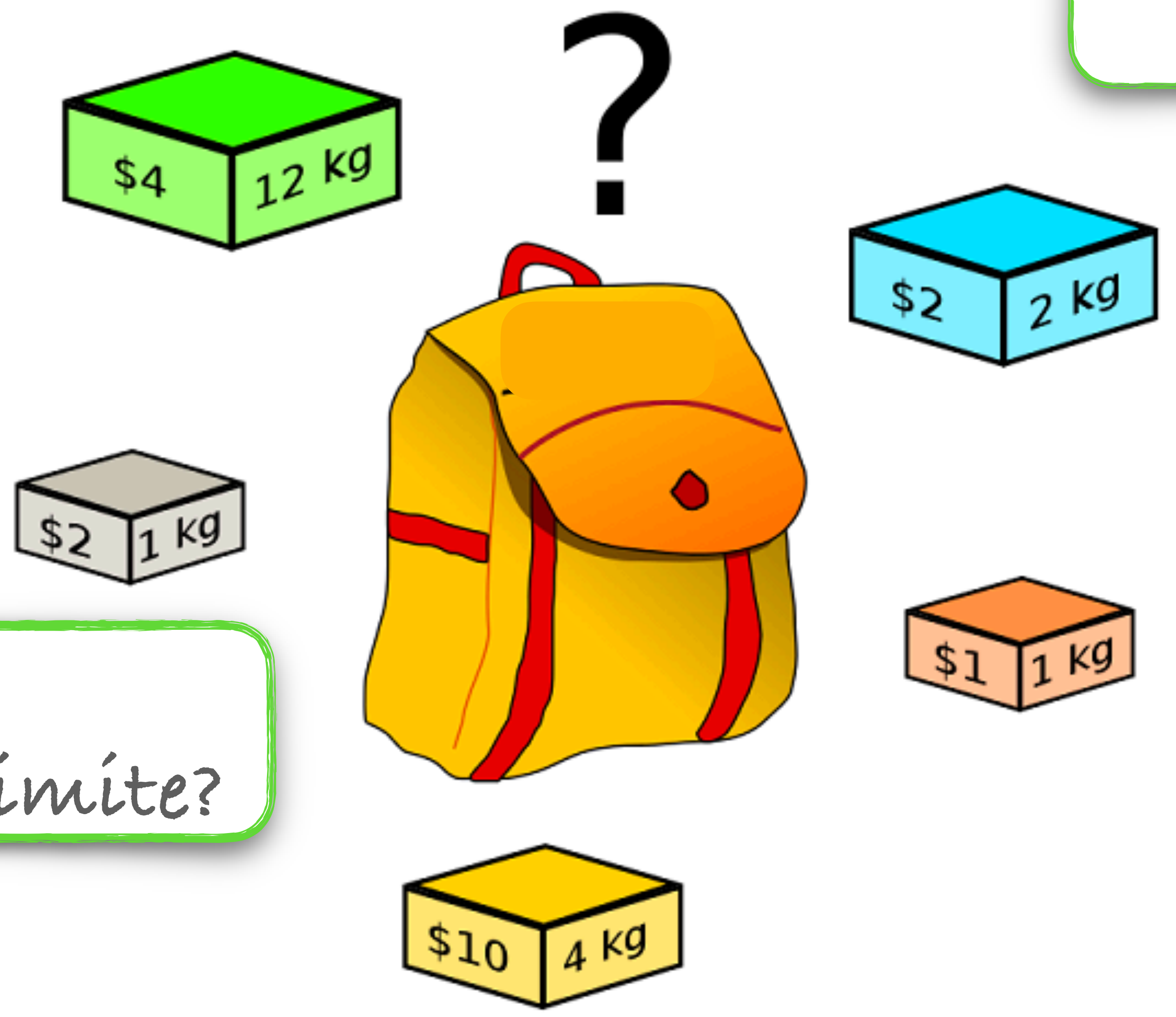
Chaque objet a un poids w_j

et une valeur c_j

On a un poids maximum W

Comment obtenir la valeur maximum sans dépasser le poids limite?

$$w_0 = 12 \; ; \; c_0 = 4$$
$$w_1 = 2 \; ; \; c_1 = 2$$
$$w_2 = 1 \; ; \; c_2 = 2$$
$$w_3 = 1 \; ; \; c_3 = 1$$
$$w_4 = 4 \; ; \; c_4 = 10$$
$$W = 17$$



Minimiser $\quad -4x_0 - 2x_1 - 2x_2 - 1x_3 - 10x_4$

Avec $\quad 12x_0 + 2x_1 + 1x_2 + 1x_3 + 4x_4 \leq 17$

$$x_j \in \{0,1\}, \quad j = 0,1,2,3,4$$

QUBO?

17 = 10001

$$12x_0 + 2x_1 + 1x_2 + 1x_3 + 4x_4$$
$$+3y_0 + 3y_1 + 3y_2 + 4y_3 + 4y_4 = 17$$

Minimiser $\quad 12x_0 + 2x_1 + 1x_2 + 1x_3 + 4x_4$

$$+B(3y_0 + 3y_1 + 3y_2 + 4y_3 + 4y_4 - 17)^2$$

Avec $\quad x_j, y_j \in \{0,1\}, \quad j = 0,1,2,3,4$

https://www.hackerearth.com/practice/notes/the-knapsack-problem/

QUBO pour les problèmes d'optimisation

Knapsack Problem

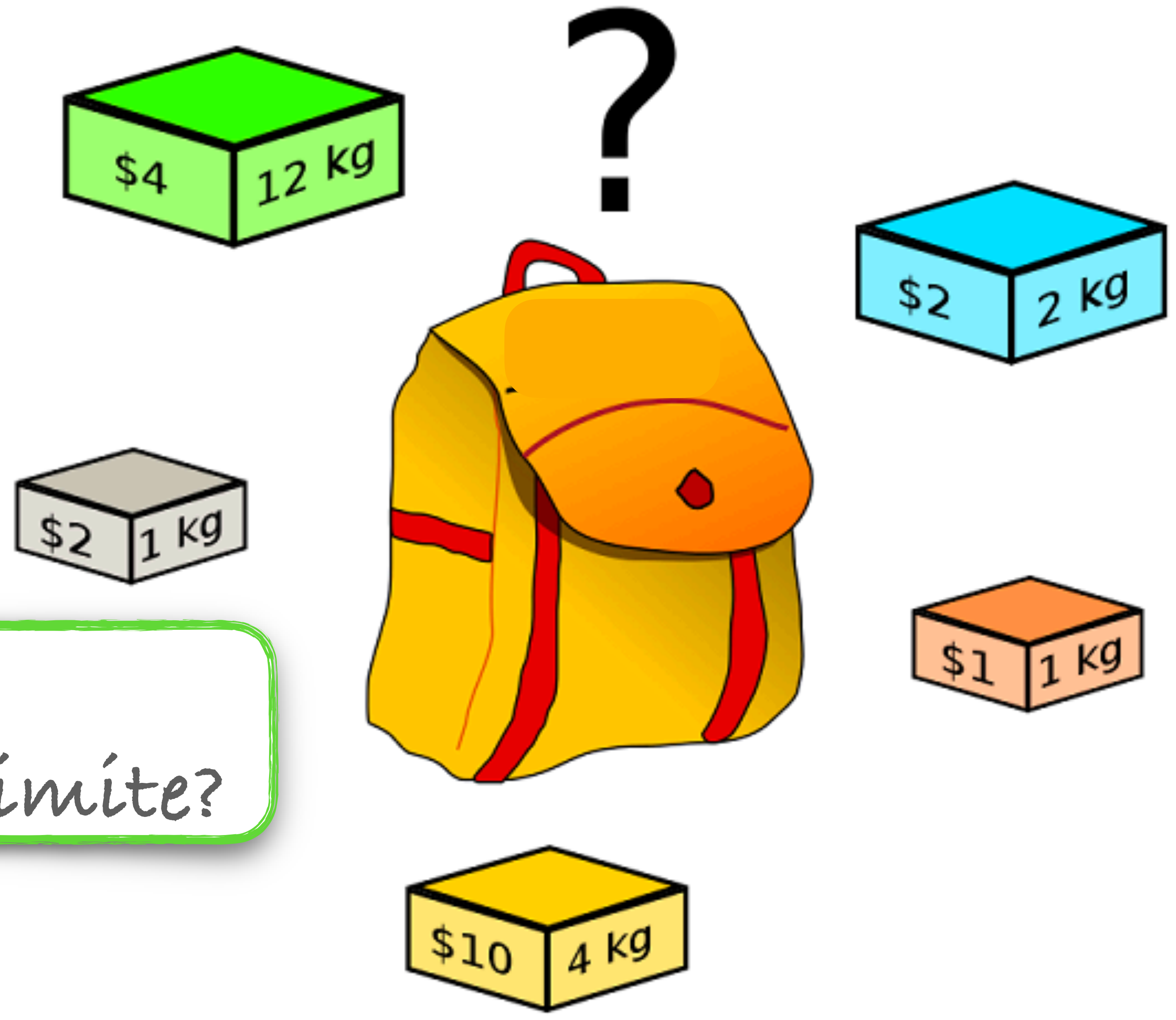
Liste d'objets $j = 0, \ldots, m$

Chaque objet a un poids w_j
et une valeur c_j

On a un poids maximum W

Comment obtenir la valeur maximum sans dépasser le poids limite?

$$w_0 = 12 \; ; \; c_0 = 4$$
$$w_1 = 2 \; ; \; c_1 = 2$$
$$w_2 = 1 \; ; \; c_2 = 2$$
$$w_3 = 1 \; ; \; c_3 = 1$$
$$w_4 = 4 \; ; \; c_4 = 10$$
$$W = 25$$



Une variation serait d'utiliser des entiers plutôt que du binaire pour représenter le nombre de fois qu'un paquet peut être utilisé

