Étude des performances des machines à recuit quantique pour la résolution de problèmes combinatoires

Daniel Vert

université
PARIS-SACLAY

Etude des performances des machines à recuit quantique pour la résolution de problèmes combinatoires

Thèse de doctorat de l'université Paris-Saclay

École doctorale n° 580, Sciences et Technologies de l'Information et de la Communication (STIC)
Spécialité de doctorat : Informatique
Unité de recherche : Université Paris-Saclay, CEA, Institut LIST, 91191, Gif-sur-Yvette, France.
Référent : Faculté des sciences d'Orsay

Thèse présentée et soutenue à Paris Saclay le 22 Mars 2021, par

Daniel VERT

Composition du jury :

Tristan Meunier Directeur de recherche CNRS, Institut NEEL/CNRS	Rapporteur & Examinateur
Marc Sevaux Professeur des universités, Université de Bretagne-Sud	Rapporteur & Examinateur
Dritan Nace Professeur des universités, Université de Technologie de Compiègne	Président
Elham Kashefi Directrice de recherche CNRS, Université Paris-Sorbonne	Examinatrice
Daniel Estève Directeur de recherche CEA, Université Paris-Saclay	Examinateur
Louis Granboulan Expert sénior, Airbus Group	Examinateur
Renaud Sirdey Directeur de recherche CEA, Université Paris-Saclay	Directeur de thèse
Stéphane Louise Directeur de recherche CEA, Université Paris-Saclay	Codirecteur de thèse

Thèse de doctorat

NNT : 2021UPASG026

"In my opinion, the crucial experiment (which has not yet been done) would be to compare the adiabatic algorithm head-on against simulated annealing and other classical heuristics. The evidence for the adiabatic algorithm's performance would be much more convincing if the known classical algorithms took exponential time on the same random instances."

Scott Aaronson

Remerciements

Je tiens tout d'abord à remercier grandement mes directeur de thèse, Mesieurs R. Sirdey et S. Louise, pour toute l'aide qu'ils m'ont apportée tout au long de cette thèse. Qu'ils soient aussi remerciés pour leurs disponibilités permanentes et pour les nombreux encouragements qu'ils m'ont prodigués.

Le travail qui est présenté dans ce mémoire a été effectué au CEA List au sein du Laboratoire LCYL. Aussi, merci à Messieurs O. Heron et C. Gamrat, de m'avoir accueilli et mis à ma disposition tous les moyens nécessaires aussi bien techniques que financiers, afin de mener à bien ce travail.

Merci à Monsieur D. Nace pour avoir accepté de présider mon jury de thèse. Un grand merci à Messieurs T. Meunier et Monsieur M. Sevaux, pour avoir accepté d'étudier mes travaux et d'en être les rapporteurs. Merci à Madame E. Kashefi et Messieurs D. Estève et L. Granboulan de l'attention qu'ils portent à mon travail en acceptant d'être membres du jury. Je tiens plus particulièrement remercier D. Estève, pour m'avoir mis en contact avec l'équipe UCLA et m'avoir offert la possibilité de présenter mes travaux lors d'un séminaire.

J'aimerais ensuite remercier toutes les personnes que j'ai rencontrées au LCYL tout au long de ma thèse. En particulier, je tiens à remercier mes anciens collègues F. Hebbach, M. Zuber, P. Glanon., M. Aitaba, B. Ben Hedia, M. Asavoae, E. Hamelin. G. Bottonte, L. Dolci et B. Binder pour toutes les discussions intéressantes que nous avons eues. Je voudrais également exprimer ma gratitude à M. Giudici et E. Pichon, secrétaires du Département DSCIN, pour leurs aides durant ces trois dernières années.

Je tiens à remercier grandement mes comparses, sans elles ce manuscrit n'aurait pu voir le jour, Olivia S. (alias ma mie), Audrey B. (alias accueil), Doriane G. (alias militaire), Laura B. (alias ex coloc) et Naomie J. (et lapin). Un grand merci pour leur soutien inconditionnel et surtout pour m'avoir supporté.

Enfin, je voudrais exprimer ma profonde gratitude à mes parents, Georges et Françoise, ainsi que mon frère David pour leurs inflectibles soutiens et leurs aides si précieuses au quotidien.

UNIVERSITÉ PARIS SACLAY

Résumé

Etude des performances des machines à recuit quantique pour la résolution de problèmes combinatoires

par Daniel VERT

L'informatique quantique suscite un regain d'intérêt avec les récentes annonces de plusieurs acteurs. La raison la plus évidente est que certains algorithmes quantiques peuvent résoudre en temps polynomial les mêmes tâches qui sont actuellement considérées comme non polynomiales sur les ordinateurs classiques. Ces dernières années, sont apparues des machines quantiques analogiques, dont les calculateurs actuellement commercialisés par la société canadienne D-Wave sont les premiers représentants, fonctionnant selon un principe de recuit à accélération quantique. D'un point de vue abstrait, une telle machine peut être considérée comme un oracle spécialisé dans la résolution d'un problème d'optimisation NP-difficile avec un algorithme fonctionnellement analogue au recuit simulé mais avec une accélération quantique. Outre les analogies formelles entre le recuit simulé et le recuit quantique, il y a une analogie entre l'état actuel de la technique et celui du recuit simulé lors de son introduction. Les travaux menés dans le cadre de cette thèse consistent à comprendre le fonctionnement d'une telle machine et à déterminer dans quelle mesure elle contribue à la résolution de problèmes combinatoires. L'enjeu est de savoir s'il existe ou non une accélération de nature quantique dans ces machines par rapport aux autres ordinateurs. L'idée est de fournir une première étude sur le comportement du recuit quantique lorsqu'il est confronté à un problème connu pour piéger le recuit classique. Le problème de couplage biparti a été choisi spécifiquement pour être difficile à résoudre au moyen d'un recuit simulé. Comparer les performances entre ces deux recuits donne un premier étalon et permet de mieux caractériser leurs potentiels. Nos résultats tendent à monter que, l'obligation de dupliquer les qubits limite fortement la taille des problèmes qui peuvent être intégrés et conduit souvent à des solutions non valables. Cela signifie que l'utilisation opérationnelle d'un recuit quantique nécessite une ou plusieurs étapes de post-traitement. Même si l'informatique quantique analogique présente de nombreux intérêts, nous soutenons qu'à moins d'avoir des interconnexions de qubits beaucoup plus denses, il sera difficile pour cette approche de concurrencer les algorithmes classiques sur les problèmes du monde réel.

