

# Compléments sur QAOA

Dimitri Watel

November 17, 2025

Dans ce document, on revient sur quelques détails de l'algorithme QAOA, comment l'utiliser dans un cas d'un problème d'optimisation.

**Equation de Schrödinger** Un système quantique est défini par une matrice nommée *Hamiltonien* selon l'équation de Schrödinger:

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

Si  $H$  ne dépend pas du temps, alors une solution à cette équation est

$$|\varphi(t)\rangle = \exp\left(-i \frac{t}{\hbar} H\right) |\varphi(0)\rangle \quad (1)$$

où  $\exp A = \sum_{k=0}^{+\infty} \frac{A^k}{k!}$ .

**Etat stable** Un état stable d'un système est représenté par un vecteur propre  $|\phi\rangle$  de la matrice  $H$ . Ce vecteur est associé à la valeur propre  $\lambda$  la plus petite de  $H$ .

Pour répondre à la question posée en cours,  $H$  ne peut avoir que des valeurs propres réelles car un hamiltonien a la particularité d'être une matrice hermitienne. C'est à dire une matrice qui est sa propre matrice adjointe:  $H = H^\dagger$ . Il ne s'agit pas donc de n'importe quel matrice. (Source : <https://quantumcomputing.stackexchange.com/>...)

Notez bien que:

- un hamiltonien n'est pas forcément unitaire.
- toutes les matrices unitaires ne sont pas hermitienne
- puisque  $H$  est hermitienne, on peut la décomposer en portes de Paulis (donc somme de produits tensoriels de matrices  $X, Y, Z$  et  $I_2$ ).

**Théorème adiabatique** Le théorème adiabatique énonce que, si un système régit par un état  $H$  est dans un état stable de  $H$  et qu'on modifie  $H$  en un hamiltonien très proche  $H'$  alors le système va évoluer dans un état stable de  $H'$ . QAOA tente de reproduire une telle évolution de l'hamiltonien.

Supposons qu'on dispose d'un état stable  $|\varphi_B\rangle$  d'un hamiltonien  $H_B$ . Supposons qu'on dispose également d'un hamiltonien  $H_C$ . Alors en faisant évoluer doucement  $H_B$  vers  $H_C$ ,  $|\varphi_B\rangle$  devrait évoluer vers un état stable  $|\varphi_C\rangle$  de  $H_C$ .

Pour l'évolution de  $H$ , on considère généralement la suivante:

$$H(t) = (1 - t)H_B + tH_C$$

Pour décrire l'évolution de  $|\varphi(t)\rangle$  au cours du temps, on ne peut pas utiliser la formule (1) décrite précédemment car  $H(t)$  dépend du temps. Pour s'en sortir, on va discrétiser le temps.

On considère un pas de temps  $dt$  très petit et on suit le processus suivant:

- à l'instant 0, on met le système dans l'état  $|\varphi_B\rangle$  avec l'hamiltonien  $H(0) = H_B$ . On laisse le système évoluer pendant un instant  $dt$ .
- à l'instant  $dt$ , on change l'hamiltonien et on le remplace par  $H(dt)$ . On laisse le système évoluer pendant un instant  $dt$ .
- ...
- à l'instant  $j \cdot dt$ , on change l'hamiltonien et on le remplace par  $H(j \cdot dt)$ . On laisse le système évoluer pendant un instant  $dt$ .
- On poursuit jusqu'à ce que  $j \cdot dt = 1$ .

Dans quel état est le système à l'instant  $t$  ? Cette fois, on peut utiliser l'équation (1) car, entre les instants  $j \cdot dt$  et  $(j + 1) \cdot dt$ , le système possède un hamiltonien constant.

$$|\varphi(j \cdot dt + t)\rangle = \exp\left(-i \frac{t}{\hbar} H(j \cdot dt)\right) |\varphi(j \cdot dt)\rangle$$

$$|\varphi((j + 1) \cdot dt)\rangle = |\varphi(j \cdot dt + dt)\rangle = \exp\left(-i \frac{dt}{\hbar} H(j \cdot dt)\right) |\varphi(j \cdot dt)\rangle$$

Donc on peut calculer  $|\varphi(1)\rangle = |\varphi_C\rangle$  recursivement. Cette méthode suppose que  $dt$  est très petit pour que l'évolution discrète soit la plus proche possible de l'évolution continue entre  $H_B$  et  $H_C$ .

