

Journée Pack Quantique

ATELIER – Méthodes de calcul hybrides pour résoudre des problèmes d'optimisation combinatoire

Optimization combinatoire

- Trouver une solution optimale parmi un ensemble fini, mais large de façon prohibitive, d'options
- Un espace de solution de taille exponentielle, trop d'options à explorer

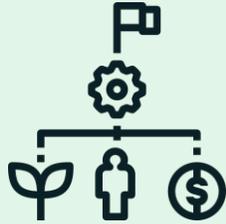


Optimization combinatoire



Chaîne logistique

Système optimal d'approvisionnement, de gestion des opérations, de logistique et de marketing



Allocation des ressources

Attribuer et gérer les actifs de manière à soutenir les objectifs de planification stratégique d'une organisation



Conception de réseau

Prendre de meilleures décisions de conception qui auront des implications considérables



Planification

Commande optimale et coordination dans le temps de certaines opérations



Portefeuille

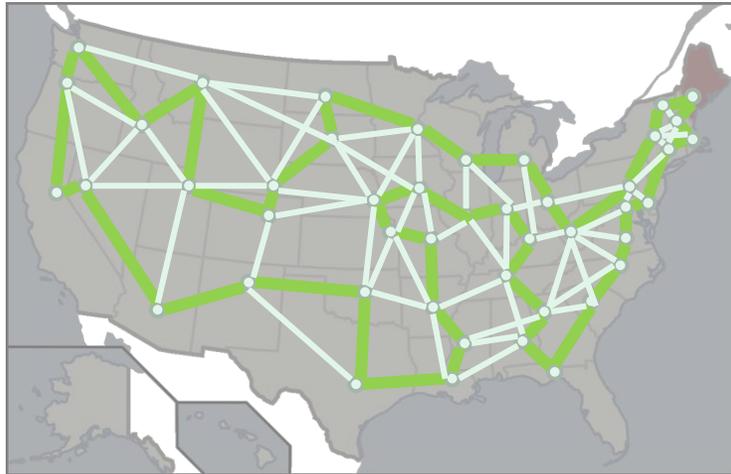
Maximiser le rendement attendu tout en minimisant les risques financiers

Problèmes sous forme de graphe

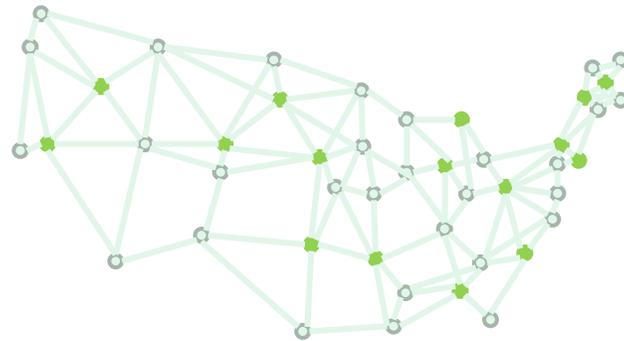
Qu'est-ce que c'est une '**graphe**' ?

- Des structures mathématiques pour modéliser des objets et des relations entre eux
- Sont constituées de sommets (points) liés par des arêtes (liens)

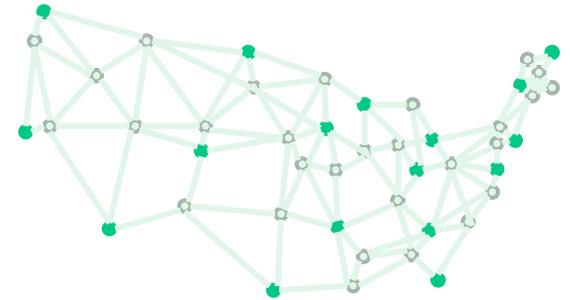
Il y a des liens intrinsèques entre des problèmes combinatoires et des problèmes sur graphes.



Find an itinerary which visits every location while minimizing total distance



Find bipartition z that maximizes the number of edges which are "cut" (one vertex in each set)