Mots-clés : *Informatique quantique, Problèmes d'optimisations, ordinateur quantique adiabatique, Couplage biparti, graphe d'interconnexion des qubits*

UNIVERSITÉ PARIS SACLAY

Abstract

Study of the performance of quantum annealing machines for solving combinatorial problems

by Daniel VERT

Quantum computing is gaining renewed interest with recent announcements from several players. The most obvious reason is that some quantum algorithms can solve in polynomial time the same tasks that are currently considered non-polynomial on classical computers. In recent years, analog quantum machines have appeared, of which the computers currently marketed by the Canadian company D-Wave are the first representatives, operating on a quantum accelerated annealing principle. From an abstract point of view, such a machine can be considered as an oracle specialized in solving a NP-hard optimization problem with an algorithm functionally analogous to simulated annealing but with quantum acceleration. In addition to the formal analogies between simulated annealing and quantum annealing, there is an analogy between the current state of the technique and that of simulated annealing when it was introduced. The work carried out in this thesis consists in understanding the functioning of such a machine and in determining to what extent it contributes to the solution of combinatorial problems. The challenge is to know whether or not there is an acceleration of a quantum nature in these machines compared to other computers. The idea is to provide a first study on the behavior of quantum annealing when confronted with a problem known to trap classical annealing. The bipartite matching problem has been specifically chosen to be difficult to solve using simulated annealing. Comparing the performances between these two annealers gives a first benchmark and allows to better characterize their potentials. Our results tend to show that the obligation to duplicate qubits strongly limits the size of the problems that can be integrated and often leads to invalid solutions. This means that the operational use of quantum annealing requires one or more post-processing steps. Although analog quantum computing has many interests, we argue that unless we have much denser qubit interconnections, it will be difficult for this approach to compete with classical algorithms on real-world problems.

Keywords : *Quantum computing, Optimization problems, Adiabatic quantum computer, Bipartite matching, Qubit interconnection graph*

Table des matières

Listes des Publications

Publications

[VSL19a] Daniel Vert, Renaud Sirdey, Stéphane Louise,
On the limitations of the chimera graph topology in using analog quantum computers, *Proceedings of the ACM International Conference on Computing Frontiers*, 226–229, 2019 (doi: 10.1145/3310273.3322830).

[VSL20a] Daniel Vert, Renaud Sirdey, Stéphane Louise,
Revisiting old combinatorial beasts in the quantum age : quantum annealing versus maximal matching, *Proceedings of the 25th International Conference on Computational Science (Quantum Computing Workshop)*, 473–487, 2020 (doi: /10.1007/978-3-030-50433-5_37).

[VSL20b] Daniel Vert, Renaud Sirdey, Stéphane Louise,
Operational Quantum Annealers are Cursed by Their Qubits Interconnection Topologies, *Proceedings of the IEEE Computer Society Annual Symposium on VLSI (1st International Workshop on Quantum Computing : Circuits Systems Automation and Applications)*, 282–287, 2020 (doi: 10.1109/ISVLSI49217.2020.00058).

[VSL21] Daniel Vert, Renaud Sirdey, Stéphane Louise,
Benchmarking quantum annealing against "hard" instances of the bipartite matching problem, *Special issue for Springer journals (SN Computer Science)*, 1-12, 2021 (doi: 10.1007/s42979-021-00483-1).

Résumés

[VSL19] Daniel Vert, Renaud Sirdey, Stéphane Louise,
Vers l'exploitation de calculateurs quantiques analogiques pour l'optimisation, *Société Française de Recherche Opérationnelle et d'Aide à la Décision (ROADEF)*, Février 2019.

Communications

Poster Daniel Vert, Renaud Sirdey, Stéphane Louise,
Exploitation and limitations of analog quantum computer for optimization, *Fifteenth International Summer School on Advanced Computer Architecture and Compilation for High-Performance and Embedded Systems*, 14-20 July 2019, Fiuggi, Italy.

Séminaire Daniel Vert,
Application à l'optimisation combinatoire, *Direction Scientifique du CEA Tech, Architecture et programmation d'ordinateurs Quantiques*, 16-17 Septembre 2019, Paris Bercy.

Séminaire Daniel Vert,
Résolution de problèmes combinatoires pathologiques par recuit quantique, *Institut Rayonnement-Matière de Saclay (Iramis), Service de Physique de l'Etat Condensé (SPEC), équipe Quantronix*, Salle Itzykson, Bât.774, Orme des Merisiers, Janvier 2020.

Chapitre 1

Introduction

1.1 Contexte

À l'heure actuelle, l'informatique quantique est devenue un sujet de recherche et d'étude incontournable. Cette branche singulière de l'informatique acquière une visibilité avec les annonces majeures de grands acteurs du numérique comme IBM, Google, Intel et même Microsoft. Malgré l'enthousiasme de la communauté il faut appréhender les résultats obtenus avec beaucoup de précaution et de modestie.

Sur le long terme, l'informatique quantique possède un potentiel d'impact significatif sur bon nombre de domaines (scientifique, technologique, industriel, etc.). Bien que l'informatique quantique porte de nombreux espoirs sur la résolution de problèmes pratiques, il n'est pas encore évident de l'affirmer avec certitude. Aujourd'hui, l'informatique quantique permet de résoudre certains problèmes (assez) spécifiques pour lesquels nous ne connaissons pas réellement d'impacts concrets et reste limitée sur les problèmes pratiques. L'objectif d'obtenir une accélération quantique n'est pas encore évident malgré l'annonce d'une première forme de suprématie quantique [19] et le principal enjeu est de déterminer si de telles machines sont capables de rivaliser avec leurs homologues classiques, voire même de les surpasser en terme de puissance de calcul. Une des applications possibles pour ce type de machine provient du fait qu'il existe des algorithmes qui permettent de résoudre en temps polynomial des problèmes dont les meilleurs algorithmes connus sont non polynomiaux. Néanmoins, d'un point de vue expérimental, l'émergence d'ordinateurs quantiques qui seraient capables de rivaliser avec les performances des ordinateurs classiques reste du domaine de la spéculation. Les réalisations actuelles sont encore préliminaires, même si certaines expériences sont impressionnantes au niveau physique (Expérience de Google sur la suprématie [19]), il reste encore à concevoir, réaliser et analyser des machines à plus grandes échelles. Les ordinateurs quantiques sont loin de passer à l'échelle et restent plus du domaine de la recherche fondamentale que de la commercialisation.

1.1.1 Défis du quantique

Les défis de l'informatique quantique sont nombreux, technologique, algorithmique et particulièrement sur la résolution de problèmes. Même si l'apport du calcul quantique reste encore à évaluer, nous pouvons répertorier quatre grandes classes de problèmes par domaines :

- L'optimisation combinatoire avec les problèmes de trajets [126, 120], de coloriage notamment [64, 165, 111].
- Simulation complexe du fonctionnement de la matière au niveau moléculaire [170], la simulation des interactions atomiques [113] et biologiques [163].
- Factorisation de nombres entiers afin de casser les codes de sécurité type RSA [59], l'algorithme de Shor [139, 101].
- L'entraînement des IA neuromorphiques en particulier le Deep Learning [27] requiert une puissance de calcul très importante pour les supercalculateurs afin d'entraîner efficacement des réseaux de taille raisonnable [45].

Il existe bon nombre de domaines d'applications émergeant comme la finance [134], l'assurance, et même l'exploration spatiale [1]. Tous les domaines qui ont des problèmes avec une combinatoire trop importante pour pouvoir être résolu rapidement pourront être touchés par les avancées de l'informatique quantique. L'intérêt principal du calcul quantique est de diminuer l'échelle de temps de calcul qui, à l'heure actuelle, reste encore trop important pour permettre de résoudre efficacement certains problèmes. En pratique, il n'existe pas encore de réponse clairement formulée et identifiée (malgré bon nombre d'annonces spéculatives) qu'auront ces machines sur la résolution de problèmes difficiles (*NP*-difficiles). Pour l'instant, le développement de cette branche se focalise essentiellement sur des problèmes d'ordres techniques (tester les principes de la physique quantique par exemple) et ce sont leurs évolutions technologiques qui poussent à identifier et développer des algorithmes spécifiques plus performants.