**Circuit de QAOA** On peut donc simuler le théorème adiabatique avec le circuit suivant:

$$|\varphi_B\rangle \rightarrow \boxed{\exp\left(-i \frac{dt}{\hbar} H(0 \cdot dt)\right)} \rightarrow \boxed{\exp\left(-i \frac{dt}{\hbar} H(1 \cdot dt)\right)} \rightarrow \cdots \rightarrow \boxed{\exp\left(-i \frac{dt}{\hbar} H(t \cdot dt)\right)} \simeq |\varphi_C\rangle$$

Il reste donc à ce stade quelques questions en suspens:

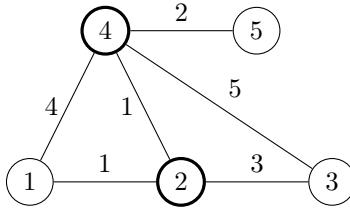
1. Est-ce qu'atteindre  $|\varphi_C\rangle$  peut être utile pour résoudre un problème d'optimisation ?
2. Que choisir pour  $H_B$  et  $|\varphi_B\rangle$  ?
3. Comment construire la porte  $\exp\left(-i\frac{dt}{\hbar}H(j \cdot dt)\right)$  ? (Et accessoirement, pourquoi est-ce que cette matrice est unitaire alors que  $H(h \cdot dt)$  ne l'est pas forcément).
4. Si on regarde bien dans la littérature, on voit que QAOA est un algorithme paramétré (ou on dit qu'il utilise des Ansatz). Où interviennent les paramètres ?
5. Comment savoir si on a effectivement atteint un état stable à la fin du circuit ?

**Optimiser avec  $|\varphi_C\rangle$ .** Un problème qu'il est possible de résoudre avec cette technique est Maximum-Cut. C'est un exemple classique qu'on retrouve dans beaucoup de documents traitant de QAOA car il est très simple de transformer une entrée de ce problème en un hamiltonien  $H_C$  dont on souhaite un état stable.

Le problème Maximum-Cut consiste, connaissant un graphe  $G = (V, E)$  non orienté, et des poids  $\omega(e)$  sur chaque arête  $e$ , à trouver une partition  $V_1, V_2$  de  $V$ . Une telle partition est nommée une coupe de  $G$ . Pour chaque coupe, on définit  $C(V_1, V_2)$  comme la somme des poids des arêtes reliant  $V_1$  et  $V_2$ . On ne compte pas les arêtes reliant deux sommets de  $V_1$  et celles reliant deux sommets de  $V_2$ .

$$C(V_1, V_2) = \sum_{\substack{(u,v) \in E \\ u \in V_1 \\ v \in V_2}} \omega(u, v)$$

On souhaite renvoyer la coupe de poids maximum. Par exemple dans le graphe suivant, la coupe  $(\{2, 4\}, \{1, 3, 5\})$  a un poids 15, seule l'arête  $(2, 4)$  n'est pas comptée dans la coupe.



On peut montrer qu'il existe un hamiltonien  $H_C$  dont les vecteurs propres encodent les coupes de  $G$  telle que la valeur propre associée à une coupe est

l'opposé de son poids dans le graphe, ainsi trouver une coupe de poids maximum revient à trouver un état stable de  $H_C$ . Pour cela, on peut commencer par réécrire max-Cut sous forme d'un polynôme à minimiser. Trouver la coupe de poids maximum est équivalent à choisir, pour chaque sommet  $u \in V$  une valeur binaire  $\{0, 1\}$ , 0 indiquant qu'on met  $u$  dans  $V_1$  et 1 indiquant qu'on le met dans  $V_2$ , de sorte à minimiser:

$$f(x) = \sum_{u,v \in E} (-1 \cdot \omega(u, v)) (x_u \cdot (1 - x_v) + x_v \cdot (1 - x_u))$$

En effet, si  $x_u$  et  $x_v$  sont du même côté, alors  $x_u = x_v$  et  $x_u \cdot (1 - x_v) + x_v \cdot (1 - x_u) = 0$ . Par contre si  $x_u \neq x_v$  alors  $x_u \cdot (1 - x_v) + x_v \cdot (1 - x_u) = 1$ . En encodant notre coupe  $V_1, V_2$  avec  $x$ , on a bien  $f(x) = -C(V_1, V_2)$ .