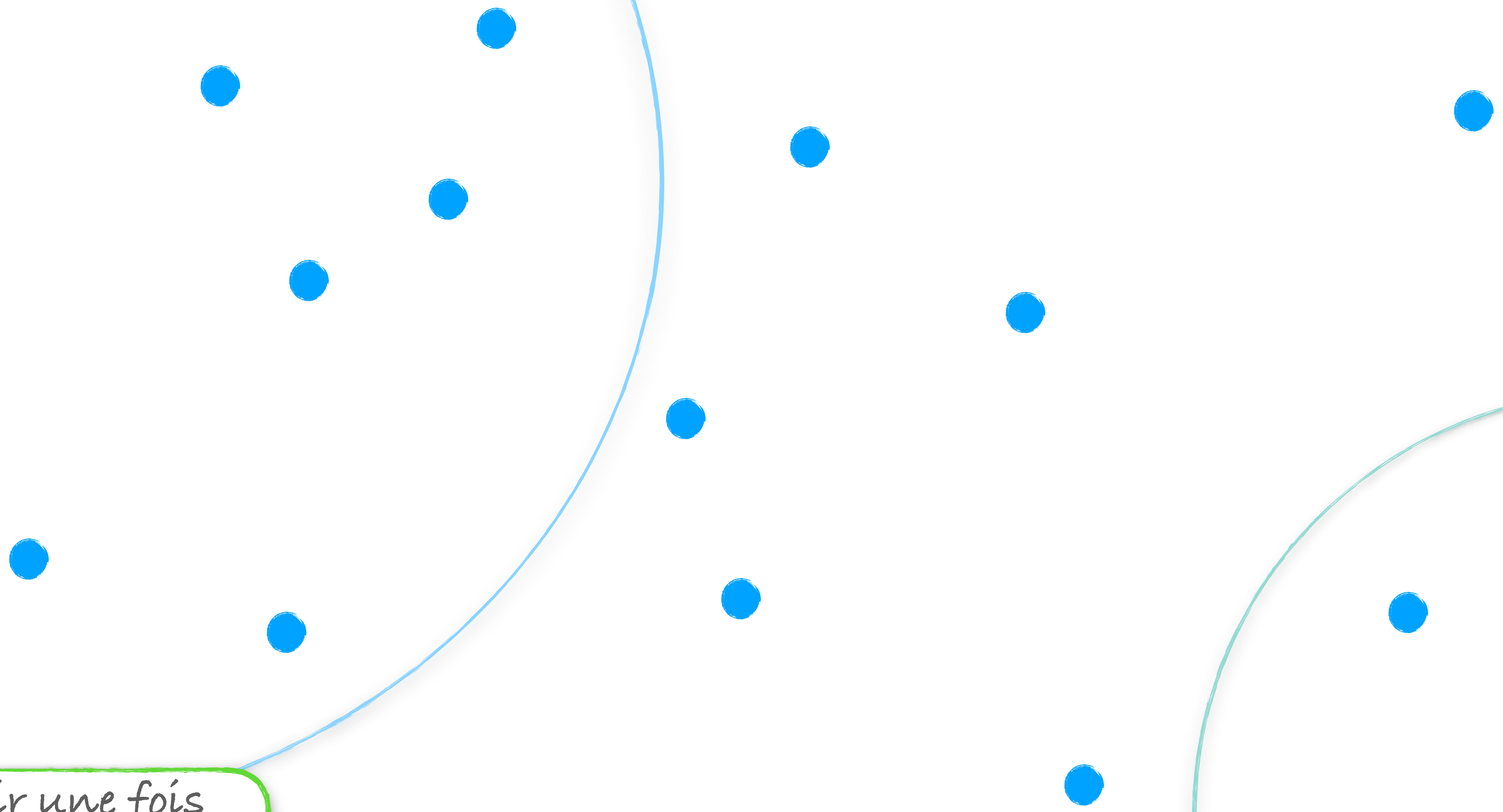
https://www.hackerearth.com/practice/notes/the-knapsack-problem/

QUBO pour les problèmes d'optimisation

TSP

Prenons des villes

● Villes/noeuds



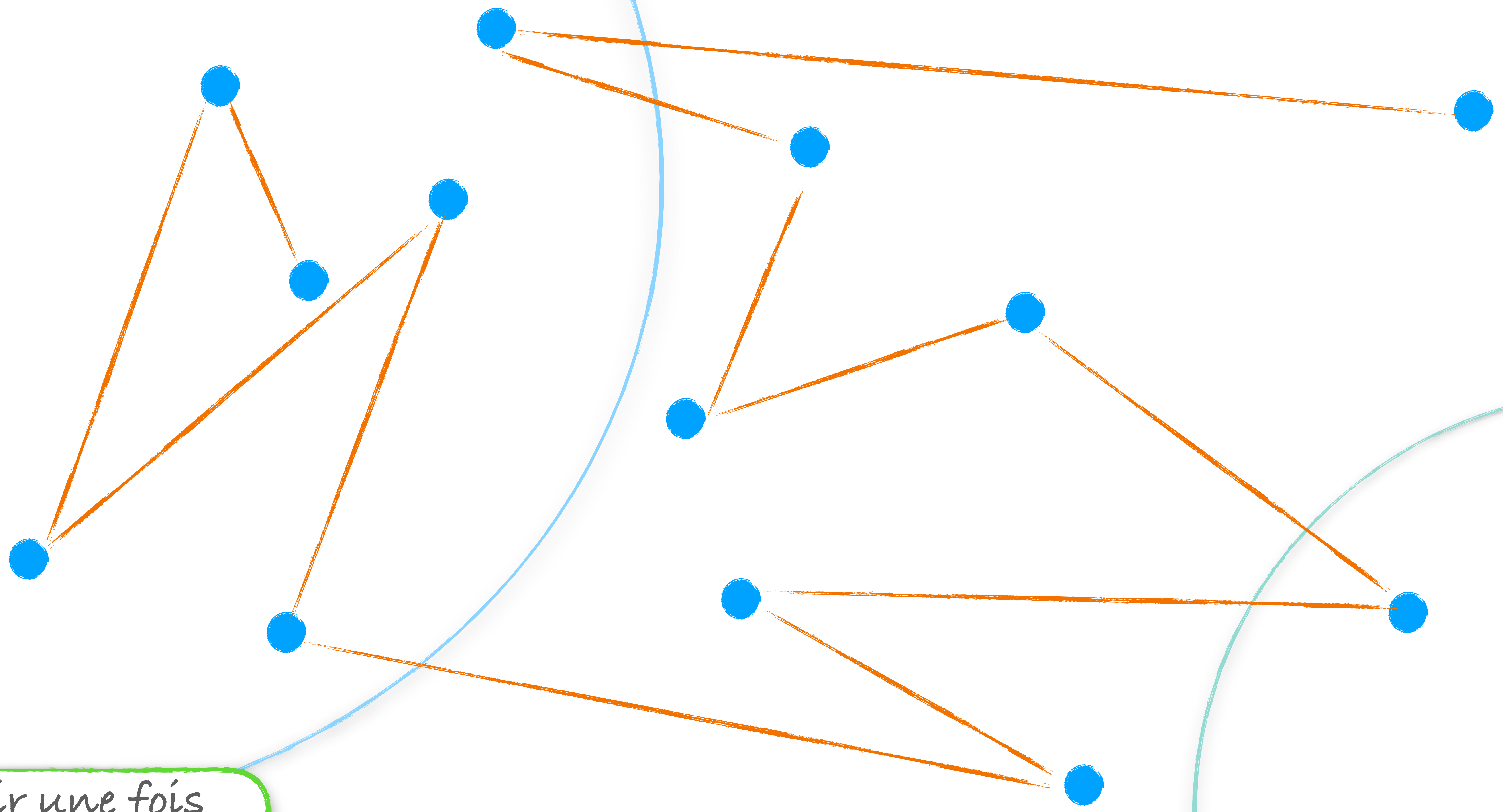
Comment parcourir une fois toutes les villes avec le minimum de distance, temps, coût…?

QUBO pour les problèmes d'optimisation

TSP

Prenons des villes

● Villes/noeuds



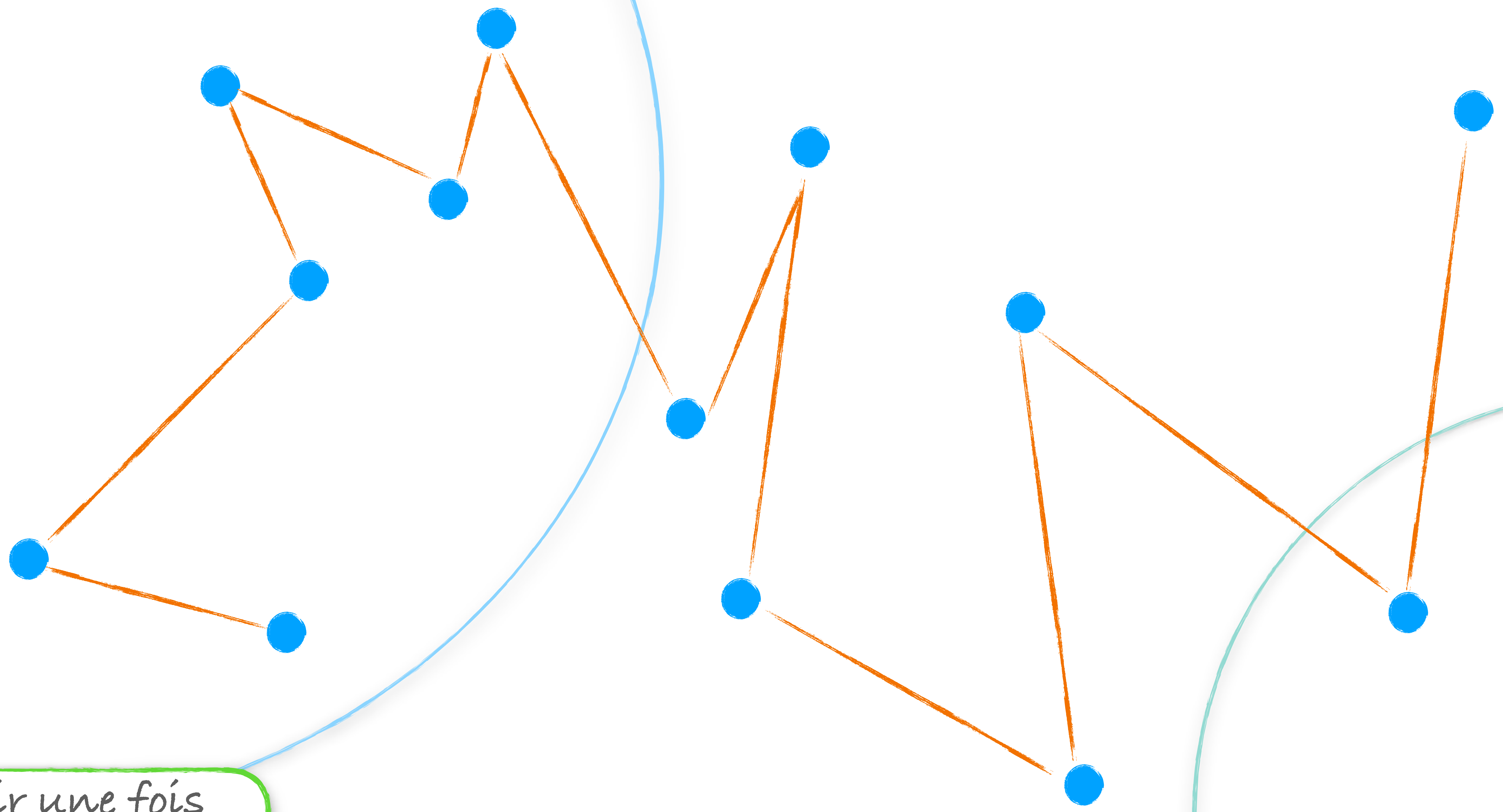
Comment parcourir une fois toutes les villes avec le minimum de distance, temps, coût…?