1.2 Tours d'horizon

Dans les années 80, le physicien Richard Feynman, avait l'intuition de créer des simulateurs quantiques [80] qui exploiteraient les effets de la mécanique quantique pour simuler leurs phénomènes tandis que cette simulation serait quasiment impossible pour des ordinateurs classiques. Dix ans après cette intuition, les premières briques très expérimentales d'ordinateurs et d'algorithmes quantiques ont été réalisées physiquement. Depuis, la recherche dans le domaine n'a

1. https://www.spacelaw.fr/comment-lordinateur-quantique-va-revolutionner-le-spatial

cessé d'évoluer [2]. Malgré tout, beaucoup d'algorithmes quantiques ne sont pas encore utilisables sur des ordinateurs quantiques, ni même sur des simulateurs et restent des concepts théoriques. En effet, le nombre de qubits disponible reste encore trop modeste (53 qubits d'IBM, 54 qubits de Google) pour que ces algorithmes puissent résoudre des problèmes de tailles non triviales, et ainsi rivaliser avec la puissance de calcul des super-ordinateurs voire les dépasser [21].

1.2.1 La course à la Suprématie Quantique

C'est en 2018 que la notion de suprématie quantique commence à émerger et à devenir une réalité portée par plusieurs acteurs majeurs du domaine tel que Google [3] et IBM. Considéré comme le "*Saint Graal*" de l'informatique quantique, ce palier est atteint lorsqu'un algorithme quantique exécuté sur une implémentation matérielle d'un ordinateur quantique est capable de calculer en un temps raisonnable un résultat hors d'atteinte des ordinateurs classiques [140, 141]. En théorie de la complexité, il nous faut donc trouver un problème qui peut être résolu par l'ordinateur quantique et qui a facteur d'accélération suffisamment important, et pour les meilleurs cas, exponentiel par rapport au meilleur algorithme classique [96, 135]. Pour cela, le nombre de qubits utilisables doit être supérieur au seuil auquel aucun supercalculateur ne pourra être en mesure de gérer la croissance exponentielle de calcul nécessaire pour simuler son équivalent quantique. Néanmoins, à l'heure actuelle, nous ne savons pas s'il existe un moyen algorithmique classique plus efficace pour simuler un système quantique qui évite de reposer sur une taille mémoire qui augmente exponentiellement avec le nombre de qubits mis en jeux.
Dans l'article de R. König et al. [38], les auteurs définissent que, si les ordinateurs quantiques peuvent réellement réaliser des opérations inaccessibles aux ordinateurs classiques, alors ce sera uniquement sur des problèmes très particuliers sans avoir de portée applicative concrète (ex. expérience de Google décrite ci-dessous). Plusieurs tests [4] ont tenté de montrer la supériorité des ordinateurs quantiques face aux algorithmes classiques, mais jusqu'à présent, ils ont rapidement été rattrapé par des algorithmes plus optimisés [105] et plus performants.

C'est en 2017 que Google a annoncé avoir démontré la suprématie quantique en résolvant le problème d'échantillonnage avec un réseau de 49 qubits supraconducteurs [53]. En octobre de la même année, IBM fait la démonstration d'une simulation de 56 qubits sur un supercalculateur et

2. Le *Quantum Algorithme Zoo* `http:/quantumalgorithmzoo.org/` offre une soixantaine d'algorithmes plus ou moins aboutis.

3. Le terme de suprématie a été inventée en 2011 par John Preskill dans un communiqué au Congrès de Solvay [140] mais le concept d'avantage en informatique quantique remonte aux propositions de Yuri Manin en 1980 [119] et de Richard Feynman en 1981 [80]

4. Parmi ces tests, nous pouvons citer la proposition d'échantillonnage de bosons d'Aaronson et Arkhipov [3], les problèmes spécialisés de D-Wave [106] et l'échantillonnage de nombres aléatoires certifiés [4].

permet d'augmenter le seuil théorique limite du nombre de qubits nécessaires pour affirmer obtenir une suprématie quantique [137]. En 2018, des travaux théoriques montrent que la suprématie quantique devrait être prochainement atteignable. Pour cela, les auteurs montrent que sur un réseau bidimensionnel de 7×7 qubits, si les taux d'erreur sont suffisamment faibles [31], alors l'algorithme quantique peut surpasser un algorithme optimisé.
Enfin, le 20 septembre 2019, le *Financial Times* affirme que *"Google prétend avoir atteint la suprématie quantique avec un réseau de 54 qubits dont 53 fonctionnels, qui a été utilisés pour effectuer une série d'opérations en 200 secondes au lieu de 10 000 ans avec un supercalculateur"*. Le 23 octobre, Google a officiellement confirmé ces affirmations [19] et IBM a directement répondu en suggérant que certaines affirmations étaient excessives et laisse entendre que leur calcul prendrait 2.5 jours au lieu de 10 000 ans [138] sur un supercalculateur. Même si Google a montré expérimentalement une suprématie quantique il n'en reste pas moins que le problème est artificiel et peu utile. La controverse vient du fait que dans la définition de la suprématie quantique, il serait souhaitable de résoudre un problème qu'un supercalculateur ne peut pas résoudre et il serait également souhaitable que le problème traité soit utile. Néanmoins, la course entre quantique et classique a permis de faire un bond en avant dans le développement d'algorithmes.

1.2.2 Un autre genre d'ordinateur quantique

Contrairement aux ordinateurs d'IBM et de Google, la société canadienne D-Wave commercialise et utilise un autre type d'architecture bien particulier. L'ordinateur est basé sur un procédé de recuit quantique (détaillé au chapitre 3) qui consiste à préparer un hamiltonien initialement prévu pour connecter les qubits entre eux. Cet hamiltonien possède un état choisi qui permet de définir un problème donné. Le processus de recuit quantique va faire évoluer adiabatiquement l'hamiltonien du système simulé par la machine (un verre de spin) d'un état initialement connu vers l'hamiltonien final dont la configuration d'énergie minimale correspond à la solution d'un problème à résoudre. Ce type d'ordinateur est théoriquement capable de résoudre expérimentalement des problèmes de la classe NP-complets (classe que nous allons détailler au chapitre 2) tel que le voyageur de commerce ou encore le problème du sac à dos.