On effectue un changement de variable pour écrire  $f$  sous forme d'un polynôme avec des variables  $(-1, 1)$ . Connaissant  $x_u$ , on va le transformer en  $z_u = \frac{1+x_u}{2}$ . Ainsi si  $x_u = 0$  alors  $z_u = -1$  et sinon  $z_u = 1$ .

$$x_u \cdot (1 - x_v) + x_v \cdot (1 - x_u) = \frac{1}{2}(1 - z_u \cdot z_v)$$

Donc

$$f(z) = \sum_{u,v \in E} \left( \frac{-1}{2} \cdot \omega(u, v) \right) (1 - z_u \cdot z_v)$$

On transforme enfin  $f$  en une matrice  $H_C$  en remplaçant  $z_u \cdot z_v$  par les portes  $Z_u \cdot Z_v$  où  $Z_i$  est le circuit où on place une porte  $Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$  sur la  $i$ -ième ligne (et des identités ailleurs). Par exemple, dans le graphe précédent,  $Z_1 \cdot Z_4$  est la porte  $Z \otimes I_2 \otimes I_2 \otimes Z \otimes I_2$ .

$$H_C = \sum_{u,v \in E} \left( \frac{-1}{2} \cdot \omega(u, v) \right) (I_{2^n} - Z_u \cdot Z_v)$$

Donc, pour le graphe donné précédemment en exemple, on aurait

$$\begin{aligned} H_C &= \frac{-1}{2} \cdot (I_{2^n} - Z_1 \cdot Z_2) \\ &+ \frac{-4}{2} \cdot (I_{2^n} - Z_1 \cdot Z_4) \\ &+ \frac{-3}{2} \cdot (I_{2^n} - Z_2 \cdot Z_3) \\ &+ \frac{-1}{2} \cdot (I_{2^n} - Z_2 \cdot Z_4) \\ &+ \frac{-5}{2} \cdot (I_{2^n} - Z_3 \cdot Z_4) \\ &+ \frac{-2}{2} \cdot (I_{2^n} - Z_4 \cdot Z_5) \end{aligned}$$

Cette matrice n'est pas unitaire. Elle est une somme de matrices unitaires. On peut maintenant regarder un vecteur propre de cette matrice. Puisque  $I_2$  et  $Z$  sont des matrices diagonales alors un produit tensoriel de  $Z$  et de  $I_2$  est aussi diagonal. Cela signifie que les vecteurs propres de cette matrice sont les vecteurs de base et la valeur propre d'un tel vecteur est juste la valeur de la diagonale à la ligne correspondant au vecteur de base (la 1e pour  $|0\rangle$ , la 2e pour  $|1\rangle$ , ...).

Soit l'arête  $(u, v) \in E$ . Considérons un vecteur de base  $|\underline{x}\rangle$  où le  $u$ -ieme bit et le  $v$ -ieme bits sont respectivement:

- $x_u = x_v = 0$ . Dans ce cas,  $Z_u \cdot Z_v |\underline{x}\rangle$  revient à

$$|\underline{x}\rangle \left\{ \begin{array}{c} |x_u\rangle = |0\rangle \text{ --- } \boxed{Z} \text{ --- } |0\rangle \\ \vdots \\ |x_v\rangle = |0\rangle \text{ --- } \boxed{Z} \text{ --- } |0\rangle \end{array} \right\} |\underline{x}\rangle$$

- $x_u = x_v = 1$ . Dans ce cas,  $Z_u \cdot Z_v |\underline{x}\rangle$  revient à

$$|\underline{x}\rangle \left\{ \begin{array}{c} |x_u\rangle = |1\rangle \text{ --- } \boxed{Z} \text{ --- } -1 \cdot |1\rangle \\ \vdots \\ |x_v\rangle = |1\rangle \text{ --- } \boxed{Z} \text{ --- } -1 \cdot |1\rangle \end{array} \right\} |\underline{x}\rangle$$

- $x_u = 1, x_v = 0$ . Dans ce cas,  $Z_u \cdot Z_v |\underline{x}\rangle$  revient à

$$|\underline{x}\rangle \left\{ \begin{array}{c} |x_u\rangle = |1\rangle \text{ --- } \boxed{Z} \text{ --- } -1 \cdot |1\rangle \\ \vdots \\ |x_v\rangle = |0\rangle \text{ --- } \boxed{Z} \text{ --- } |0\rangle \end{array} \right\} -1 \cdot |\underline{x}\rangle$$