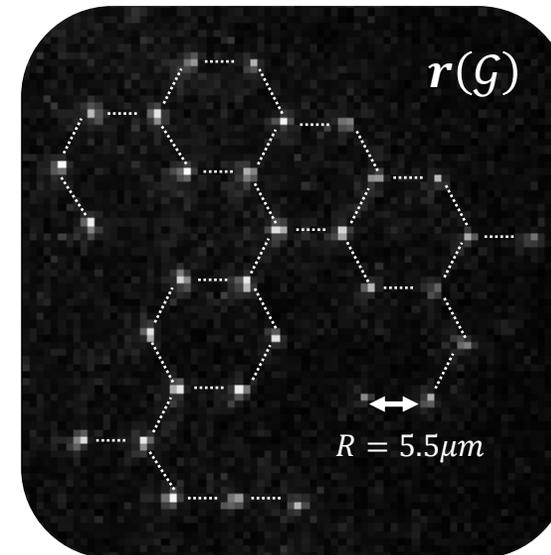
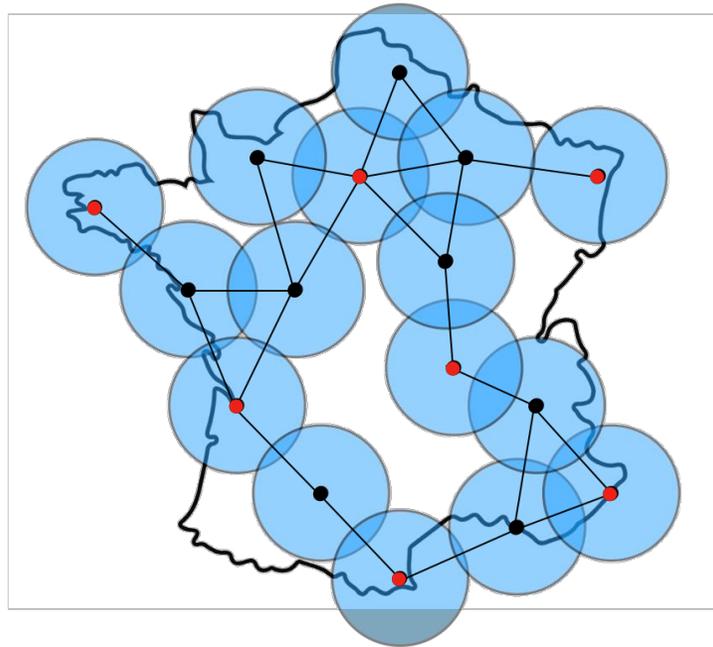


Find the largest set z of vertices subject to constraint that no two are adjacent

Encoder des problèmes de graphe sur un QPU à atômes neutres

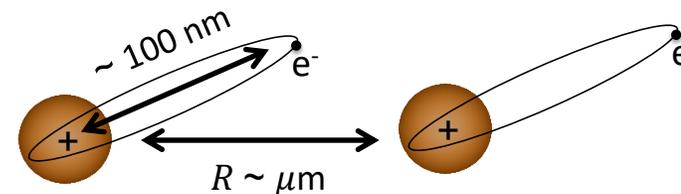
Les sommets sont les atomes

Les arêtes sont des interactions (dipôle-dipôle)



Topologie du graphe

$$H_G = \sum n_i n_j$$



Coloration des graphes

Let's see why cold atoms are so hot right now!

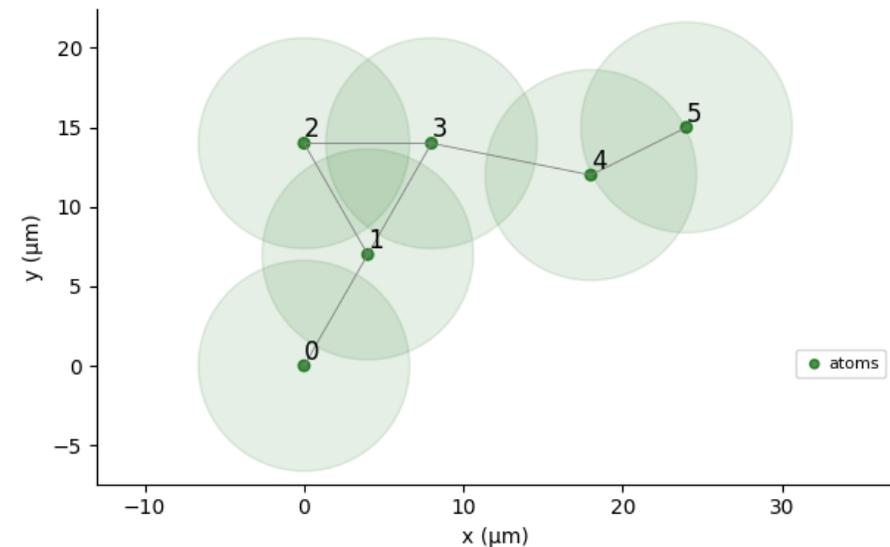
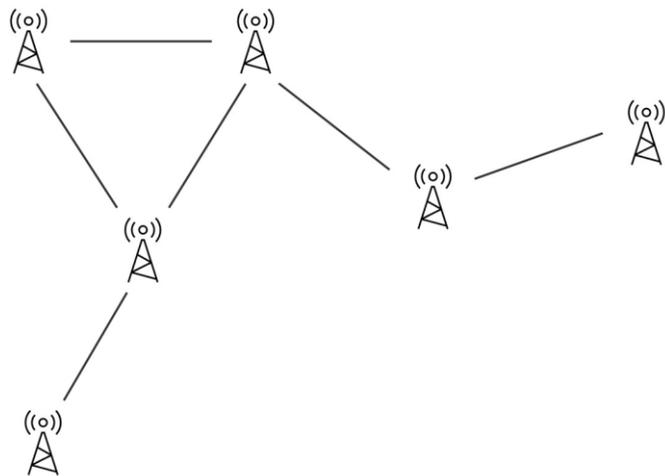


Maximum Independent Set

Un ensemble indépendant (**independent set**) d'un graph est un sous-ensemble de sommets tel que chaque pair de sommets n'a pas d'arête entre eux

Un ensemble indépendant maximale (maximum independent set ou MIS) est un ensemble indépendant de taille maximale dans un graph donné.