1.3 Problématique et enjeux de la thèse

L'objectif de cette thèse est d'évaluer expérimentalement les performances que peuvent réaliser les ordinateurs quantiques analogiques qui utilisent une forme de recuit adiabatique. L'enjeu est de savoir s'il existe ou non un facteur d'accélération quantique dans ces machines par rapport aux

autres ordinateurs classiques. Plus précisément, nous nous sommes posés la question suivante : le type d'algorithme mis en œuvre dans les machines à recuit quantique (comme nous l'avons vu à la section précédente) possède-t-il une accélération en terme de temps et possède t-il des performances de calcul supérieures pour la résolution de problèmes difficiles par rapport à son homologue classique, le recuit simulé ?
Tout d'abord la question est de savoir dans quelle proportion la machine est capable de résoudre un problème ayant un nombre de connexions supérieur aux nombre de connexions disponibles dans la topologie de cette architecture ? Afin de pouvoir répondre à cette question nous nous sommes orientés sur l'élaboration d'algorithmes pour intégrer des problèmes à forte densité de variables afin de connaître la limite intrinsèque de la connectivité dans la machine.
Le deuxième enjeu est de pouvoir comparer les capacités d'une machine à recuit quantique avec un recuit simulé sur un problème difficile pour lui mais de nature polynomiale (facile à résoudre). En effet, la plupart des études montrent des résultats sur des problèmes difficiles et n'apportent pas de réponses claires quant aux performances de la machine. De plus, les comparaisons faites avec les solutions sur un recuit quantique sont comparées avec des solveurs avancés (recherche tabou, simplexe, etc.). Dans ce contexte, notre objectif est de mieux appréhender la comparaison des résultats en se focalisant sur le même type d'algorithme (le recuit) et de résoudre un problème qui est facile en apparence mais difficile pour le recuit. Nous cherchons alors à répondre à la question : dans quelle mesure l'ordinateur quantique adiabatique arrive-t-il à obtenir la solution optimale alors que son homologue classique est facilement piégé ? Comparer les performances entre ces deux recuits dans le pire cas nous donne un premier élément de réponse sur l'existence d'une accélération quantique par rapport au recuit classique. Enfin, savoir si ces machines avec plus de 2000 qubits sont capables de résoudre efficacement des problèmes concrets et connaître leurs performances permet de mieux caractériser leurs potentiels sur des problèmes de plus grandes tailles que ceux adressables pour l'instant.

1.3.1 Plan du manuscrit

Ce manuscrit est organisé de manière synthétique ci-dessous suivant le contenu des chapitres.

Le chapitre 2 dresse un panorama partant de la physique quantique jusqu'à l'ordinateur quantique tel que nous le connaissons aujourd'hui. Pour pouvoir appréhender le chapitre dédié à l'ordinateur quantique analogique, il nous faut d'abord expliquer les principes qui régissent ces machines. Ce chapitre décrit aussi les classes de complexité qui sont associées aux problèmes que la machine devrait pouvoir résoudre et les outils qui sont aujourd'hui disponibles pour exploiter efficacement les machines.

Le chapitre 3 présente en détail le fonctionnement interne et le graphe d'interconnexion des qubits dans les machines quantiques analogiques telles qu'elles sont vendues par la société Canadienne D-Wave. À partir des principes de base nous pouvons appréhender différents types de problèmes potentiellement transposables sur l'architecture de la machine.

Le chapitre 4 fait tout d'abord un état de l'art de la capacité des ordinateurs quantiques analogiques à résoudre des problèmes. Il présente également la polémique autour de ces machines. Enfin, nous présentons l'algorithme classique fonctionnellement analogue au recuit quantique, le recuit simulé, introduit dans les années 80.

Le chapitre 5 examine une nouvelle approche pour résoudre des problèmes QUBO denses en ne faisant appel qu'une seule fois au recuit quantique. L'approche la plus conventionnelle est de venir intégrer les variables du problème sur plusieurs qubits, mais cette approche nécessite de nombreuses invocations du recuit afin de résoudre des problèmes de petites tailles.
Une autre approche consisterait à étudier l'existence de relaxations creuses conformes à la topologie d'interconnexion des qubits de la machine. Et dans un second temps de déterminer un sous-ensemble de coefficients non nuls qui offre toujours des solutions de bonne qualité au problème d'origine. Nous allons détailler un algorithme afin de déterminer si de telles relaxations pratiques existent ou non, et plus précisément, si elles sont faciles à trouver. Nos expériences suggèrent que ce n'est pas le cas et, par conséquent, que la résolution de problèmes arbitraires, même de taille modérée, avec un seul appel à un recuit quantique n'est pas possible, du moins dans les limites de la topologie.

Le chapitre 6 étudie expérimentalement le comportement des ordinateurs quantiques analogiques lorsqu'ils sont confrontés à des instances du problème de couplage de cardinalité maximale spécifiquement conçue pour être difficiles à résoudre au moyen d'un recuit simulé. Nous testons un processeur "Washington"(2X) avec 1098 qubits opérationnels sur différentes tailles d'instances pathologiques et nous observons que pour tous ces cas, sauf les plus triviaux, il ne parvient pas à obtenir une solution optimale. Ainsi, nos résultats suggèrent que le recuit quantique, au moins tel qu'il est mis en œuvre dans les machines types D-Wave, tombe lui aussi dans les mêmes pièges que le recuit simulé. Cela nous fournit donc des preuves supplémentaires allant dans le sens de l'existence de problèmes polynomiaux qui restent difficiles à résoudre avec certains algorithmes pourtant efficaces en pratique.

Le chapitre 7 examine dans quelle mesure la topologie d'interconnexion des qubits explique les résultats expérimentaux du chapitre 6. Nous fournissons des éléments de réponses sur le

fait que la topologie conduit à des problèmes de taille artificiellement importante et peut en partie expliquer les observations mentionnées ci-dessus. Nous pouvons donc envisager que des topologies d'interconnexions plus denses permettraient de réaliser le potentiel de l'approche par recuit quantique sur ce problème.

Le chapitre 8 conclut ce manuscrit par un résumé des différentes contributions et des résultats obtenus avant d'ouvrir la discussion sur des perspectives de recherche à venir sur le sujet.

Première partie

Contexte et préliminaires

Chapitre 2

De la physique à l'informatique quantique

2.1 Introduction

La mécanique quantique est aujourd'hui l'un des modèles physiques le plus prédictif et le plus étendu qui existe. Elle permet de décrire précisément la nature à l'échelle microscopique, les interactions entre des atomes, les particules élémentaires et peut-être même les constituants ultimes de la matière [17]. À l'autre extrême, elle est nécessaire pour les assemblages moléculaires et jouerait un rôle prédominant aux plus grandes échelles de l'univers (fluctuations quantiques de l'univers primordial [42]) [1]. De plus, l'utilisation indirecte de la mécanique quantique dans notre société permet de nombreuses applications directes. L'ordinateur résulte de la compréhension du transport de charge dans les semi-conducteurs, les lasers sont directement liés aux échanges entre la matière et le rayonnement, le GPS est calibré sur des horloges atomiques, l'IRM découle de la mesure de l'excitation des spins nucléaires dans les champs magnétiques produits par les bobines supraconductrices, etc.

La puissance de calcul des ordinateurs quantiques a fait l'objet d'études approfondies depuis l'observation de Feynman [80] en 1982. Deutsch [61] en 1985 a été le premier à établir une base solide en introduisant un modèle entièrement quantique pour le calcul et en donnant la description d'un ordinateur quantique numérique (universel). Mais vraisemblablement, la preuve la plus importante du potentiel de puissance des ordinateurs quantiques vient de la découverte par Shor d'un algorithme quantique, en temps polynomial [157] en l'occurence, en $O(N^2)$. Cet algorithme permet de trouver les facteurs premiers des nombres composés et de calculer le logarithme discret. Des résultats plus récents incluent l'algorithme quantique de Grover [90] qui s'applique sur les problèmes de recherches non triés et offre une accélération quadratique en $O(\sqrt{N})$

1. La mécanique quantique s'applique sur une échelle de presque 54 ordres de grandeur en taille, et reste aujourd'hui assez précise pour déterminer des grandeurs avec des précisions pouvant parfois dépasser les 10^{-12} comme par exemple les mesures de transitions fines électromagnétiques pour certains atomes, et en particuliers celles qui sont utilisées pour la définition moderne du mètre.

sur les algorithmes de recherche exhaustive. Enfin l'algorithme quantique de Hallgren [95] pour l'équation de Pell est exponentiellement plus rapide que n'importe quel algorithme classique connu.