Donc

$$(I_{2^n} - Z_u \cdot Z_v) |\underline{x}\rangle = \begin{cases} 0 \cdot |\underline{x}\rangle & \text{si } x_u = x_v \\ 2 \cdot |\underline{x}\rangle & \text{sinon} \end{cases}$$

Donc

$$\left( \frac{-1}{2} \cdot \omega(u, v) \right) (I_{2^n} - Z_u \cdot Z_v) |\underline{x}\rangle = \begin{cases} 0 \cdot |\underline{x}\rangle & \text{si } x_u = x_v \\ -\omega(u, v) \cdot |\underline{x}\rangle & \text{sinon} \end{cases}$$

Ainsi, si  $H_C |\underline{x}\rangle = f(x) |\underline{x}\rangle$ , la valeur propre de  $|\underline{x}\rangle$  est l'opposé du poids de la coupe encodée par  $x$ .

**Pourquoi est-ce intéressant ?** Maintenant qu'on a cette association, on voit qu'en utilisant le théorème adiabatique, on peut faire tendre un état quantique vers une solution optimale d'une instance de Max-Cut. Et donc avec QAOA, on devrait pouvoir reproduire ce phénomène. Autre détail intéressant, Max-Cut est un problème dit *NP-Complet*. Cela veut dire

- qu'on ne sait pas le résoudre en temps polynomial à l'heure actuelle (et on pense que c'est impossible). QAOA est donc une bonne alternative au fait d'utiliser un algorithme classique exponentiel.
- que beaucoup de problèmes industriels peuvent se réencoder en un problème Max-Cut. QAOA permettrait donc de résoudre tous ces problèmes.

**Choisir pour  $H_B$  et  $|\varphi_B\rangle$ .** Pour le choix de cette matrice, il en faut une où il est simple de déterminer et construire un état stable  $|\varphi_B\rangle$ . On peut prendre par exemple  $H_B = X^{\otimes n}$ .

$$\begin{array}{c} \text{---}\oplus\text{---} \\ \text{---}\oplus\text{---} \\ \vdots \\ \text{---}\oplus\text{---} \end{array}$$

Les vecteurs propres de la porte  $X$  sont  $|+\rangle$  et  $|-\rangle$ . En effet  $H|+\rangle = |+\rangle$  et  $H|-\rangle = (-1)|-\rangle$ . Un état stable de  $X$  est donc  $|-\rangle$ . Un état stable de  $X^{\otimes n}$  est, par exemple,  $|++++\dots+-\rangle$ . On peut construire cet état simplement avec  $H^{\otimes n}|000\dots 01\rangle$ .

**Construire la porte  $\exp\left(-i\frac{dt}{\hbar}H(j \cdot dt)\right)$ .** Pour des raisons de simplicité, on va réécrire  $\frac{dt}{\hbar}H(j \cdot dt)$  sous la forme  $\gamma H_C + \beta H_B$  où  $\gamma$  et  $\beta$  dépendent de  $dt$ ,  $j$  et  $\hbar$ . Ainsi

$$\exp\left(-i\frac{dt}{\hbar}H(j \cdot dt)\right) = \exp(-i\gamma H_C - i\beta H_B)$$

On souhaiterais pouvoir dire

$$\exp(-i\gamma H_C - i\beta H_B) = \exp(-i\gamma H_C) \exp(-i\beta H_B)$$

Cependant cette égalité n'est vraie que si  $H_B$  et  $H_C$  commutent, ce qui n'est pas le cas ici. Toutefois, on peut utiliser la formule suivante

$$\exp(-i\gamma H_C - i\beta H_B) = \lim_{p \rightarrow +\infty} \left( \exp\left(\frac{-i\gamma H_C}{p}\right) \exp\left(\frac{-i\beta H_B}{p}\right) \right)^p$$

Pour une grande valeur de  $p$ , on a donc

$$\exp(-i\gamma H_C - i\beta H_B) \simeq \left( \exp\left(\frac{-i\gamma H_C}{p}\right) \exp\left(\frac{-i\beta H_B}{p}\right) \right)^p$$

Notre circuit QAOA peut donc se réécrire comme suit

$$|\varphi_B\rangle \rightarrow \boxed{e^{-i\gamma_1 H_C}} \boxed{e^{-i\beta_1 H_B}} \boxed{e^{-i\gamma_2 H_C}} \boxed{e^{-i\beta_2 H_B}} \dots \boxed{e^{-i\gamma_k H_C}} \boxed{e^{-i\beta_k H_B}} \rightarrow |\varphi_C\rangle$$

Reste donc la question, connaissant des constantes  $\gamma$  et  $\beta$ , quels circuits encodent les portes  $\exp(-i\gamma H_C)$  et  $\exp(-i\beta H_B)$ .