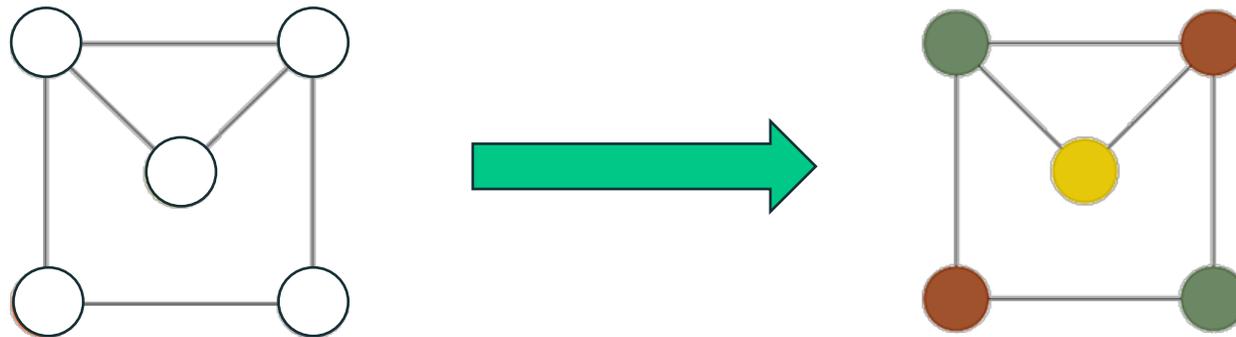
Les dynamiques de Rydberg peuvent être utilisées pour encoder les contraintes d'indépendance pour un registre atomique, modélisé par un graph unit-disk



Coloration des graphes

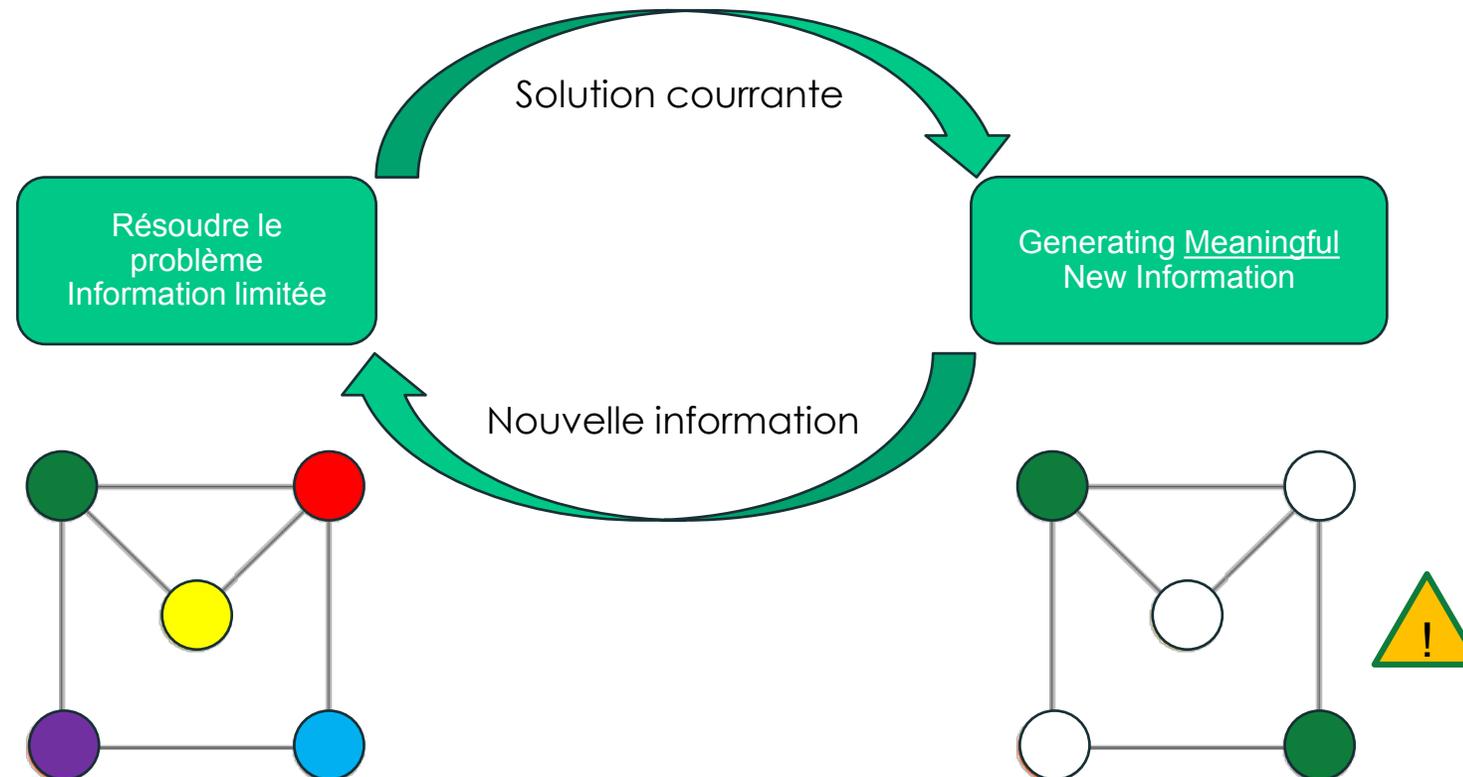
L'objectif c'est

- Colorer tout les sommets du graphe avec un nombre minimal de couleurs
- Les sommets d'une arête ne peuvent pas avoir la même couleur