2.2 Notions de mécanique quantique

La mécanique quantique est décrite par des concepts et principes qui sont contraires à l'intuition courante. Ce domaine de la physique se base sur des superpositions d'états, des interférences entre atomes (voir d'un atome avec lui même) et le principe d'intrication de particules pour former un système lié à des distances théoriquement infinies. La base mathématique est un espace Hilbertien dont la taille augmente exponentiellement avec le nombre de systèmes élémentaires considérés via le produit tensoriel.

2.2.1 Rappels mathématiques

Espace vectoriel des fonctions d'ondes La fonction d'onde décrit l'état spatial d'un objet quantique. Si l'espace est à trois dimensions paramétrées par la position $x, y, z \in \mathbb{R}$, alors la fonction d'onde donne :

$$\psi : (x, y, z) \in \mathbb{R} \rightarrow \psi(x, y, z) \in \mathbb{C} \tag{2.1}$$

L'ensemble des fonctions d'ondes noté H forme alors un espace vectoriel à valeurs complexes. En effet, en prenant deux fonctions d'ondes $\psi_1, \psi_2 \in H$, la somme et le produit par une constante complexe appartiennent aussi à cet ensemble : $\phi = \psi_1 + \psi_2 \in H$ et $\theta = \lambda\psi_1 \in H$. En notation de **Dirac**[2], les deux équations ci-dessus donnent : $|\phi\rangle = |\psi_1\rangle + |\psi_2\rangle$ et $|\theta\rangle = \lambda\,|\psi_1\rangle$.

Produit scalaire Pour distinguer quantitativement deux fonctions d'ondes, nous introduisons le produit scalaire entre ces deux fonctions. Le produit scalaire Hermitien[3] de deux fonctions d'ondes $|\psi_1\rangle$ et $|\psi_2\rangle$ est le nombre complexe noté $\langle\psi_1|\psi_2\rangle$ défini par :

2. Cette notation introduite par Paul Dirac dans les années 30 [65] facilite l'écriture des équations en mécanique quantique et caractérise l'aspect vectoriel représentant l'état quantique.

3. Par définition, tout un produit scalaire Hermitien $\langle\psi_1|\psi_2\rangle \in \mathbb{C}$ sur un espace vectoriel H doit vérifier les propriétés suivantes :
   - $\langle\psi_2|\psi_1\rangle = \langle\bar{\psi_1}|\,|\psi_2\rangle$ complexe conjugué
   - $\langle\psi_1|\lambda\phi + \mu\theta\rangle = \lambda\,\langle\psi_1|\phi\rangle + \mu\,\langle\psi_1|\theta\rangle$ linéarité
   - $\langle\psi|\psi\rangle \geq 0$

$$\langle \psi_1 | \psi_2 \rangle = \int_{\mathbb{R}^3} \bar{\psi_1}(x,y,z)\psi_2(x,y,z)dxdydz \tag{2.2}$$

où $\bar{\psi_1}(x,y,z)$ est le nombre complexe conjugué de $\psi_1(x,y,z)$. La notation de Dirac $\langle \psi_1 | \psi_2 \rangle$ du produit scalaire fait intervenir $\langle \psi_1 |$ qui corresponds au conjugué de la fonction ψ_1. Comme en géométrie Euclidienne, nous pouvons interpréter le produit scalaire comme la composante du vecteur $|\psi_2\rangle$ projetée orthogonalement sur le vecteur $|\psi_1\rangle$. Les deux fonctions sont orthogonales si et seulement si $\langle \psi_1 | \psi_2 \rangle = 0$. De plus, $\|\psi\|^2 = \langle \psi | \psi \rangle$ définit la norme au carré de la fonction ψ et la fonction s'écrit :

$$\|\psi\|^2 = \langle \psi | \psi \rangle = \int_{\mathbb{R}^3} |\psi(x,y,z)|^2 dxdydz \tag{2.3}$$

alors la norme $\|\psi\|^2$ représente la surface sous la courbe positive de $|\psi(x,y,z)|^2$. Le vecteur est normalisé si $\|\psi\| = 1$, cela signifie que le vecteur est de longueur 1. Cette notion est essentielle pour interpréter $|\psi(x,y,z)|^2$ comme une densité de probabilité de présence lors de la détection de la particule. Il faut remarquer qu'il y a des fonctions dont la norme est infinie (par exemple pour une onde plane). L'espace des fonctions pour lesquelles la norme est finie est appelé l'espace de Hilbert H des fonctions d'ondes (dans la suite de la section, l'espace de Hilbert de spin $1/2$ est de dimension finie $N = 2$ pour les qubits).

L'équation d'évolution L'équation de Schrödinger exprime l'évolution dans le temps de la fonction d'onde quantique $\psi(x,t)$ de la particule. Cette équation donne la loi d'évolution du vecteur $|\psi(t)\rangle$ dans l'espace de Hilbert :

$$\frac{d\,|\psi(t)\rangle}{dt} = -i\hbar\hat{H}\,|\psi(t)\rangle \tag{2.4}$$

L'Eq. 2.4 donne précisément la modification à tout instant de l'onde et cette modification dépend de la masse de la particule mais aussi des forces qu'elle subit à travers la fonction potentiel $V(x)$ [4]. L'opérateur hamiltonien $\hat{H}$ est un générateur de l'évolution temporelle. Enfin, c'est une équation linéaire. Si nous connaissons l'évolution de $|\psi_1(t)\rangle$ et de $|\psi_2(t)\rangle$, alors la somme $|\phi(0)\rangle = |\psi_1(0)\rangle + |\psi_2(0)\rangle$ évolue comme la somme des évolutions : $|\phi(t)\rangle = |\psi_1(t)\rangle + |\psi_2(t)\rangle$. Cette propriété est le principe même de la superposition (développé dans la section suivante).

4. En notation de fonction d'onde, l'Eq. 2.4 se développe par exemple dans un cadre simplifié unidimensionnel : $i\hbar\frac{\partial\psi}{\partial t}(x,t) = -\frac{\hbar^2}{2m}\frac{\partial^2\psi}{\partial^2 x}(x,t) + V(x)\psi(x,t)$ avec m la masse, $V(x)$ le potentiel et $\hbar$ la constante de Planck

2.2.2 Principes physiques

Superposition Un état quantique est une combinaison linéaire à coefficients complexes, de norme un, d'états observables [5]. Cette notion résulte du premier postulat de la mécanique quantique qui énonce : *"quel que soit l'état d'un système quantique il est représenté par un vecteur dans son espace vectoriel d'Hilbert"*. La conséquence est que l'état d'une quantité doit être représenté comme une somme dans l'espace de Hilbert en dimension fini de N vecteurs et chaque vecteur représente une quantité observable dans cet espace. En notation de Dirac, la superposition d'un état quantique $|\Psi\rangle$ s'écrit [91] :

$$|\Psi\rangle = c_0 |\alpha_0\rangle + c_1 |\alpha_1\rangle + ... + c_{N-1} |\alpha_{N-1}\rangle + c_N |\alpha_N\rangle$$

Les c_i sont les coefficients complexes de la combinaison linéaire de l'état global $|\Psi\rangle$, et les $|\alpha_i\rangle$ représentent les vecteurs de la base dont dépend l'observable. En effectuant une mesure, l'interprétation habituelle de la mécanique quantique (dite interprétation de Copenhague [98]) postule que le vecteur possédant l'ensemble des états possibles est projeté sur un des vecteurs de la base et l'état $|\Psi\rangle$ devient alors un état classique. Dans les calculateurs, il n'est pas nécessaire d'avoir plus de deux états à contrôler. En effet, seulement 2^N sont nécessaires pour effectuer des calculs pour N qubits. Pour un qubit seulement deux états distincts sont mesurables (Par exemple : le sens de magnétisation du spin d'un électron, la polarisation d'un photon, la fréquence d'un courant oscillant dans un supraconducteur, etc).