QUBO pour les problèmes d'optimisation

TSP

Prenons des villes

● Villes/noeuds



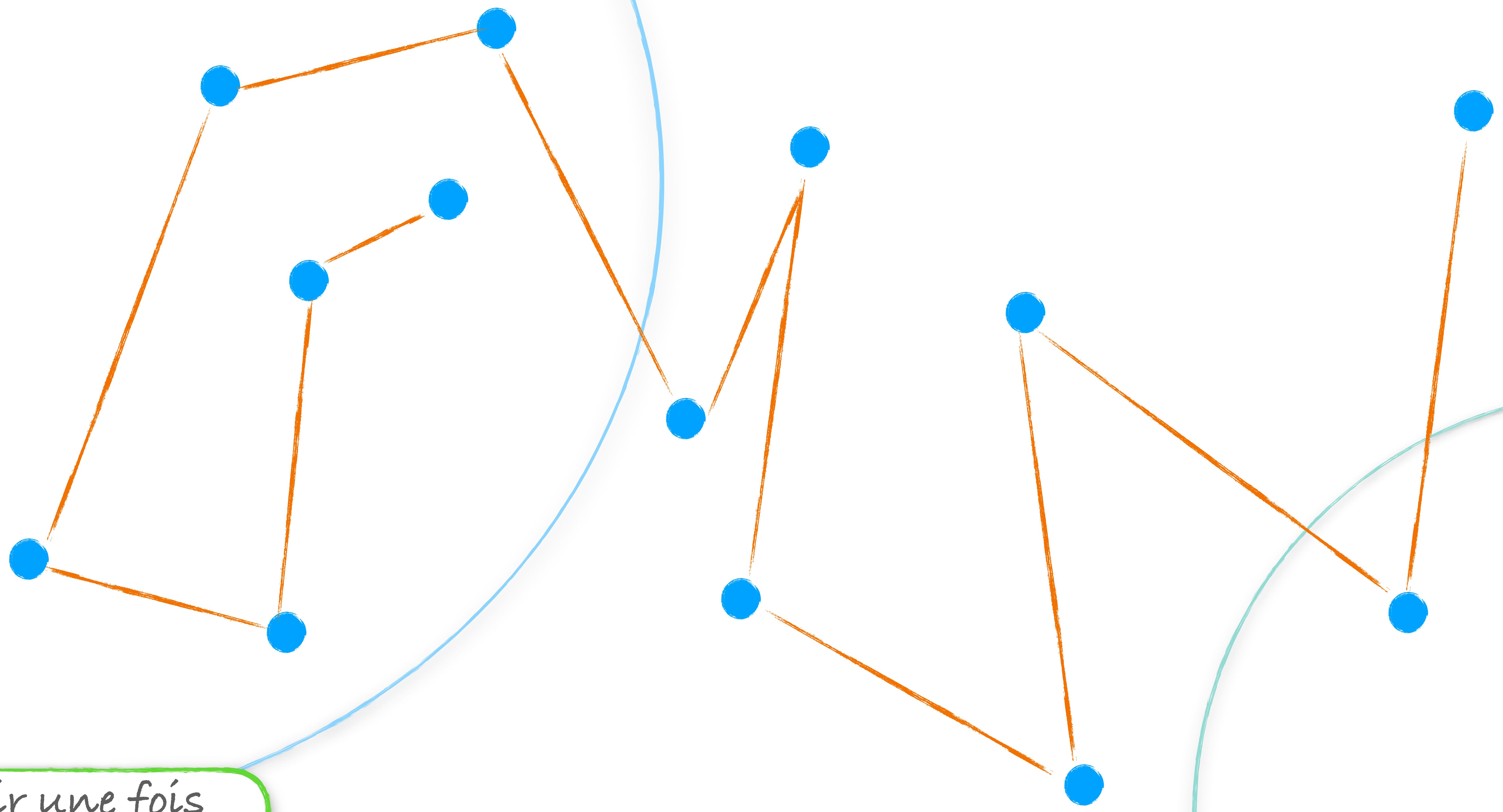
Comment parcourir une fois toutes les villes avec le minimum de distance, temps, coût...?

QUBO pour les problèmes d'optimisation

TSP

Prenons des villes

● Villes/noeuds



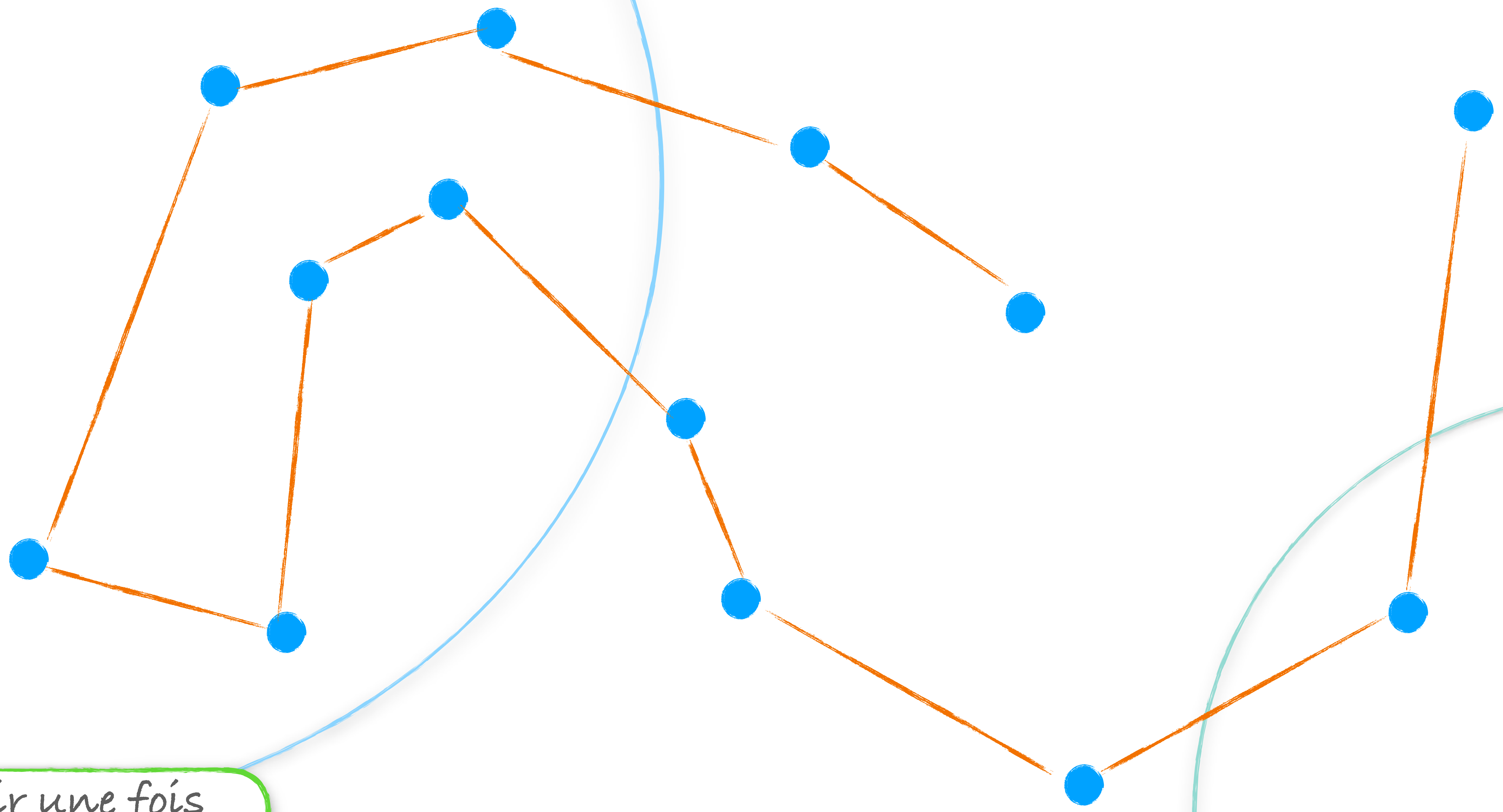
Comment parcourir une fois toutes les villes avec le minimum de distance, temps, coût...?

QUBO pour les problèmes d'optimisation

TSP

Prenons des villes

● Villes/noeuds



Comment parcourir une fois toutes les villes avec le minimum de distance, temps, coût...?