Coloration des graphes

- La méthode de décomposition nous permet de diviser le problème en deux parties :
1. Master Problem: Résoudre un problème simplifié avec de l'information limitée
 2. Sub-Problem: Générer de la nouvelle information utile au Master Problem



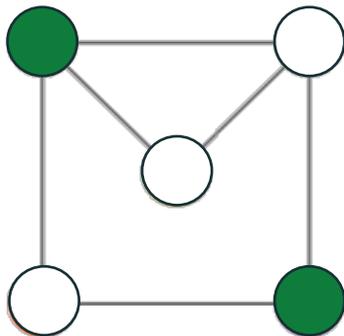
Le verrou : Générer de la nouvelle information est classiquement difficile

Génération Classique

Le Sub-Problem est souvent aussi difficile à résoudre que le problème originel

Générer de la nouvelle information est lente et couteuse en ressources de calcul

Il est compliqué de garantir l'utilité de la nouvelle information générée

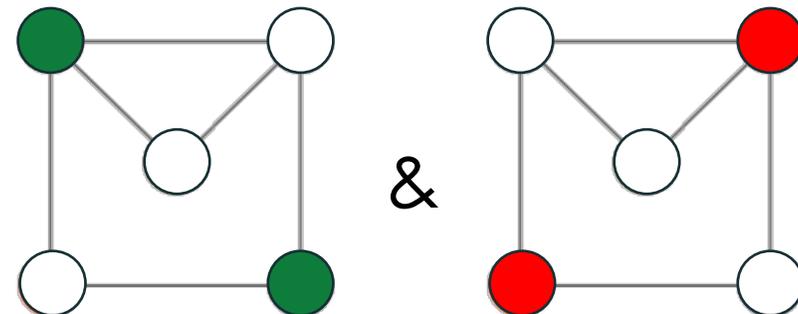


Génération quantique

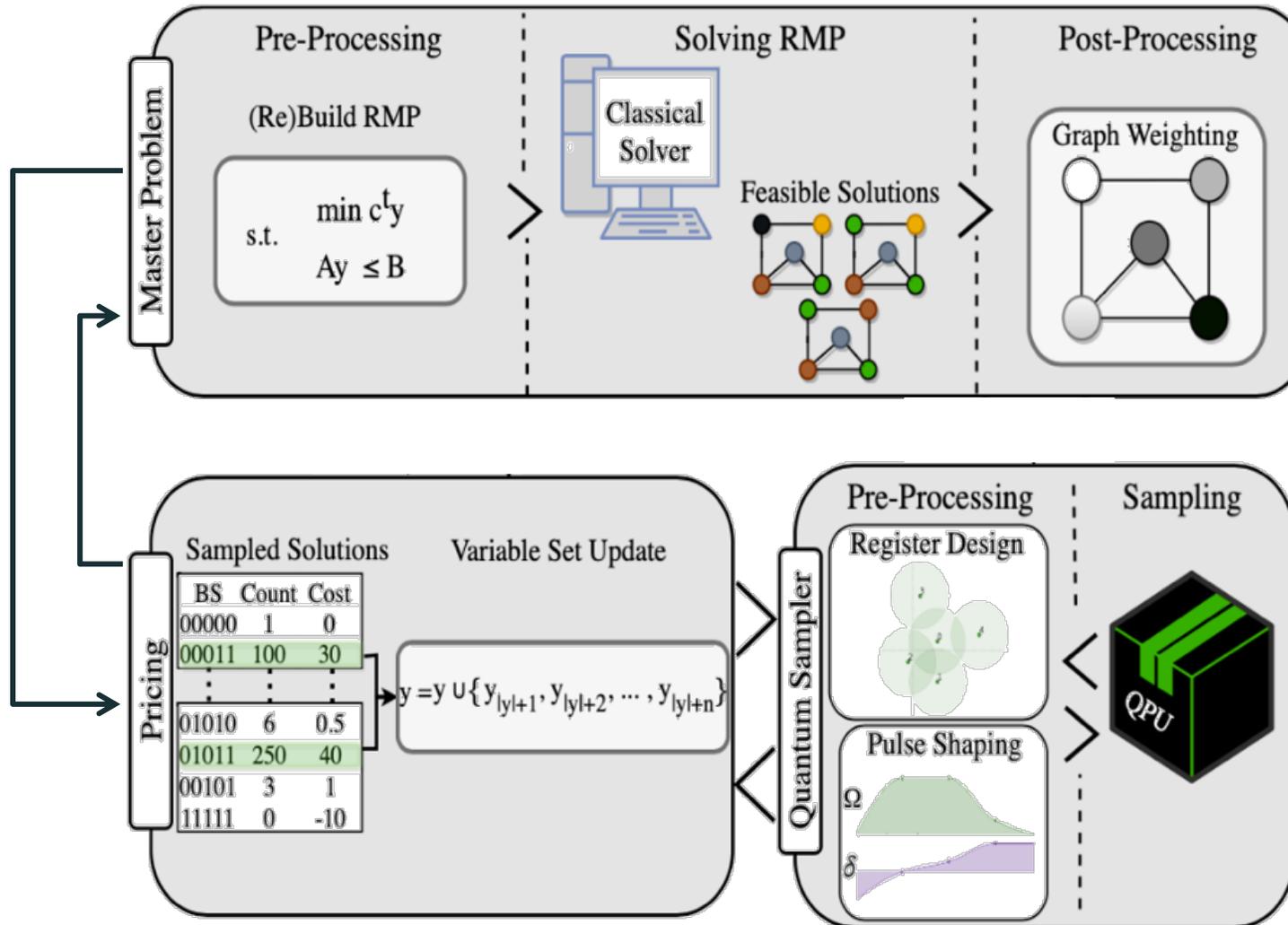
La taille du Sub-Problem est plus petit donc des méthodes hybrides / quantiques sont faisables