Pour qu'un calcul soit théoriquement plus efficace qu'avec un ordinateur classique, la superposition d'états [130] doit avoir un espace de Hilbert de dimension N (voir rappel mathématique). L'intérêt est d'avoir un mot [6] possédant un maximum de qubits pour obtenir un grand nombre d'états superposés. Cette propriété importante permet aux calculateurs quantiques de paralléliser les calculs à un niveau **théoriquement** inégalable avec les meilleurs supercalculateurs.

Intrication Quand des particules sont dit intriqués, elles ont comme propriétés de partager la même fonction d'onde, même à des distances arbitrairement éloignées [152]. Cette propriété leur confère un état quantique corrélé sans être pour autant identique [7].

5. Pour chaque quantité mesurable (spin, position, quantité de mouvement, etc.) lui correspond un opérateur observable [71].
6. i.e. assemblage de plusieurs qubits
7. Principe de non-localité (1935) [72].

Quand deux (ou plusieurs) qubits sont intriqués, une mesure effectuée sur l'un aura instantanément l'effet sur l'autre sans attendre un délai de transmission d'information entre eux. Alain Aspect (1982) a démontré dans [20] qu'agir sur un photon aura une conséquence directe sur l'autre. En informatique quantique numérique, le principe d'intrication est utilisé à l'aide de portes quantiques à plusieurs qubits (deux ou trois) pour les connecter entre eux et avoir un état quantique commun. Deux qubits non intriqués sont dans deux états quantiques à deux dimensions alors que deux qubits intriqués sont représentés par un état à quatre dimensions. Cette intrication est associée à une impossibilité de dissocier les qubits jusqu'à la mesure (ou tout phénomène irréversible sur la fonction d'onde).

Par exemple, en prenant un système à deux qubits $\{Q_1, Q_2\}$ décrit par un vecteur d'état, l'état est un vecteur dans l'espace de Hilbert $H_1 \otimes H_2$ [8]. Alors l'ensemble des états s'écrivent sous la forme du produit tensoriel entre Q_1 et Q_2 comme :

$$|\Psi_{1+2}\rangle = |\Psi_1\rangle \otimes |\Psi_2\rangle \tag{2.5}$$

L'Eq. 2.5 montre bien des états séparables car $|\Psi_1\rangle \otimes |\Psi_2\rangle = |\Psi_1\rangle |\Psi_2\rangle$. En effet, les systèmes Q_1 et Q_2 sont dans un état quantique strictement identifié. Alors, agir sur Q_2 n'influence en rien l'état du système Q_1. En revanche, un état intriqué est un état non séparable qui s'écrit sous la forme $|\Psi_{1+2}\rangle = \alpha |\psi_1\rangle |\psi_2\rangle + \beta |\phi_1\rangle |\phi_2\rangle$. En prenant une base de l'espace d'Hilbert H_1 tel que $\{|+\rangle_1, |-\rangle_1\}$ et H_2 de la forme $\{|+\rangle_2, |-\rangle_2\}$, l'état séparable est donné par :

$$|\Psi_{sep}\rangle = \frac{1}{\sqrt{2}}(|+\rangle_1 |-\rangle_2 - |-\rangle_1 |-\rangle_2) = \frac{1}{\sqrt{2}}(|+\rangle_1 - |-\rangle_1) \otimes |-\rangle_2 \tag{2.6}$$

Alors que l'état intriqué s'écrit :

$$|\Psi_{int}\rangle = \frac{1}{\sqrt{2}}(|+\rangle_1 |-\rangle_2 - |-\rangle_1 |+\rangle_2) \tag{2.7}$$

Quand le système est intriqué, l'état d'un qubit n'est pas clairement défini, c'est l'état de l'ensemble du système qui est défini et qui évolue au cours du temps. Lors de la mesure, son état choisit un état classique avec une amplitude de probabilité donnée par les coefficients (α, β) de l'état intriqué. Quand le qubit est mesuré, l'état de l'autre qubit est donné par corrélation. La principale caractéristique de l'état 2.7 est une corrélation des mesures réalisées sur Q_1 avec les mesures réalisées sur Q_2. Par exemple, nous mesurons l'état de Q_1 dans la base $+/-$ avec comme résultat $+$ (dans l'Eq. 2.7, nous avons 50% de chance), le système total est projeté dans l'état

8. le symbole $\otimes$ représente le produit vectoriel entre les deux espaces.

$|+\rangle_1 |-\rangle_2$ de telle façon que la mesure de Q_2 donnera –. Ainsi, un système intriqué forme un ensemble, qui n'est pas séparable en deux systèmes indépendants tant qu'il est intriqué.

Effet tunnel De par sa nature corpusculaire et ondulatoire, la matière à une probabilité non nulle de traverser d'importantes barrières énergétiques [122]. A titre d'exemple considérons deux métaux conducteurs que nous rapprochant pour piéger des électrons.

En plaçant deux métaux à l'équilibre aussi proche l'un de l'autre que possible, ce dispositif constitue une barrière de potentiel pour leurs électrons. L'énergie est inférieure à la hauteur de la barrière que représente l'espace entre les deux métaux. Le principe de ce dispositif est alors de rapprocher suffisamment près deux métaux conducteurs de telle sorte que les fonctions d'ondes des électrons puissent se recouvrir. Ainsi, la barrière de potentiel n'est plus infranchissable pour les électrons. L'effet tunnel de nature purement quantique possède une forte probabilité de se produire [9].