Pour la porte  $\exp(-i\gamma H_C)$ , on va supposer, comme dans le problème de coupe maximum, que  $H_C$  s'écrit comme une somme de produits tensoriels de portes  $I_2$  et de  $Z$ . Dans ce cas, notons  $A_i$  les différents produits, on a  $H_C = \sum_{i=1}^k \alpha_i A_i$  où  $A_i$  est un produit tensoriels de  $n$  portes  $I_2$  et  $Z$  et où  $\alpha_i$  est un nombre réel.

On peut noter que les portes  $A_i$  commutent,  $A_i A_j = A_j A_i$ . En effet,  $A_i A_j$  est équivalent à un circuit où chaque ligne contient zero, une ou deux portes  $Z$ . Mettre le circuit  $A_j$  avant  $A_i$  revient donc au même. Et puisque ces portes commutent alors on a la propriété suivante

$$\exp(A_i + A_j) = \exp A_i \exp A_j$$

Donc

$$\exp(-i\gamma H_C) = \prod_{i=1}^k \exp(-i\alpha_i \gamma A_i)$$

Il suffit donc de trouver le circuit de  $\exp(-i\alpha_i \gamma A_i)$  pour chaque  $A_i$ . Sans perte de généralité, supposons donc que  $H_C = Z \otimes Z \otimes I_2 \otimes \dots \otimes I_2$ .

$$\exp(-i\gamma H_C) = \sum_{k=0}^{+\infty} \frac{(-i\gamma)^k}{k!} H_C^k$$

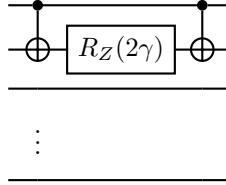
On rappelle que mettre deux portes  $Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$  à la suite, donc la porte  $Z^2$  est en fait l'identité. Donc mettre deux fois  $H_C$  de suite (qui possède 0 ou 1 porte  $Z$  par ligne) revient aussi à l'identité,  $H_C^2 = I_{2^n}$ . On peut donc, dans la somme, séparer les  $k$  pairs et les  $k$  impairs.

$$\begin{aligned}\exp(-i\gamma H_C) &= \sum_{k=0}^{+\infty} \frac{(-i\gamma)^{2k}}{(2k)!} I_{2^n} + \frac{(-i\gamma)^{2k+1}}{(2k+1)!} H_C \\ \exp(-i\gamma H_C) &= \cos(\gamma) I_{2^n} - i \sin(\gamma) H_C\end{aligned}$$

On rappelle, que, dans  $H_C$ , la première et la seconde ligne contiennent une porte  $Z$ . Ainsi, pour tout vecteur de base  $|\underline{x}\rangle$

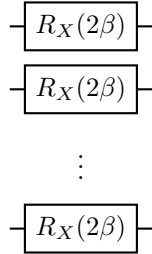
$$\begin{aligned}\exp(-i\gamma H_C) |\underline{x}\rangle &= \begin{cases} (\cos(\gamma) - i \sin(\gamma)) |\underline{x}\rangle & \text{si } x_1 = x_2 \\ (\cos(\gamma) + i \sin(\gamma)) |\underline{x}\rangle & \text{sinon} \end{cases} \\ \exp(-i\gamma H_C) |\underline{x}\rangle &= \begin{cases} \exp(-i\gamma) |\underline{x}\rangle & \text{si } x_1 = x_2 \\ \exp(i\gamma) |\underline{x}\rangle & \text{sinon} \end{cases}\end{aligned}$$

On peut donc obtenir  $\exp(-i\gamma H_C)$  avec le circuit suivant:



$$\text{où } R_Z(2\gamma) = \begin{pmatrix} \exp(-i\gamma) & 0 \\ 0 & \exp(i\gamma) \end{pmatrix}.$$

