

IQRO_QAOA

Night

January 2024

1 Description du problème

1.1 Premier problème

Le problème **Max-Cut** étant défini pour des sommets et non des arêtes, posons $x_{i,j}$ comme la variable associée à l'arête formée par les sommets **i, j**.

$$\forall i, j \ x_{i,j} \in \{0, 1\}$$

On cherche à minimiser sous contraintes qu'au moins une arête par sommet soit comprise dans U la somme $\sum_{i,j \in E} x_{i,j}$

$$\forall i \in V : \exists j \in V : x_{i,j} = 1$$

Soit

$$\forall j \in V, \sum_{i \in V} x_{i,j} \geq 1$$

1.2 Modélisation si tous les sommets sont de degré 1 ou 2

On a donc comme nouvelles contraintes:

$$\forall j \in V$$

- Si $\deg(j) = 1$, $x_{1,j} \geq 1$
- Si $\deg(j) = 2$, $x_{1,j} + x_{2,j} \geq 1$

2 Résolution avec QAOA

2.1 Reformulation du problème en forme QUBO

On pose alors la contrainte

$$g(x) = 1 - \sum_{i,j \in E} x_{i,j}$$

Nous avons donc:

$$p_C(x) = \begin{cases} 0, & \text{sig}(x) \leq 0 \\ p_C(x) > 0, & \text{sig}(x) > 0 \end{cases}$$

On doit distinguer **2 types de contraintes**

Notons $x_{1,j}$ la variable correspondant à un sommet de degré 1, $x_{2,j} \geq 1$ si correspondant à une 2nde arête dans le cas où le degré du sommet considéré vaut 2.

- $p_{C1}(x) = (1 - x_{1,j})$
Si $\deg(j) = 1$, la contrainte $x_{1,j} \geq 1$ sera respectée ($x_{2,j} = 0$) et la fonction de pénalité sera $p_C(x) = (1 - 1) = 0$.
 p_{C1} est bien **linéaire** (polynomiale de degré 1).

- $p_{C2}(x) = (1 - (x_{1,j} + x_{2,j}) + x_{1,j}x_{2,j})$
Si $\deg(j) = 2$, la contrainte $x_{1,j} + x_{2,j} \geq 1$ sera respectée et la fonction de pénalité sera $p_C(x) = (2 - 2) = 0$.
Le terme $x_{1,j}x_{2,j}$ permet de s'assurer dans le cas où les 2 arêtes sont contenues, que $p_{C2} = 1 - (1+1) + 1 = 0$
 p_{C2} est bien **polynomiale**.

Si la contrainte n'est pas respectée pour le degré 1 ou 2, la fonction de pénalité attribuera une valeur positive. Ainsi nous avons trouvé nos **2 types de contraintes au plus polynomiales**

Ainsi le modèle **QUBO** vaut sous contrainte de $g(x)$:

$$Q(x) = (x_1, x_2) = \sum_{i,j \in E} x_{i,j} + \sum_{j \in D1} (1 - x_{1,j}) + \sum_{j \in D2} (1 - (x_{1,j} + x_{2,j}) + x_{1,j}x_{2,j})$$

2.2 Hamiltonien

2. Écrire l'Hamiltonien associé. (2 points)

3. Résoudre ce problème sur le simulateur IBM à 32 qubits, pour le graphe G donné Figure 1 avec la fonction QAOA pré-codée sur Qiskit. Illustrer l'influence de la profondeur p du circuit. (4 points)

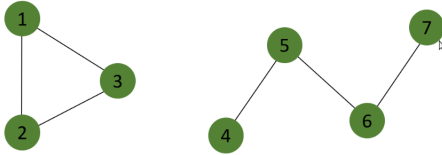


Figure 1: Instance G pour le problème de couverture minimale par les arêtes.

3. Résolution avec l'algorithme de Grover (9 points)

Développons avec le graphe donné, il faut simplifier selon les degrés (**en notant $x_{1,j} = x_{j,1} = x_{j1}$ unique arête pour ne pas créer de confusion:**

$$Q(x) = \sum_{i,j \in \{1,7\}} x_{i,j} + \sum_{j \in \{4,7\}} (1 - x_{j1}) + \sum_{j_{1,2} \in \{1,2,3,5,6\}} (1 - (x_{j1} + x_{j2}) + x_{j1}x_{j2})$$

En introduisant λ le **coefficient pénalité** pour toutes les contraintes on a donc:

$$H(x) = \sum_{i,j \in \{1,7\}} x_{i,j} + \lambda \sum_{j \in \{4,7\}} (1 - x_{j1}) + \lambda \sum_{j_{1,2} \in \{1,2,3,5,6\}} (1 - (x_{j1} + x_{j2}) + x_{j1}x_{j2}) / 2$$

Enfin on peut poser $z_{i,j} = 2 * x_{i,j} - 1$ suivant le modèle d'Ising pour simplifier l'expression.

2.3 Résolution

Voir `tp_note.py`, on trouve un minimum de **4 arêtes**.

3 Grover

3.1 Contraintes

Pour exprimer le min (problème d'optimisation) en problème de **décision**:

On minimise sous contraintes $\sum x_{ij}$
 $\exists x \in X tq f(x) \leq k$

3.2 Proposition schématique de l'oracle du problème de décision

L'oracle de Grover se divise en deux parties :

- **Partie 1 - Marquage des solutions réalisables** : Marquage des états quantiques correspondant aux solutions réalisables du problème. Dans ce contexte, marquer les états quantiques représentant des couvertures minimales par les arêtes.
- **Partie 2 - Sélection des solutions inférieures à k** : Utilisation d'un comparateur quantique pour sélectionner les états quantiques dont la valeur est inférieure à k .

3.3 Implémentation pour le graphe de la Figure 1, avec $k = 3$:

On utilise le circuit quantique pour appliquer ces opérations. Le circuit comparateur inférieur strict (Figure 2) peut être utilisé pour effectuer la comparaison. Les qubits $|q5\rangle, \dots, |q8\rangle$ sont des qubits auxiliaires utilisés dans le processus de comparaison.

- On applique la Partie 1 pour marquer les solutions réalisables. - Puis le circuit comparateur (Figure 2) pour sélectionner les solutions inférieures à k .

3.4 Oracle du problème de décision pour le graphe Figure 1, avec $k = 3$:

L'oracle combiné peut être formulé comme suit, en utilisant l'opérateur de comparaison X pour marquer les solutions réalisables et le circuit comparateur (Figure 2) pour sélectionner les solutions inférieures à k :

$$Oracle = X_{solréalisables} \cdot CircuitCompareur_{k=3}$$

Cette expression doit être adaptée en fonction de la représentation spécifique des états quantiques.

3.5 Conclusion

L'oracle ainsi construit permettrait d'identifier les solutions réalisables du problème de couverture minimale par les arêtes et de sélectionner celles dont la valeur est inférieure à $k = 3$.