L'application d'une différence de potentiel noté V_p entre les deux métaux conducteurs permet de faire circuler le courant. Alors, selon l'approximation WKB [10], le courant tunnel est proportionnel à la probabilité de transfert $P_{1\to2}$ des électrons du conducteur c_1 vers un conducteur c_2. En définissant d, la distance de séparation, E l'énergie d'un électron par rapport à l'énergie de Fermi [11] du conducteur c_1 et $\bar{\phi} = (\phi_1 + \phi_2)/2$ avec ϕ_1 et ϕ_2 représente le travail de sortie des électrons (l'énergie nécessaire pour arracher un électrons de la surface), alors la probabilité d'obtenir l'effet tunnel est :

$$P_{1\to2} = exp\left[-2d\sqrt{\frac{2m}{\hbar^2}\left(\bar{\phi} - E - \frac{eV_p}{2}\right)}\right] \tag{2.8}$$

L'effet tunnel est un processus physique qui conserve l'énergie de la particule transféré (l'énergie totale d'un système est conservée). Pour des électrons avec une tension de polarisation $eV_p \ll \bar{\phi}$ et avec une énergie proche de l'énergie de Fermi $E \simeq 0$, la probabilité de passage devient : $P_{1\to2} = exp(-2kd)$ avec k est la constante d'atténuation du vide ($k = \sqrt{2m\bar{\phi}/\hbar^2}$) et vaut 10 nm^{-}1. L'électron possède alors une probabilité de passage d'environ 80% alors que pour un proton ($m_p \simeq 2000m_e$), la probabilité passe à 10^{-19} à cause de l'effet de masse. Ainsi, l'effet

9. Dans cette configuration, il existe un courant tunnel non nul au travers duquel les électrons peuvent circuler.
10. L'approximation vient des inventeurs Wentzel, Kramers et Brillouin et est une méthode permettant une résolution approchée de l'équation de Schrödinger. Le but de cette approximation est de retrouver le régime classique lorsque $\hbar$ tend vers 0.
11. L'énergie de Fermi désigne l'énergie du plus haut état quantique occupé dans un système par des fermions (ici des électrons) à 0 Kelvin.

tunnel utilisé dans les ordinateurs quantiques dépend de la largeur de la barrière (du puits de potentiel) mais aussi du type de qubits utilisés.

2.3 Les qubits

Les ordinateurs classiques sont constitués d'un ensemble de circuits électroniques élémentaires construit à partir de transistors. Observer l'état d'un *cbit* [12] revient à déterminer une représentation physique de ses deux états. La représentation la plus fréquente consiste à utiliser deux potentiels électriques qui ne se chevauchent pas pour les états 0 et 1 [13] et est entièrement déterministe. Malgré tout, des erreurs peuvent survenir lors du calcul et sont corrigées via des systèmes de correction utilisant de la redondance.
En revanche, les qubits sont les éléments de base des ordinateurs quantiques et sont fonctionnellement différents des cbits [153]. Les équations déterministes que nous connaissons bien deviennent des équations probabilistes et les principes qui sont propres aux qubits leurs permettent d'être dans un état de superposition entre $|0\rangle$ et $|1\rangle$ [14].

Candidat pour être un qubit Les défis à surmonter pour concevoir technologiquement un calculateur quantique opérationnel restent encore complexes et tout objet quantique n'est pas nécessairement un bon candidat au titre de qubit. D. DiVincenzo [67] donne cinq critères qui doivent être satisfaits par tout candidat au titre de qubit :

1. **Qubits à deux états :** Chaque qubit doit pouvoir se trouver dans deux états bien distincts, les autres états possibles doivent posséder des densités de probabilités négligeables,
2. **Qubits initialisable :** Chaque qubit doit être manipulable pour être mis dans un état de base bien défini,
3. **Portes quantiques :** Chaque qubit doit pouvoir être utilisé pour construire un jeu de portes quantiques universelles i.e. en nombre et variété suffisante pour préparer n'importe quel type d'état quantique (ce critère n'est pas forcément inclus dans les ordinateurs de type D-Wave),
4. **Etat mesurable :** Chaque qubit doit pouvoir être mesuré de manière individuelle à la fin du calcul,

12. cbit pour "*classical bit*", un bit dans un état 0 ou 1
13. Un bit est de valeur 1 si le courant passe et 0 sinon.
14. 0 est l'état fondamental ("ground state") et 1 l'état excité ("excited state").

5. **Temps de cohérence long :** Le temps de cohérence [15] doit être suffisamment long par rapport au temps que nécessite le calcul. Nous parlons de décohérence intrinsèque quand les états superposés retombent dans un état classique avant la fin du calcul, et de décohérence environnementale quand les qubits subissent une perturbation extérieure. Pour le calcul quantique numérique, ce point est central, il faut garder la cohérence des états de grande taille afin que l'algorithme puisse exécuter l'ensemble des tâches. Pour le calcul quantique adiabatique il faut que le temps de cohérence des qubits soit suffisamment long par rapport au temps que nécessite la durée du processus adiabatique (que nous détaillerons au Chapitre 3) afin de respecter autant que possible les conditions du théorème adiabatique.

2.3.1 Sphère de Bloch et règle de Max Born

Dans un modèle probabiliste classique défini sur $\mathbb{R}$, un bit a une probabilité p d'avoir la valeur 0 et $1-p$ d'avoir la valeur 1. En revanche, pour un qubit le modèle mathématique défini sur $\mathbb{C}$ peut-être représenté comme une sphère de Bloch (schématisé sur la figure 2.1). Ce modèle représente l'état d'un qubit avec un vecteur à deux dimensions et dont la norme est de 1 (voir rappel mathématique). Ce modèle permet au qubit d'être encodé sur une infinité d'états distincts entre $|0\rangle$ et $|1\rangle$. Les états intermédiaires sont des vecteurs de longueur 1 avec un angle θ par rapport à z et un angle ψ par rapport à x. Alors, les états $|0\rangle$ et $|1\rangle$ représentent les états de sortie à la mesure.

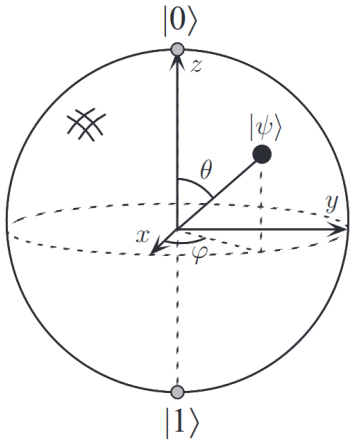

FIGURE 2.1 – Représentation de la shpère de Bloch d'un qubit (tiré du livre "*Quantum Computation and Quantum Information*" [131])

15. le temps pendant lequel les qubits restent en état de superposition

Initialement, l'état du qubit est une superposition de l'état $|0\rangle$ et de l'état $|1\rangle$, le paramètre α^2 donne alors la probabilité d'obtenir l'état $|0\rangle$ et β^2 l'état $|1\rangle$. En réalité, nous avons besoin que de deux angles et un unique module pour représenter deux complexes qui vérifient que la somme des probabilités [16] donne obligatoirement 1 (norme) avec la relation : $\alpha^2 + \beta^2 = 1$ (la fonction d'onde d'un qubit). Cette relation élevée au carré représente alors la densité de probabilité de présence du qubit et la norme est directement reliée à la fonction d'onde de Schrödinger. L'Eq. 2.9 donne la relation entre α et β avec la règle de Max Born :

$$|\Psi\rangle = \alpha\,|0\rangle + \beta\,|1\rangle \tag{2.9}$$

Où $|\Psi\rangle$ représente l'état global du qubit, $\alpha = \cos(\theta/2)$ et $\beta = \exp\{i\phi\}\sin(\theta/2)$ les probabilités de l'état du qubit associées à 0 et 1. Ainsi, lorsque le vecteur d'état est initialisé horizontalement dans la sphère de Block [30], le qubit est dans un état superposant 0 et 1. La possibilité de présence est la même, mais la phase est reliée par le vecteur ϕ dans l'exponentielle. Grâce à cette représentation, l'état d'un qubit peut facilement être initialisé et modifié (via des portes logiques ou d'autres processus, que nous présenterons en détail plus tard). Le qubit est un objet quantique et comme nous l'avons vu tous les objets physiques ne peuvent pas être de bon candidat pour être des qubits.