QUBO pour les problèmes d'optimisation

TSP

Minimisation de la somme des poids sur les liens (edges) entre les villes (noeuds)

Prenons des villes

● Villes/noeuds

$j = 0, \ldots, m$

j et l sont des noeuds reliés par un lien qui a un poids

w_{jl}

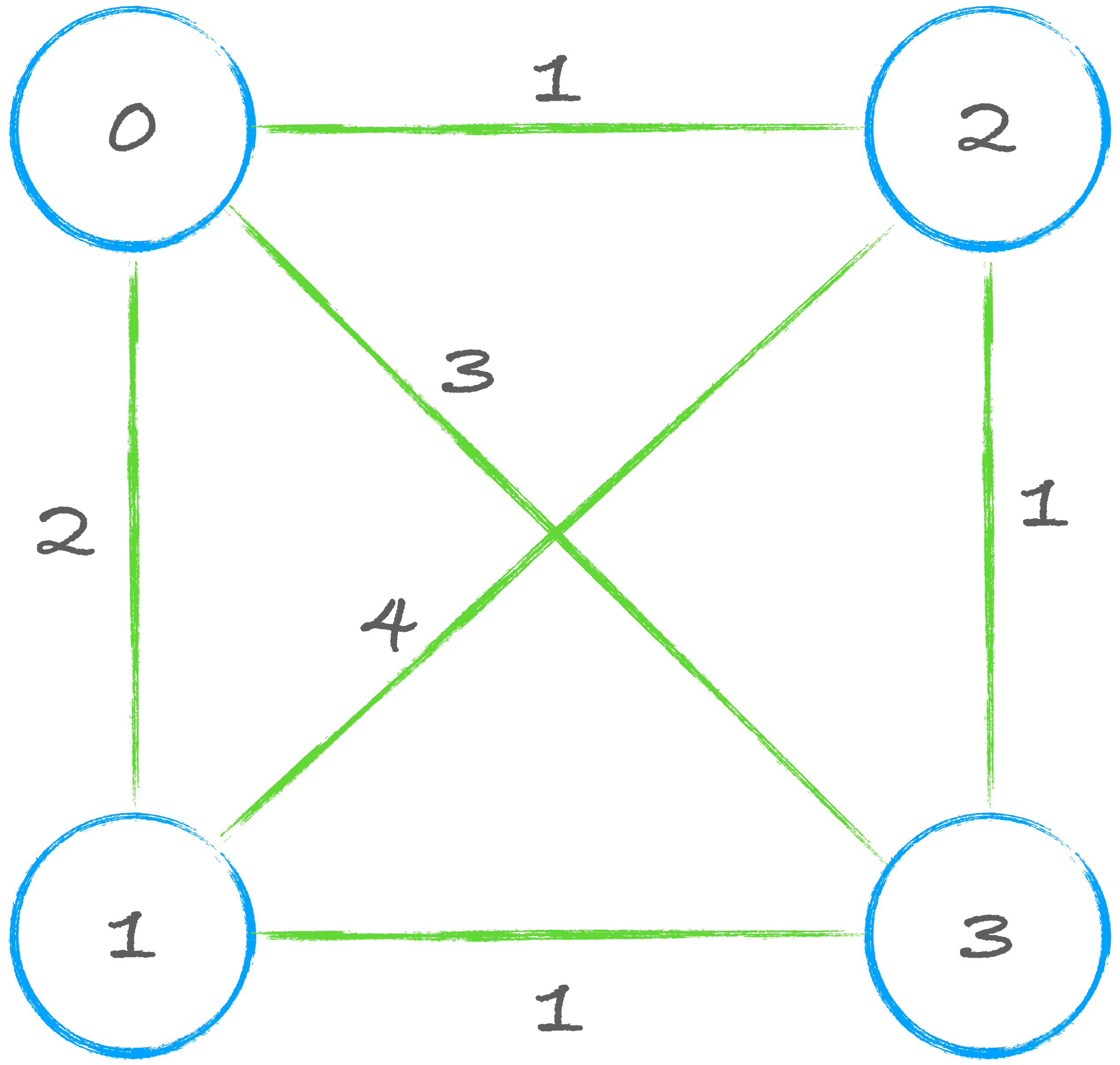
On doit trouver un chemin entre les villes

Comment parcourir une fois toutes les villes avec le minimum de distance, temps, coût...?

NP-Hard

QUBO pour les problèmes d'optimisation

TSP

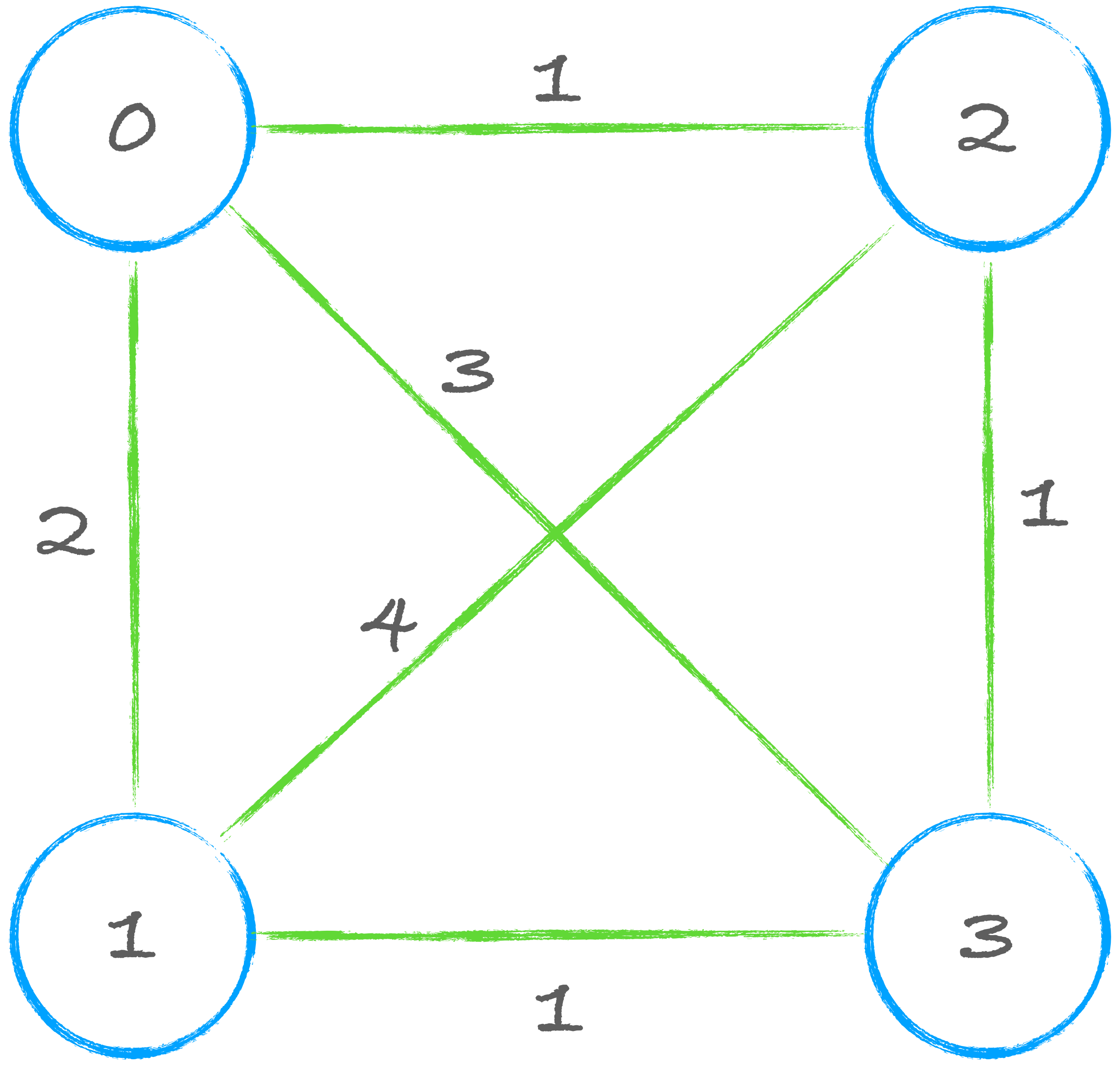


On définit x_{jl} comme des variables binaires qui représentent l'ordre de visite des différents noeuds

Si j est le l-ème noeud dans le graphe, $x_{jl} = 1$ et $x_{jh} = 0$ si $h \neq l$

QUBO pour les problèmes d'optimisation

TSP



On définit x_{jl} comme des variables binaires qui représentent l'ordre de visite des différents noeuds

Si j est le l-ème noeud dans le graphe, $x_{jl} = 1$ et $x_{jh} = 0$ si $h \neq l$

Contraintes

$$\sum_{l=0}^{m} x_{jl} = 1$$

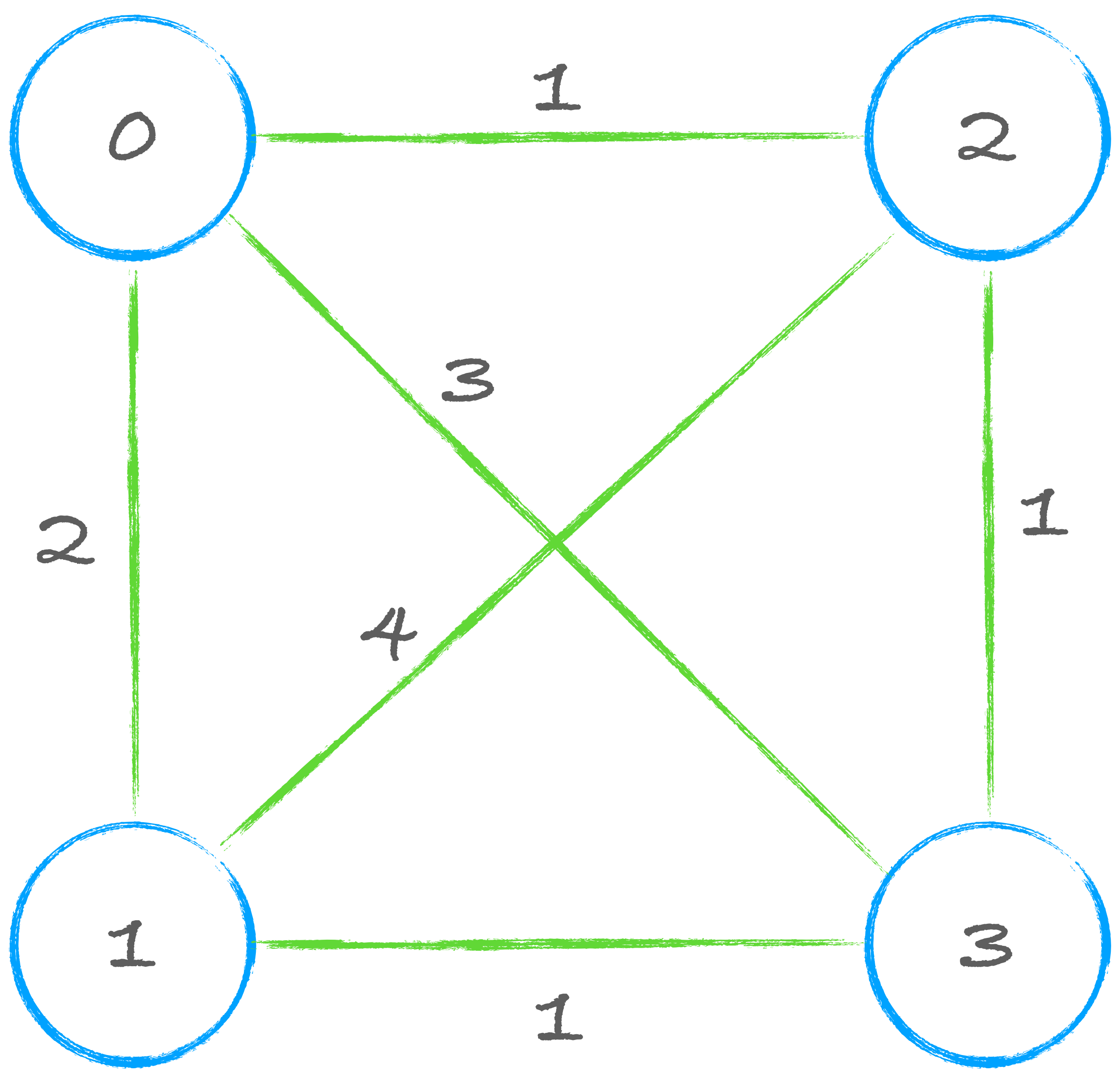
Chaque noeud ne peut être visité qu'une fois

$$\sum_{j=0}^{m} x_{jl} = 1$$

On ne peut visiter qu'une ville à la fois

QUBO pour les problèmes d'optimisation

TSP



On définit x_{jl} comme des variables binaires qui représentent l'ordre de visite des différents noeuds