Il est possible de générer plusieurs nouvelles informations dans une seule itération, ce qui accélère l'algorithme

Il est facile de garantir l'utilité de l'information générée



Méthode de génération de colonnes



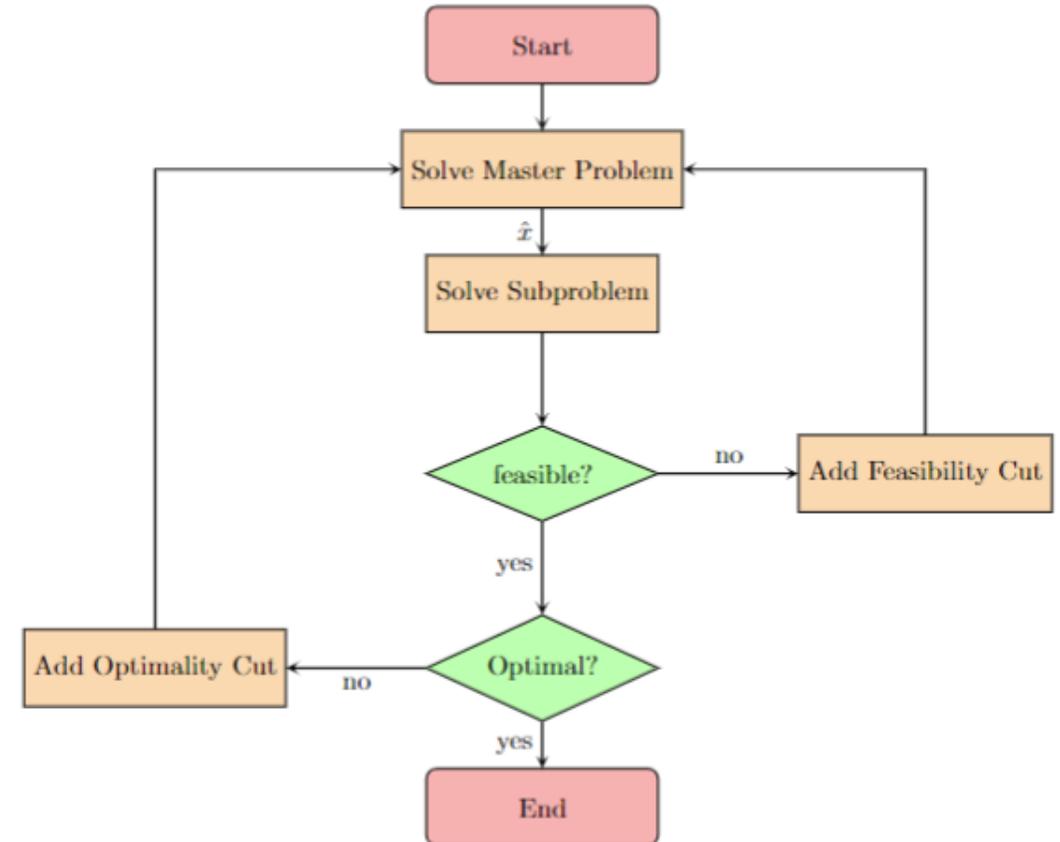
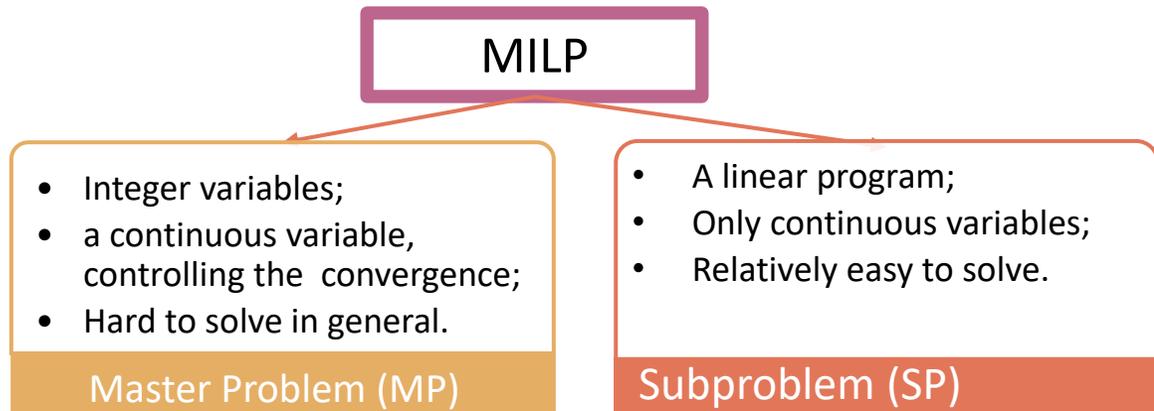
Calcul classique