Calculer  $\exp(-i\beta H_B)$  où  $H_B = X^{\otimes n}$  se fait de manière similaire. On obtient le circuit suivant (il n'y a pas de CNOT contrairement à la version avec  $H_C$  car le déroulement du calcul avec les portes  $X$  n'est pas le même).



$$\text{où } R_X(2\beta) = \begin{pmatrix} \cos \beta & -i \sin \beta \\ -i \sin \beta & \cos \beta \end{pmatrix}.$$

**Circuit paramétré ? Ansatz ?** Le circuit de QAOA simule le théorème adiabatique, mais il pose un petit problème, il est très gros.

$$|\varphi_B\rangle \rightarrow \boxed{e^{-i\gamma_1 H_C}} \boxed{e^{-i\beta_1 H_B}} \boxed{e^{-i\gamma_2 H_C}} \boxed{e^{-i\beta_2 H_B}} \dots \boxed{e^{-i\gamma_k H_C}} \boxed{e^{-i\beta_k H_B}} \simeq |\varphi_C\rangle$$

Mais, en réalité, il n'est pas nécessaire de simuler parfaitement le théorème adiabatique. Tant qu'on atteint un état stable à la fin ou même un état qui encode une bonne solution du problème est peut être satisfait. Plutôt que d'utiliser ce circuit qui contient un très grand nombre de portes, on va fixer une valeur  $k'$  plus petite (disons suffisamment petite pour que le circuit puisse être exécuté sur une machine quantique) et s'intéresser au circuit suivant

$$|\varphi_B\rangle \rightarrow \boxed{e^{-i\gamma_1 H_C}} \boxed{e^{-i\beta_1 H_B}} \boxed{e^{-i\gamma_2 H_C}} \boxed{e^{-i\beta_2 H_B}} \dots \boxed{e^{-i\gamma_{k'} H_C}} \boxed{e^{-i\beta_{k'} H_B}} \simeq |\varphi_C\rangle$$

Ce circuit est donc une approximation du circuit qui simule le théorème adiabatique. Mais pour qu'il donne de bons résultats, il va s'autoriser à modifier librement les valeurs de  $\gamma_i$  et  $\beta_i$ . La question devient donc, quelles valeurs de  $\gamma_i$  et  $\beta_i$  faut-il pour que le circuit renvoie un état stable du système avec une forte probabilité ? Si on note  $C(\gamma, \beta)$  le circuit, on obtient un circuit dit paramétré. C'est à dire un circuit dont les portes dépendent d'une valeur de paramètres. Dit autrement c'est une fonction qui à des paramètres associe un circuit. C'est ce qu'on appelle un *Ansatz*.

**Trouver l'état stable ?** Comment trouver les meilleurs paramètres ? Ou plus exactement, comment trouver l'état stable ? Il faut utiliser ce qu'on appelle un algorithme d'optimisation Boîte noire. Le circuit est une boîte noire. Il renvoie un état en sortie. Si on exécute plusieurs fois la boîte, il renvoie une distribution d'états. On peut donc estimer si le circuit est bon ou non en regardant l'espérance de  $f(x)$  sur cette distribution.

Pour chaque  $x$  donné, on obtient une valeur  $f(x)$  et on calcule la moyenne de ces valeurs. Cela nous donne une estimation de si le circuit donne plus souvent des solutions de bonne ou de mauvaise qualité. On change ensuite les paramètres  $\gamma_i$  et  $\beta_i$  et on recommence. Si ça améliore l'estimation, alors on a sans doute modifié les paramètres comme il faut. Sinon on s'est peut être trompé de direction et il faut chercher ailleurs. La méthode pour trouver les meilleurs paramètres peut varier en fonction du problème. On peut utiliser des méthodes de gradient, du machine learning, des méthodes à base d'algorithme génétique, .... Notez qu'on utilise un algorithme classique pour résoudre le problème d'optimisation des paramètres. La partie quantique est juste la boîte noire.

On arrête lorsqu'on a effectué trop d'itérations, ou lorsqu'on a convergé vers une solution qui nous semblait suffisamment bonne. Ce n'est peut être pas l'optimal, car QAOA n'est pas un algorithme exact. Mais c'est sans doute une solution qui conviendra.