Si j est le l-ème noeud dans le graphe, $x_{jl} = 1$ et $x_{jh} = 0$ si $h \neq l$

Contraintes

$$\sum_{l=0}^{m} x_{jl} = 1$$

Chaque noeud ne peut être visité qu'une fois

$$\sum_{j=0}^{m} x_{jl} = 1$$

On ne peut visiter qu'une ville à la fois

Parcourir les edges

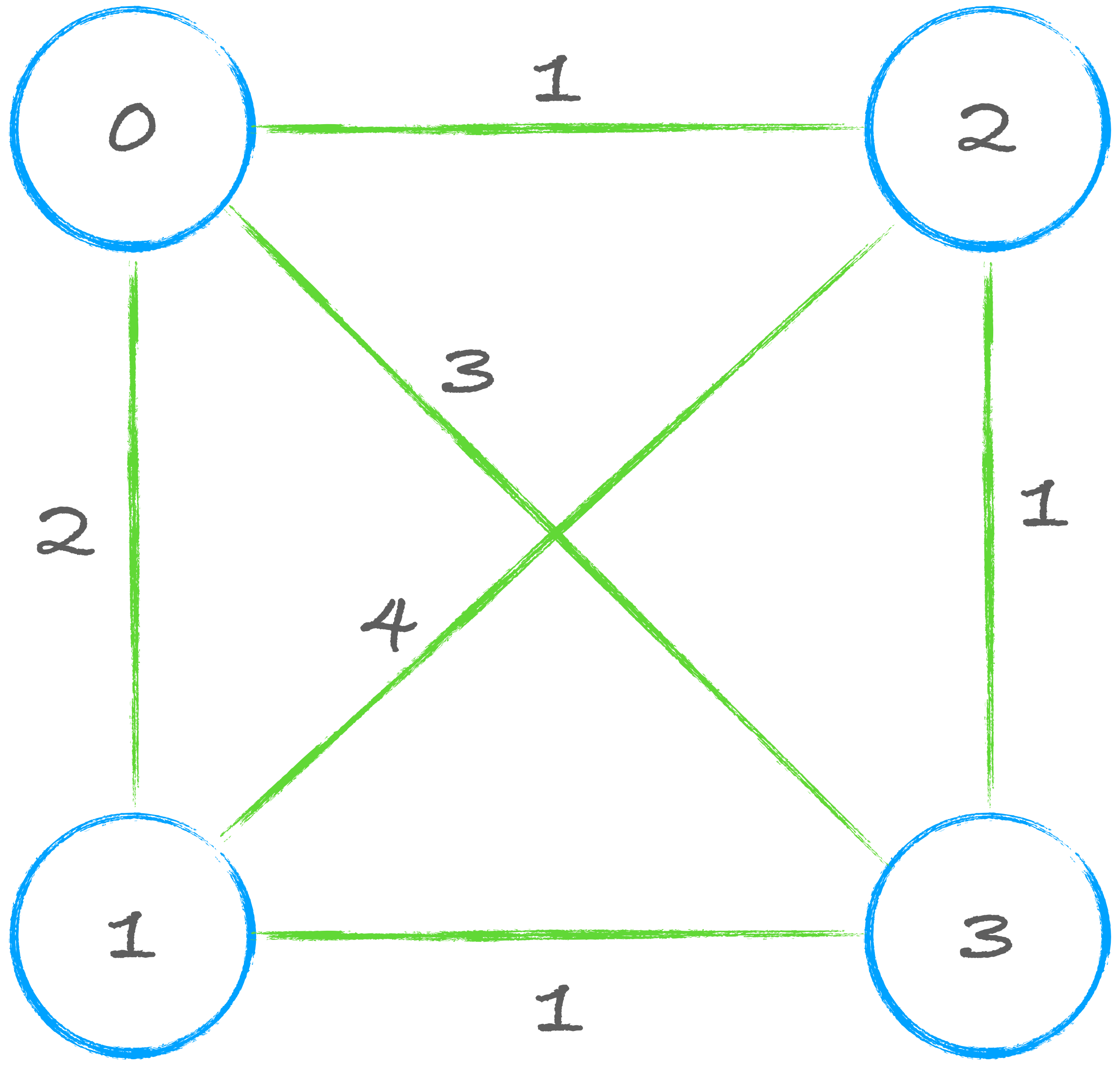
$$\sum_{l=0}^{m-1} \sum_{j=0}^{m} \sum_{k=0}^{m} w_{jk} x_{jl} x_{kl+1}$$

$w_{jj} = 0$ Rester dans la même ville n'a aucun coût

QUBO pour les problèmes d'optimisation

TSP

Formulation



QUBO

Minimiser $$\sum_{l=0}^{m-1}\sum_{j=0}^{m}\sum_{k=0}^{m} w_{jk}x_{jl}x_{kl+1} + B\left(\sum_{l=0}^{m} x_{jl} - 1\right)^2 + B\left(\sum_{j=0}^{m} x_{jl} - 1\right)^2$$

Avec $x_{jl} \in \{0,1\}, \quad j, l = 0,\ldots,m$

B doit être choisi pour que les solutions infaisables n'amènent jamais à une solution optimale

$$B = 1 + \sum_{l,k=0}^{m} x_{jk}$$

La pénalité est plus importante que le coût pour n'importe quelle configuration

QUBO pour les problèmes d'optimisation

TSP

Trouver la bonne formulation à un QUBO et comparer les formulations entre elles est un champ de recherche actif

Peut-on trouver un modèle d'Ising ou un QUBO générique pour la classe NP-Complete?

QUBO pour les problèmes d'optimisation

TSP

D'autres problèmes d'optimisation

- Quadratic Assignment Problems
- Capital Budgeting Problems
- Multiple Knapsack Problems
- Task Allocation Problems (distributed computer systems)
- Maximum Diversity Problems
- P-Median Problems
- Asymmetric Assignment Problems
- Symmetric Assignment Problems
- Side Constrained Assignment Problems
- Quadratic Knapsack Problems

- Constraint Satisfaction Problems (CSPs)
- Discrete Tomography Problems
- Set Partitioning Problems
- Set Packing Problems
- Warehouse Location Problems
- Maximum Clique Problems
- Maximum Independent Set Problems
- Maximum Cut Problems
- Graph Coloring Problems
- Number Partitioning Problems
- Linear Ordering Problems
- Clique Partitioning Problems
- SAT problems

QAOA

QAOA

Permet de discrétiser une évolution continue (quantum annealing) en petites étapes de changements discrets.

Trotterization, comme vu avec l'encodage Hamiltonien, est le processus de transformation d'une évolution continue à une évolution discrète.

QAOA permet d'approximer une solution optimale à des problèmes combinatoires.

QAOA

Permet de discrétiser une évolution continue (quantum annealing) en petites étapes de changements discrets.

Trotterization, comme vu avec l'encodage Hamiltonien, est le processus de transformation d'une évolution continue à une évolution discrète.

QAOA permet d'approximer une solution optimale à des problèmes combinatoires.

Quantum Approximate Optimization Algorithm

QAOA

Permet de discrétiser une évolution continue (quantum annealing) en petites étapes de changements discrets.

Trotterization, comme vu avec l'encodage Hamiltonien, est le processus de transformation d'une évolution continue à une évolution discrète.

QAOA permet d'approximer une solution optimale à des problèmes combinatoires.

Quantum Approximate Optimization Algorithm

Hamiltonien adiabatique
$$H(t) = A(t)H_0 + B(t)H_1$$

Avec H_0 et H_1 deux Hamiltoniens et A(t) et B(t) des fonctions qui puissent satisfaire A(0)=B(T)=1 et A(T)=B(0)=0 avec T le temps total du processus.

QAOA

Permet de discrétiser une évolution continue (quantum annealing) en petites étapes de changements discrets.

Trotterization, comme vu avec l'encodage Hamiltonien, est le processus de transformation d'une évolution continue à une évolution discrète.

QAOA permet d'approximer une solution optimale à des problèmes combinatoires.

Quantum Approximate Optimization Algorithm

Hamiltonien adiabatique

$$H(t) = A(t)H_0 + B(t)H_1$$

Avec H_0 et H_1 deux Hamiltoniens et A(t) et B(t) des fonctions qui puissent satisfaire A(0)=B(T)=1 et A(T)=B(0)=0 avec T le temps total du processus.