- Trouver la meilleure coloration parmi un ensemble de combinaisons (variables) réduit
- Créer des instances pour le sous-problème
- Exécuter les algorithmes de création de register atomiques et de design de sequence de pulse

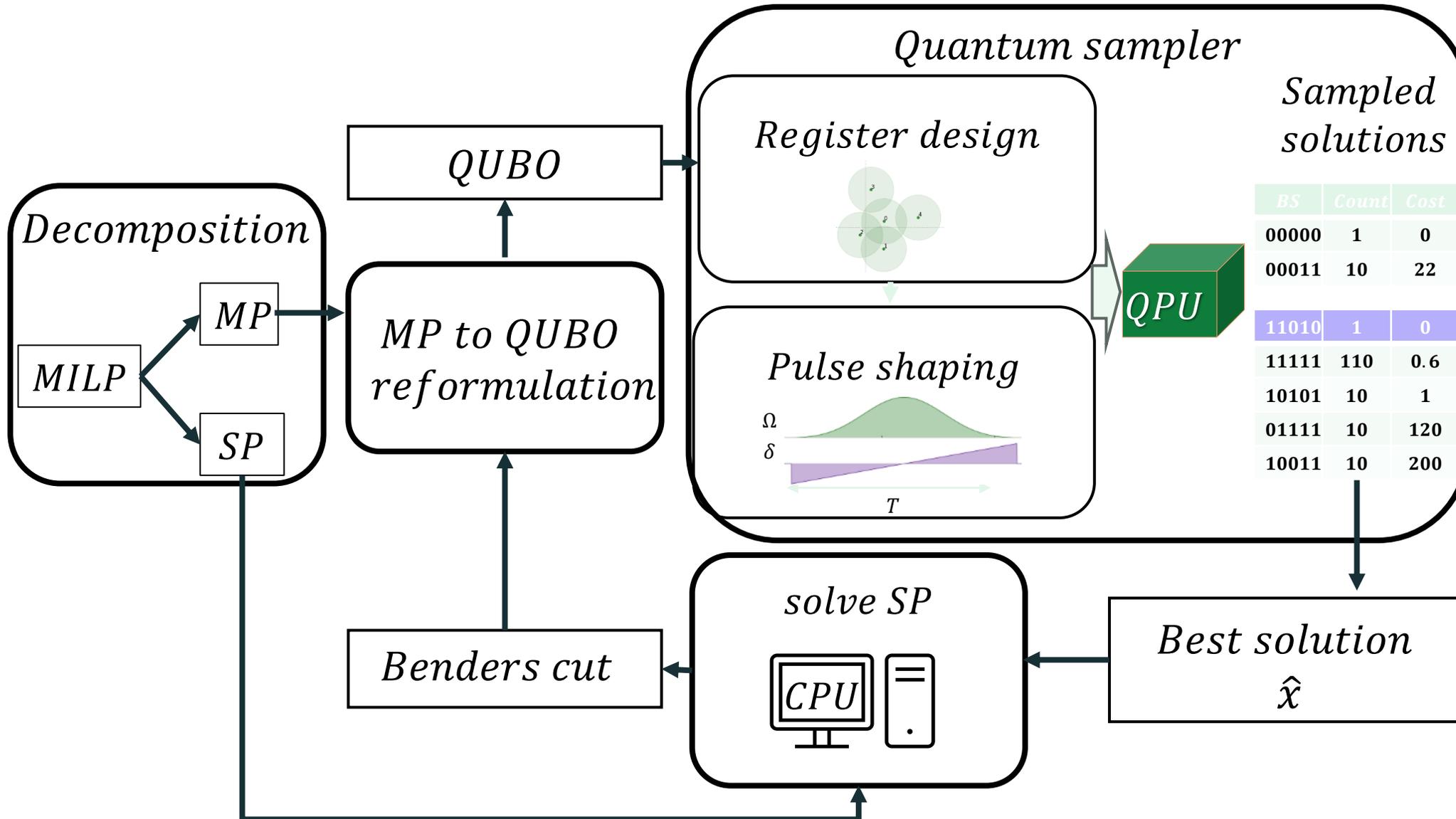
Calcul quantique

- Faire des échantillons de solutions admissibles pour chaque sous-problème
- Enrichir le problème master avec des Nouvelles configurations (variables)