L'évolution d'un système quantique est gouvernée par l'équation de Schrödinger dépendante du temps La résoudre permet d'obtenir l'état du système à tout moments t entre 0 et T.

$$i\hbar\frac{d}{dt}|\Psi(t)\rangle = \hat{H}|\Psi(t)\rangle$$

QAOA

Hamiltonien adiabatique
$H(t) = A(t)H_0 + B(t)H_1$

$$H(t) = A(t)H_0 + B(t)H_1 \xrightarrow{\text{Discrétisation}} e^{i\Delta t(A(t_c)H_0+B(t_c)H_1)}$$

t_c est un temps fixe entre [0,T]

QAOA

Hamiltonien adiabatique
$H(t) = A(t)H_0 + B(t)H_1$

$$H(t) = A(t)H_0 + B(t)H_1 \xrightarrow{\text{Discrétisation}} e^{i\Delta t(A(t_c)H_0 + B(t_c)H_1)}$$

t_c est un temps fixe entre [0,T]

État quantique final:

$$\boxed{\left(\prod_{m=0}^{p} e^{i\Delta t(A(t_m)H_0 + B(t_m)H_1)}\right)|\psi_0\rangle}$$

Avec

m: le nombre de pas d'évolution

$t_m = m\dfrac{\Delta t}{T}$: le temps de chaque pas m

$p = \dfrac{T}{\Delta t}$: le nombre maximal de pas m

QAOA

Hamiltonien adiabatique
$H(t) = A(t)H_0 + B(t)H_1$

$$H(t) = A(t)H_0 + B(t)H_1 \xrightarrow{\text{Discrétisation}} e^{i\Delta t(A(t_c)H_0 + B(t_c)H_1)}$$

t_c est un temps fixe entre [0,T]

État quantique final: $\left(\prod_{m=0}^{p} e^{i\Delta t(A(t_m)H_0 + B(t_m)H_1)}\right)|\psi_0\rangle$

Application de portes
de rotations unitaires

Circuit quantique répété

Avec

m: le nombre de pas d'évolution

$t_m = m\frac{\Delta t}{T}$: le temps de chaque pas m

$p = \frac{T}{\Delta t}$: le nombre maximal de pas m

QAOA

Hamiltonien adiabatique
$H(t) = A(t)H_0 + B(t)H_1$

$$H(t) = A(t)H_0 + B(t)H_1 \xrightarrow{\text{Discrétisation}} e^{i\Delta t(A(t_c)H_0 + B(t_c)H_1)}$$

t_c est un temps fixe entre [0,T]

État quantique final:

$$\boxed{\left(\prod_{m=0}^{p} e^{i\Delta t(A(t_m)H_0 + B(t_m)H_1)}\right)|\psi_0\rangle}$$

Avec

m: le nombre de pas d'évolution

$t_m = m\dfrac{\Delta t}{T}$: le temps de chaque pas m

$p = \dfrac{T}{\Delta t}$: le nombre maximal de pas m

QAOA

Hamiltonien adiabatique
$H(t) = A(t)H_0 + B(t)H_1$

$H(t) = A(t)H_0 + B(t)H_1 \xrightarrow{\text{Discrétisation}} e^{i\Delta t(A(t_c)H_0 + B(t_c)H_1)}$ t_c est un temps fixe entre [0,T]

État quantique final:

$$\boxed{\left(\prod_{m=0}^{p} e^{i\Delta t(A(t_m)H_0 + B(t_m)H_1)}\right)|\psi_0\rangle} \xrightarrow[\approx]{\text{Nouvelle approximation}}$$

Avec

m: le nombre de pas d'évolution

$t_m = m\dfrac{\Delta t}{T}$: le temps de chaque pas m

$p = \dfrac{T}{\Delta t}$: le nombre maximal de pas m

QAOA

Hamiltonien adiabatique
$H(t) = A(t)H_0 + B(t)H_1$

$$H(t) = A(t)H_0 + B(t)H_1 \xrightarrow{\text{Discrétisation}} e^{i\Delta t(A(t_c)H_0 + B(t_c)H_1)}$$

t_c est un temps fixe entre [0,T]

État quantique final:

$$\boxed{\left(\prod_{m=0}^{p} e^{i\Delta t(A(t_m)H_0 + B(t_m)H_1)}\right)|\psi_0\rangle} \underset{\approx}{\xrightarrow{\text{Nouvelle approximation}}} \boxed{\left(\prod_{m=0}^{p} e^{i\Delta t A(t_m)H_0} e^{i\Delta t B(t_m)H_1}\right)|\psi_0\rangle}$$

Connue sous le nom de formule de Lie-Trotter

Avec

m: le nombre de pas d'évolution

$t_m = m\frac{\Delta t}{T}$: le temps de chaque pas m

$p = \frac{T}{\Delta t}$: le nombre maximal de pas m

QAOA

Hamiltonien adiabatique
$H(t) = A(t)H_0 + B(t)H_1$

$$H(t) = A(t)H_0 + B(t)H_1 \xrightarrow{\text{Discrétisation}} e^{i\Delta t(A(t_c)H_0 + B(t_c)H_1)}$$

t_c est un temps fixe entre [0,T]

État quantique final:

$$\left(\prod_{m=0}^{p} e^{i\Delta t(A(t_m)H_0 + B(t_m)H_1)}\right)|\psi_0\rangle \xrightarrow[\approx]{\text{Nouvelle approximation}} \left(\prod_{m=0}^{p} \boxed{e^{i\Delta t A(t_m)H_0}}\,\boxed{e^{i\Delta t B(t_m)H_1}}\right)|\psi_0\rangle$$

Avec

m: le nombre de pas d'évolution

$t_m = m\frac{\Delta t}{T}$: le temps de chaque pas m

$p = \frac{T}{\Delta t}$: le nombre maximal de pas m

Deux formes Hamiltoniennes d'évolution

↓

Forme de modèle d'Ising pour encoder le problème

L'objectif est de trouver le "ground state energy" à partir d'un circuit quantique

QAOA

Formulation digitale

$$\boxed{\left(\prod_{m=0}^{p} e^{i\Delta t A(t_m) H_0} e^{i\Delta t B(t_m) H_1}\right) |\psi_0\rangle} \longrightarrow$$

Posons

$$e^{i\Delta t A(t_m) H_0} = e^{i\gamma H_0}$$
$$e^{i\Delta t B(t_m) H_1} = e^{i\beta H_1}$$

QAOA

Formulation digitale

$$\left(\prod_{m=0}^{p} e^{i\Delta t A(t_m) H_0} e^{i\Delta t B(t_m) H_1}\right) |\psi_0\rangle$$

Posons

$$e^{i\Delta t A(t_m) H_0} = e^{i\gamma H_0}$$
$$e^{i\Delta t B(t_m) H_1} = e^{i\beta H_1}$$

On va venir appliquer p fois l'alternance de ces deux termes pour γ et β donnés

$$e^{i\gamma_p H_0} e^{i\beta_p H_1} \dots e^{i\gamma_2 H_0} e^{i\beta_2 H_1} e^{i\gamma_1 H_0} e^{i\beta_1 H_1} |\psi_0\rangle$$

QAOA

Formulation digitale

$$\left(\prod_{m=0}^{p} e^{i\Delta t A(t_m) H_0} e^{i\Delta t B(t_m) H_1}\right) |\psi_0\rangle$$

Posons

$$e^{i\Delta t A(t_m) H_0} = e^{i\gamma H_0}$$
$$e^{i\Delta t B(t_m) H_1} = e^{i\beta H_1}$$

On va venir appliquer p fois l'alternance de ces deux termes pour γ et β donnés

$$e^{i\gamma_p H_0} e^{i\beta_p H_1} \dots e^{i\gamma_2 H_0} e^{i\beta_2 H_1} e^{i\gamma_1 H_0} e^{i\beta_1 H_1} |\psi_0\rangle$$

Peuvent être implanté avec des portes à 1 et 2 qubits

Nombres réels

$$\gamma = (\gamma_1, \dots, \gamma_p)$$
$$\beta = (\beta_1, \dots, \beta_p)$$

$$|\beta, \gamma\rangle$$

QAOA

Formulation digitale

$$\left(\prod_{m=0}^{p} e^{i\Delta t A(t_m) H_0} e^{i\Delta t B(t_m) H_1}\right) |\psi_0\rangle \longrightarrow e^{i\gamma_p H_0} e^{i\beta_p H_1} \dots e^{i\gamma_2 H_0} e^{i\beta_2 H_1} e^{i\gamma_1 H_0} e^{i\beta_1 H_1} |\psi_0\rangle$$

$$\begin{matrix} \gamma = (\gamma_1, \dots, \gamma_p) \\ \beta = (\beta_1, \dots, \beta_p) \end{matrix} \longrightarrow |\beta, \gamma\rangle$$

On va chercher à trouver la valeur de plus faible énergie pour H_1 en faisant:

$$\langle \beta, \gamma | H_1 | \beta, \gamma \rangle$$

On va donc passer d'un problème d'optimisation à un problème combinatoire
Pour déterminer les meilleures valeurs de β et γ

Simple estimation de valeurs réelles pour un output réel.
Le problème est que $|\beta, \gamma\rangle$ grossit exponentiellement et ne peut pas être simulé classiquement

QAOA

Formulation digitale

$$\left(\prod_{m=0}^{p} e^{i\Delta t A(t_m) H_0} e^{i\Delta t B(t_m) H_1}\right) |\psi_0\rangle$$

$$e^{i\gamma_p H_0} e^{i\beta_p H_1} \dots e^{i\gamma_2 H_0} e^{i\beta_2} \dots H_1 |\psi_0\rangle$$

$\gamma = (\gamma_1, \dots$

$\beta = (\beta$

$|\beta, \gamma\rangle$

On va chercher à trouver la valeur de plus faible én… H_1 en faisant:

$$\langle \beta, \gamma | H_1 | \beta, \gamma \rangle$$

On va donc passer d'un problème d'opti… un problème combinatoire

Pour déterminer les m… valeurs de β et γ

Simple estimation … réelles pour un output réel.

Le problème est que $|\beta, \gamma\rangle$ grossit expone…llement et ne peut pas être simulé classiquement

Efficace avec un ordinateur quantique si

H a un nombre de terme qui grossit polynomialement

QAOA

Formulation digitale

$$\left(\prod_{m=0}^{p} e^{i\Delta t A(t_m) H_0} e^{i\Delta t B(t_m) H_1}\right) |\psi_0\rangle \longrightarrow e^{i\gamma_p H_0} e^{i\beta_p H_1} \dots e^{i\gamma_2 H_0} e^{i\beta_2 H_1} e^{i\gamma_1 H_0} e^{i\beta_1 H_1} |\psi_0\rangle$$

Algorithme hybride qui a besoin de l'info classique pour calculer la fonction de coût et les nouveaux paramètres β et γ par descente de gradient

L'ordinateur quantique sera ensuite utilisé pour créer l'état quantique en utilisant les valeurs β^* et γ^* qui sont les valeurs approximées pour le niveau d'énergie le plus bas. Cet état a le maximum d'overlap avec H_1 et donc le bitstring final est la valeur des Coefficients du problème encodé dans H_1

0100110010011

QAOA

Algorithme

Choix de la valeur de p

Choix de valeurs d'initialisation de $\beta = (\beta_1, \dots, \beta_p)$ et $\gamma = (\gamma_1, \dots, \gamma_p)$

while critère d'arrêt non atteint:

Prépare l'état $|\beta, \gamma\rangle$ — Cette étape se fait sur un QC

Mesure de $|\beta, \gamma\rangle$ et estimation de $E(\beta, \gamma)$

Mise à niveau des paramètres $|\beta, \gamma\rangle$ grâce à un algorithme de minimisation