Mixed Integer Linear Programming



Décomposition de Benders



Branch and Bound

Une méthode standard pour résoudre des problèmes d'optimisation combinatoire

Branch: Diviser le problème en plusieurs sous-problèmes

Bound: Eliminer certaines branches selon des critères (pruning - tailler)

Notamment des bornes inférieures et supérieures sur l'espace de recherche

Construire une arbre :

- La racine est la graphe complète
- Pour chaque sommet, on cherche un ensemble MIS et on lui associe une branche
- Pour chaque branche on associe une couleur et on le retire du graphe

Chaque Sommet de l'arbre correspond à :

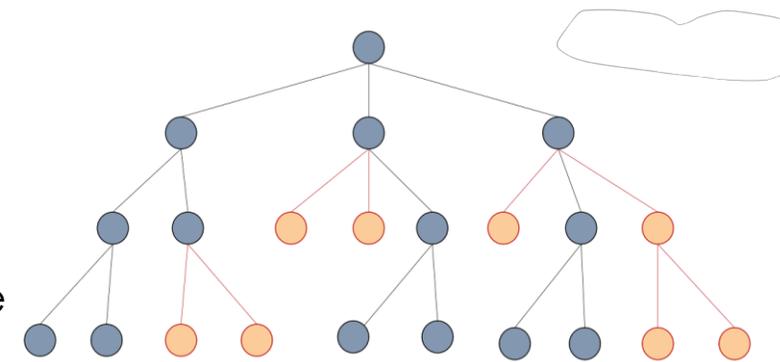
Un (sous-)graphe induit

Une couleur correspondante

Des bornes inférieures et supérieures sur les meilleures colorations atteignables à partir de ce sommet

Tailler l'arbre : retirer des sommets où on n'observe pas d'amélioration ou bien où il y a des scénarios non-faisables

L'exploration de l'arbre est réalisé avec un score de priorité



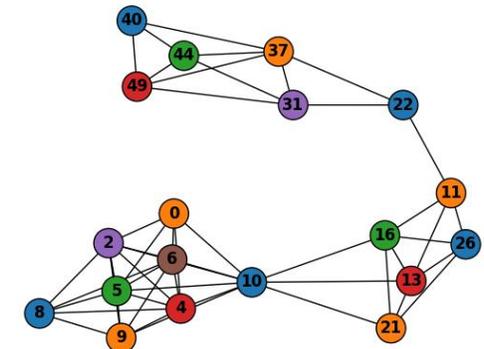
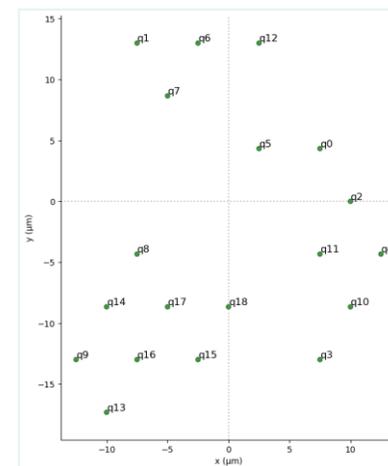
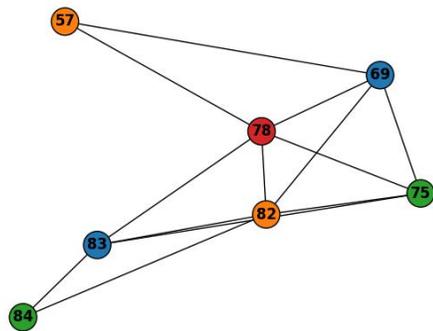
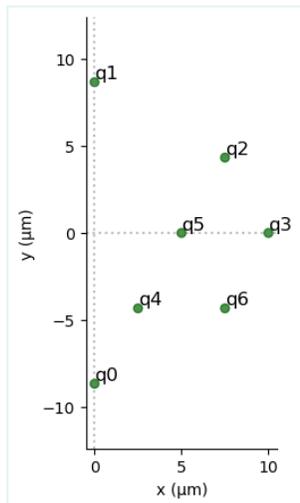
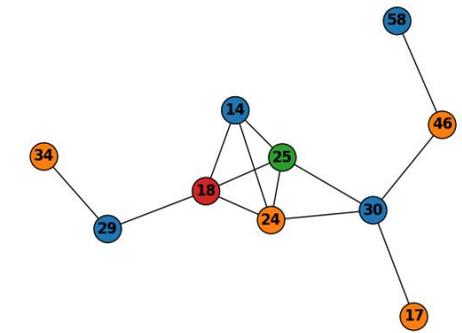
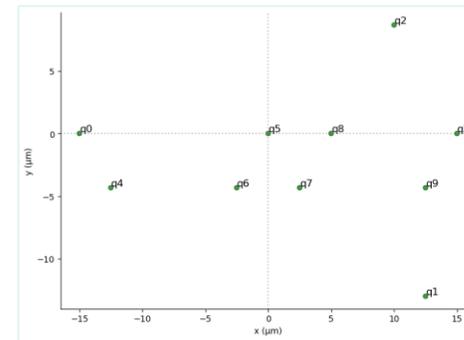
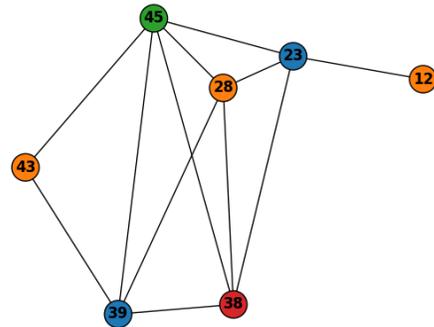
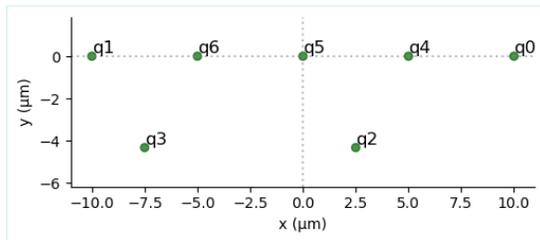
Pruning the B&B tree to reduce the number of nodes to be explored