Optention des valeurs optimales de β^* et γ^*

Prépare l'état $|\beta^*, \gamma^*\rangle$ — Cette étape se fait sur un QC

Mesure de $|\beta^*, \gamma^*\rangle$ et obtention de $E(\beta^*, \gamma^*)$

QAOA

Algorithme

Classique	Choix de la valeur de p
Classique	Choix de valeurs d'initialisation de $\beta = (\beta_1, \ldots, \beta_p)$ et $\gamma = (\gamma_1, \ldots, \gamma_p)$
Classique	while critère d'arrêt non atteint:
Quantique	Prépare l'état $\lvert\beta, \gamma\rangle$ — Cette étape se fait sur un QC
Classique	Mesure de $\lvert\beta, \gamma\rangle$ et estimation de $E(\beta, \gamma)$
Classique	Mise à niveau des paramètres $\lvert\beta, \gamma\rangle$ grâce à un algorithme de minimisation
Classique	Optention des valeurs optimales de $\beta*$ et $\gamma*$
Quantique	Prépare l'état $\lvert\beta*, \gamma*\rangle$ — Cette étape se fait sur un QC
Classique	Mesure de $\lvert\beta*, \gamma*\rangle$ et obtention de $E(\beta*, \gamma*)$

QAOA

Algorithme

En pratique

$$|\beta,\gamma\rangle = e^{i\gamma_p H_0} e^{i\beta_p H_1} \ldots e^{i\gamma_2 H_0} e^{i\beta_2 H_1} e^{i\gamma_1 H_0} e^{i\beta_1 H_1} \boxed{|\psi_0\rangle}$$

État d'énergie fondamentale de H_0

Préparé dans l'état $|+\rangle$ avec des portes de Hadamard sur tous les qubits

QAOA

Algorithme

En pratique

$$|\beta,\gamma\rangle = e^{i\gamma_p H_0} e^{i\beta_p H_1} \ldots e^{i\gamma_2 H_0} e^{i\beta_2 H_1} e^{i\gamma_1 H_0} e^{i\beta_1 H_1} \boxed{|\psi_0\rangle}$$

État d'énergie fondamentale de H_0

Préparé dans l'état $|+\rangle$ avec des portes de Hadamard sur tous les qubits

Pauli x

$$H_0 = -\sum_{j=0}^{n-1} X_j \longrightarrow R_X(2\beta)$$

Application de la porte sur tous les qubits

QAOA

Algorithme

En pratique

$$|\beta, \gamma\rangle = e^{i\gamma_p H_0} e^{i\beta_p H_1} \ldots e^{i\gamma_2 H_0} e^{i\beta_2 H_1} e^{i\gamma_1 H_0} e^{i\beta_1 H_1} \boxed{|\psi_0\rangle}$$

État d'énergie fondamentale de H_0

Préparé dans l'état $|+\rangle$ avec des portes de Hadamard sur tous les qubits

Pauli x $\quad H_0 = -\sum_{j=0}^{n-1} X_j \longrightarrow R_X(2\beta)$ Application de la porte sur tous les qubits

Ising $\quad H_1 = -\sum_{j,k} J_{jk} Z_j Z_k - \sum_j h_j Z_j \longrightarrow R_Z(2\gamma)$ Et CNOTs entre les qubits j,k

J_{jk}, h_j sont des nombres réels

QAOA

Algorithme

En pratique

$$|\beta,\gamma\rangle = e^{i\gamma_p H_0} e^{i\beta_p H_1} \ldots e^{i\gamma_2 H_0} e^{i\beta_2 H_1} e^{i\gamma_1 H_0} e^{i\beta_1 H_1} \boxed{|\psi_0\rangle}$$

État d'énergie fondamentale de H_0

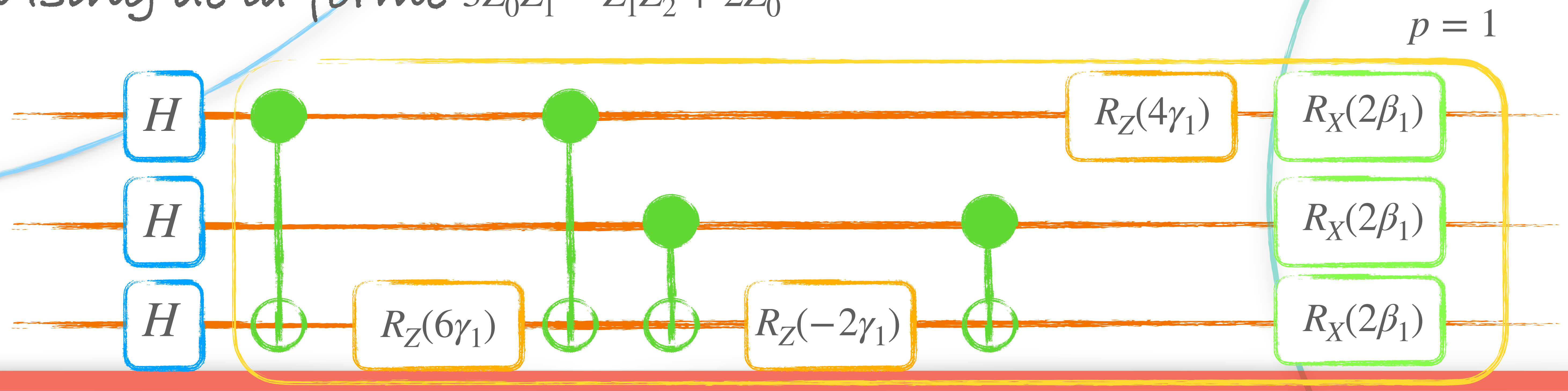
Préparé dans l'état $|+\rangle$ avec des portes de Hadamard sur tous les qubits

Pauli x $\quad H_0 = -\sum_{j=0}^{n-1} X_j \longrightarrow R_X(2\beta)$ Application de la porte sur tous les qubits

Ising $\quad H_1 = -\sum_{j,k} J_{jk} Z_j Z_k - \sum_j h_j Z_j \longrightarrow R_Z(2\gamma)$ Et CNOTs entre les qubits j,k

J_{jk}, h_j sont des nombres réels

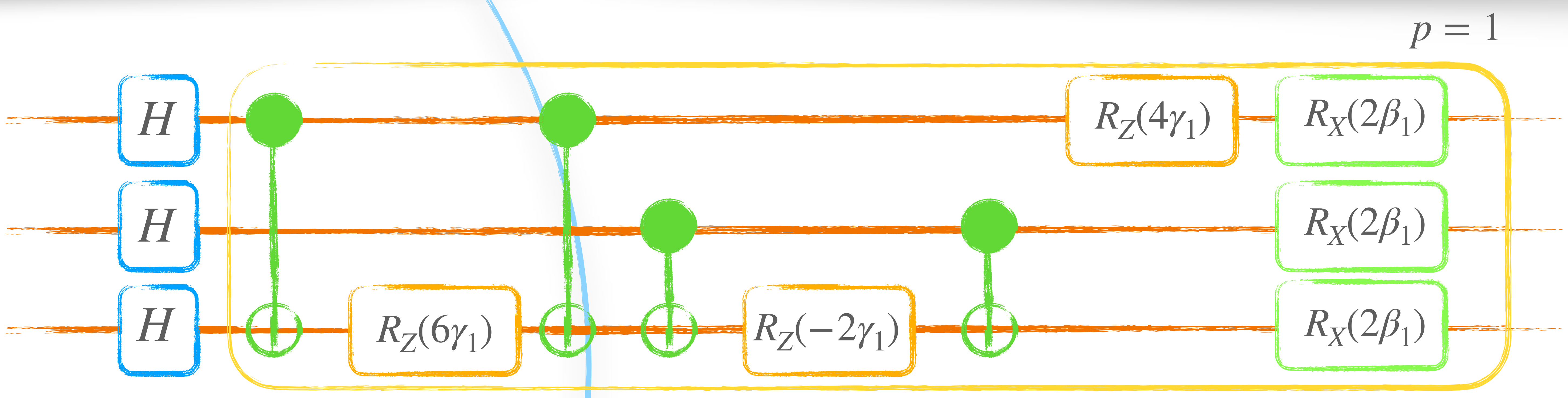
Pour 3 qubits, avec un Ising de la forme $3Z_0Z_1 - Z_1Z_2 + 2Z_0$



QAOA

Algorithme

Valeur estimées



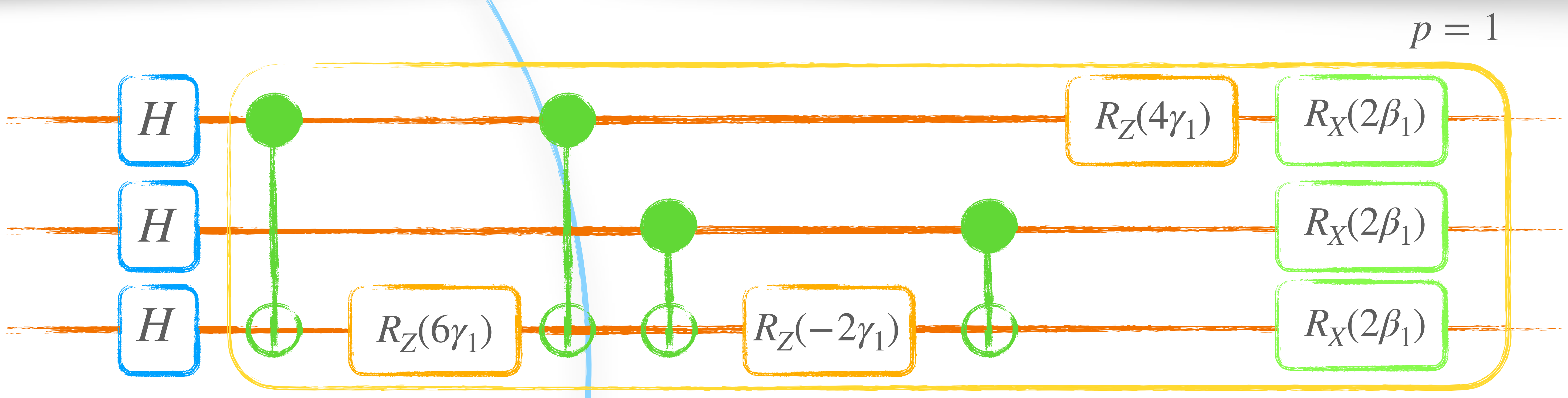
Pour 3 qubits, avec un Ising de la forme $3Z_0Z_1 - Z_1Z_2 + 2Z_0$

On veut directement estimer $\langle x | H_1 | x \rangle$

QAOA

Algorithme

Valeur estimées



Pour 3 qubits, avec un Ising de la forme $3Z_0Z_1 - Z_1Z_2 + 2Z_0$

On veut directement estimer $\langle x | H_1 | x \rangle$

Rappels
$\langle x | Z_j | x \rangle = 1$ si $x = 0$

$\langle x | Z_j | x \rangle = -1$ Le reste du temps

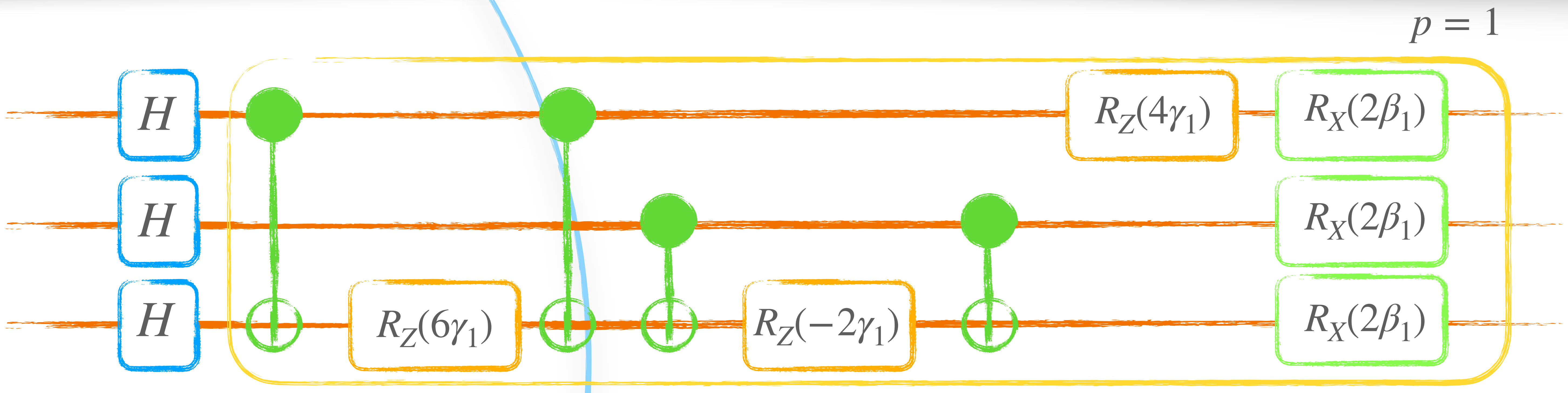
$\langle x | Z_j Z_k | x \rangle = 1$ si $x_j = x_k$

$\langle x | Z_j Z_k | x \rangle = -1$ si $x_j \neq x_k$

QAOA

Algorithme

Valeur estimées



Pour 3 qubits, avec un Ising de la forme $3Z_0Z_1 - Z_1Z_2 + 2Z_0$

On veut directement estimer $\langle x | H_1 | x \rangle$

Rappels $\langle x | Z_j | x \rangle = 1$ si x =0

$\langle x | Z_j | x \rangle = -1$ Le reste du temps

$\langle x | Z_jZ_k | x \rangle = 1$ si $x_j = x_k$

$\langle x | Z_jZ_k | x \rangle = -1$ si $x_j \neq x_k$

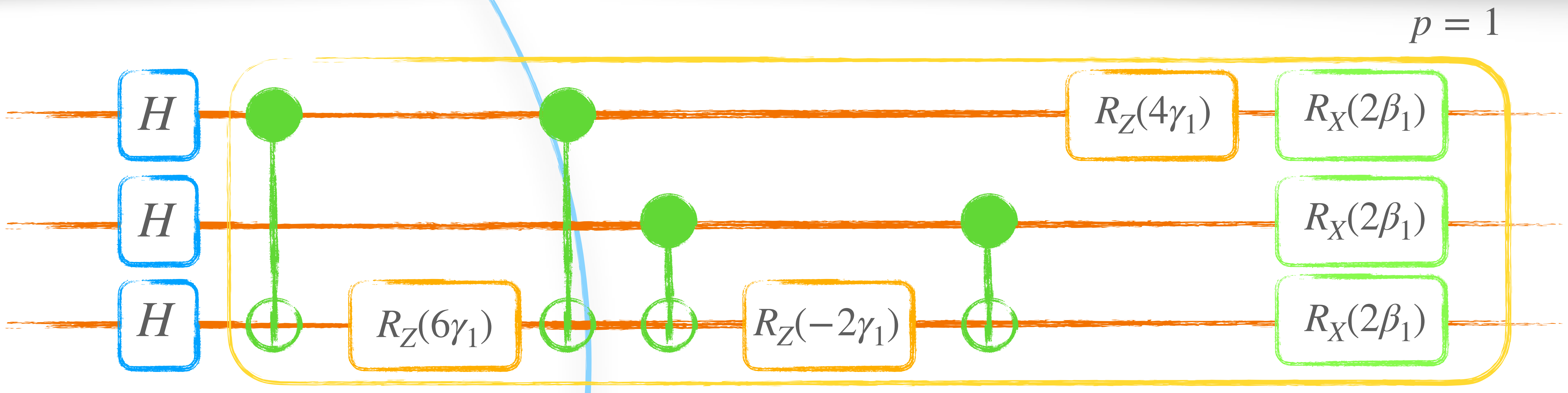
Pour l'état $|101\rangle$ $\langle 101 | H_1 | 101 \rangle = 3\langle 101 | Z_0Z_2 | 101 \rangle - \langle 101 | Z_1Z_2 | 101 \rangle + 2\langle 101 | Z_0 | 101 \rangle = 3 + 1 - 2 = 2$

Pour l'état $|100\rangle$ $\langle 100 | H_1 | 100 \rangle = 3\langle 100 | Z_0Z_2 | 100 \rangle - \langle 100 | Z_1Z_2 | 100 \rangle + 2\langle 100 | Z_0 | 100 \rangle = ?$

QAOA

Algorithme

Valeur estimées



Pour 3 qubits, avec un Ising de la forme $3Z_0Z_1 - Z_1Z_2 + 2Z_0$

On veut directement estimer $\langle x | H_1 | x \rangle$

Rappels $\langle x | Z_j | x \rangle = 1$ si x =0

$\langle x | Z_j | x \rangle = -1$ Le reste du temps

$\langle x | Z_jZ_k | x \rangle = 1$ si $x_j = x_k$

$\langle x | Z_jZ_k | x \rangle = -1$ si $x_j \neq x_k$

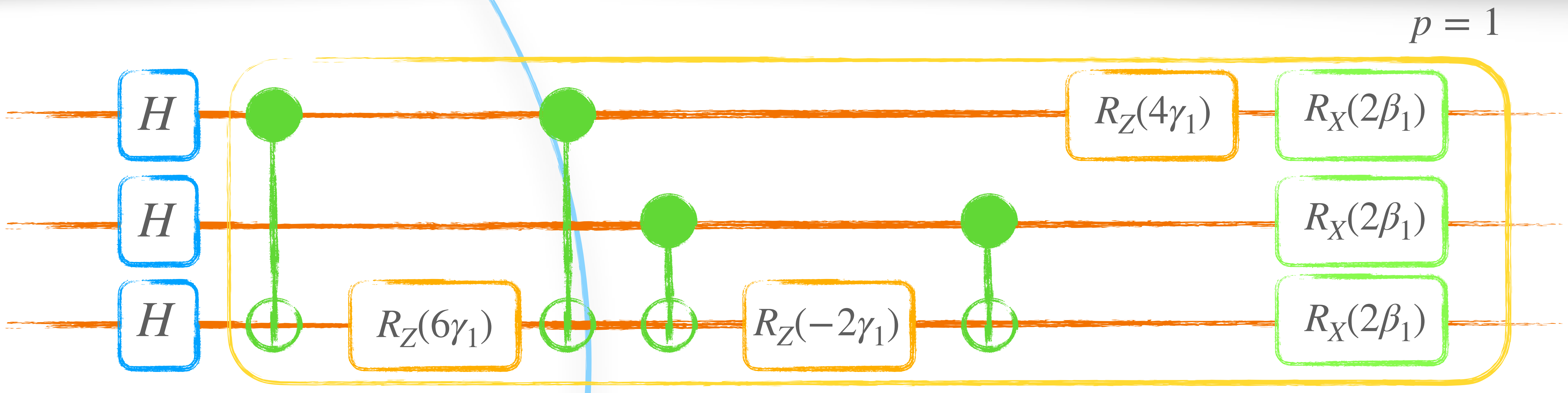
Pour l'état $|101\rangle$ $\langle 101 | H_1 | 101 \rangle = 3\langle 101 | Z_0Z_2 | 101 \rangle - \langle 101 | Z_1Z_2 | 101 \rangle + 2\langle 101 | Z_0 | 101 \rangle = 3 + 1 - 2 = 2$

Pour l'état $|100\rangle$ $\langle 100 | H_1 | 100 \rangle = 3\langle 100 | Z_0Z_2 | 100 \rangle - \langle 100 | Z_1Z_2 | 100 \rangle + 2\langle 100 | Z_0 | 100 \rangle = -3 - 1 - 2 = -6$

QAOA

Algorithme

Valeur estimées



Pour 3 qubits, avec un Ising de la forme $3Z_0Z_1 - Z_1Z_2 + 2Z_0$

On veut directement estimer $\langle x | H_1 | x \rangle$

Rappels $\langle x | Z_j | x \rangle = 1$ si x =0

$\langle x | Z_j | x \rangle = -1$ Le reste du temps

$\langle x | Z_jZ_k | x \rangle = 1$ si $x_j = x_k$

$\langle x | Z_jZ_k | x \rangle = -1$ si $x_j \neq x_k$

Forme générale pour sur un QC $\langle \beta, \gamma | H_1 | \beta, \gamma \rangle \approx \sum_x \frac{m_x}{M} \langle x | H_1 | x \rangle$ pour M mesure du système

Références

[1] Combarro E. F. & González-Castillo S., 2023, A Practical Guide to Quantum Machine Learning and Quantum Optimization, Chapitres 3, 5

[2] Karp R. M., 1972, Reducibility among combinatorial problems, Springer, 85-103, https://cgi.di.uoa.gr/~sgk/teaching/grad/handouts/karp.pdf

[3] https://www.hackerearth.com/practice/notes/the-knapsack-problem/