Quelques résultats

4 graphes: 7, 7, 10 et 19 qubits

Pour chaque sommet BB environ 25 iteration d'optimization QAOA (avec 75 shots) et une échantillon finale de (110 shots).

Chaque instance a générée la solution optimale



D'autres résultats

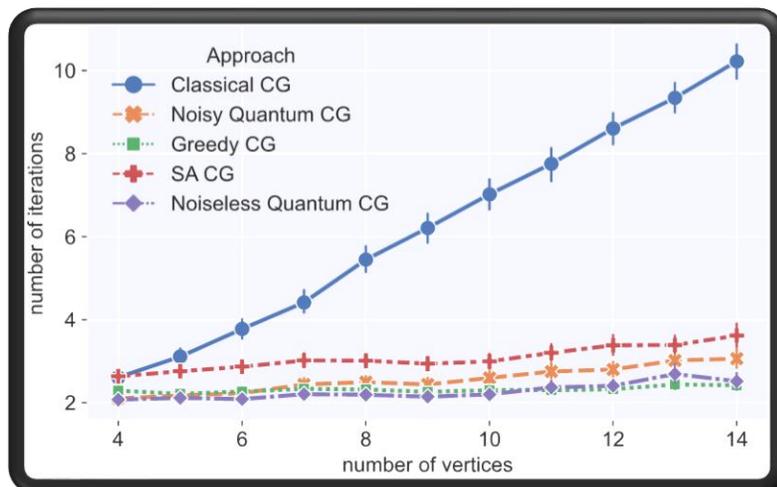
Comparison de nos résultats hybrids avec des références classiques

- 🌀 Solution Classique exacte: toujours la meilleure solution pour chaque itération
- 🌀 Heuristique State-of-the-art: peut fournir plusieurs résultats optimales

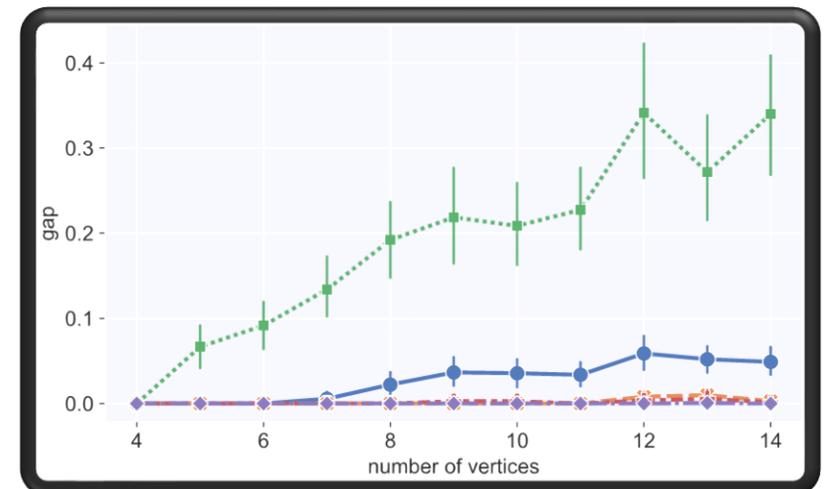
Notre méthode hybride indique les meilleures performances

- 🌀 Il est capable de fournir de meilleures résultats que le heuristique : jusqu'à 80 % de moins de couleurs utilisées
- 🌀 Il est plus rapide que des méthodes classiques : jusqu'à 6 fois plus rapide que les solveurs exactes

Nombres d'iterations qui génère des nouvelles options



Distance de la solution optimale



Conclusions

- Des ordinateurs classiques (CPUs et GPUs) jouent toujours un rôle important dans la résolution des problèmes combinatoires
 - Aide à optimiser les paramètres des QPUs
 - Nécessaire pour du pre- et post-processing des données quantiques
 - Peut résoudre de façon efficace certaines étapes des algorithmes
 - Des outils utiles pour tester des idées et des modèles jouets
- Des ordinateurs quantiques peuvent résoudre de façon plus efficace certaines classes de problèmes
 - Peut fournir de l'information critique sur le problème
 - Peut compléter des ordinateurs classiques dans la résolution des problèmes complexes
 - Cibler de façon efficace les verrous de l'algorithme