Il existe un grand nombre de technologie en cours d'exploration pour faire des qubits (supras [17], photons, ions piégés, électrons, etc). Le lecteur désireux de découvrir les voies technologiques peut se référer au document "Status of quantum computer development" [18] qui est un état des lieux de l'informatique quantique à la date de rédaction de ce manuscrit. Il évoque notamment d'autres technologies qui ont, pour l'instant, peu de chances d'aboutir et qui sont toujours en cours d'études. À l'écriture de ce manuscrit, aucune technique n'est véritablement passé à l'échelle pour les ordinateurs numériques (au-delà de 50 qubits) ou ne garantit pas d'être totalement utilisable (pour les analogiques). Chacune d'elles possède leurs avantages mais aussi beaucoup d'inconvénients : la stabilité, le taux d'erreur, la durée de cohérence, la température de fonctionnement, le processus de fabrication, etc.

16. Modèle probabiliste de Max Born en 1926 [37]
17. Cette technique est technologiquement facile à fabriquer et s'appuie sur les techniques de création de circuits CMOS. L'équipe de D. Estève au CEA fait partie des précurseurs de la création de tels qubits supraconducteurs.
18. https://www.bsi.bund.de/EN/Topics/Crypto/Cryptography/QuantumComputing/quantum_computing_node.html

2.3.2 Utilisation des qubits

Une des difficultés en informatique quantique est de faire correspondre les portes quantiques à des opérateurs unitaire et réversible. Toutes portes non réversible ne pourront donc pas être simulé par l'évolution d'un qubit. A la différence des bits classiques, il existe une famille de transformations (unitaire ou binaire) qui permettent de réaliser des opérations logiques sur les qubits [40, 66] :

— La porte X (ou NOT) : $|\Psi\rangle = \alpha|0\rangle + \beta|1\rangle \overset{X}{\Rightarrow} |\Psi_X\rangle = \alpha|1\rangle + \beta|0\rangle$

— La porte Y : $|\Psi\rangle = \alpha|0\rangle + \beta|1\rangle \overset{Y}{\Rightarrow} |\Psi_X\rangle = i\alpha|1\rangle - i\beta|0\rangle$

— La porte Z (ou flip) : $|\Psi\rangle = \alpha|0\rangle + \beta|1\rangle \overset{Z}{\Rightarrow} |\Psi_X\rangle = \alpha|0\rangle - \beta|1\rangle$

— La porte de Hadamard : $H = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$

— Toute les portes de rotations : $|\Psi\rangle = \alpha|0\rangle + \beta|1\rangle \overset{R}{\Rightarrow} |\Psi_X\rangle = \alpha|0\rangle + \beta e^{i\phi}|1\rangle$

De même que la porte **NAND** (NON-ET) qui est universelle [19] en logique classique, il y a des portes universelles pour la logique réversible [20] qui est un cas particulier de la logique quantique. Par exemple la porte de Toffoli [24] : Si deux premiers qubits sont dans l'état $|1\rangle$ elle applique une porte Pauli-X (**NOT**) sur un troisième qubit, sinon 0. Puisqu'elle est l'analogue quantique d'une porte classique, elle est entièrement décrite par sa table de vérité et peut être écrite comme $|a, b, c\rangle = |a, b, c \otimes ab\rangle$.

Il en existe un grand nombre permettant d'appliquer des opérations logiques sur les qubits : La porte **SWAP** qui effectue une inversion sur deux qubits, les portes Contrôlés (**cX cY cZ**) qui font un **NOT** que sur le second qubit quand le premier est dans l'état $|1\rangle$, la porte d'Ising (ou porte **XX**), etc. Plus un système contient de portes, plus l'algorithme mis en œuvre devient complexe.

2.4 Les classes de complexités

Maintenant que nous avons détaillé les considérations purement quantique, nous allons nous concentrer sur la résolution des problèmes d'optimisation dans cette section.

19. Le principe d'une fonction universelle est de pouvoir être exprimée avec des fonctions basiques tel que OUI, NON, OU, ET.

20. Une porte logique P est dite réversible si pour une sortie B, il existe une entrée A qui s'écrit $P(A) = B$. Et inversement, la porte inverse P' doit être de la forme : $P'(B) = A$.

2.4.1 Définition des classes

Dans le cadre de la théorie de la complexité, la classe des problèmes qui nécessite un temps polynomial pour vérifier la solution (classe dite NP) est particulièrement importante. Parmi ces problèmes, nous pouvons distinguer ceux pour lesquel le temps nécessaire pour trouver une solution augmente polynomialement avec la taille du problème (classe dite P). Également, un sous-ensemble particulièrement important de problèmes de NP est appelé NP-complet [115] et a comme propriété que, n'importe quel problème dans NP peut être transformé en problème dans NP-complet en un temps polynomial. Des centaines de problèmes sont connus et il y a un enjeu fort autour de cette thématique : *Si quelqu'un trouve un algorithme de temps polynomial pour un problème dans NP-complet, alors cet algorithme pourrait être utilisé comme programme pour résoudre tous les autres problèmes dans NP en temps polynomial.*

Le fait que personne n'ait encore réussi à trouver un algorithme classique en temps polynomial justifie la difficulté de ces problèmes. Par contre, personne n'a pu prouver qu'un tel algorithme ne peut pas être construit pour un problème de type NP-complet. La question est de savoir : existe-t-il un algorithme qui résout des problèmes de cette classe en temps polynomial ? C'est l'un des questionnement majeur de l'informatique classique (et également une question ouverte pour les ordinateurs quantiques). Est-il possible qu'un problème dans NP-complet soit résolu en un temps polynomial sur une machine quantique ? Ceci a mené à identifier des ensembles d'instances qui sont particulièrement difficiles pour des algorithmes classiques.
Dans le cadre de la théorie de la NP-complétude, nous nous restreindrons aux problèmes de décision, c'est-à-dire ceux dont la solution est soit vraie, soit fausse. La question de l'existence d'un algorithme de résolution de complexité polynomiale nous amène à définir toutes ces classes :

Classe P Cette classe est définie pour des problèmes de décision qui sont résolus efficacement en un temps polynomial avec le nombre de données à traiter N, N étant la taille du problème. Les problèmes liés à cette classe sont dits faciles à résoudre (par exemple : recherches dans des listes, recherche d'un chemin dans un graphe, multiplication de matrices, etc.). Le temps de calcul nécessaire pour résoudre des problèmes de cette classe est simplement proportionnel à N^M avec M un entier ne dépendant pas du problème et ne demande pas de moyen important en terme de calcul (au moins pour les petites valeurs de M). Nous verrons au Chapitre 7 que certains problèmes polynomiaux, comme le problème de couplage biparti, sont en réalité difficile à résoudre suivant l'algorithme utilisé.

Classe NP Cette classe est définie pour des problèmes dont il est aisé de vérifier efficacement la validité d'une solution (une solution valide peut être confirmée en un temps polynomial) mais pas toujours résolue efficacement. En théorie, certains des problèmes associés à cette classe sont plus complexes que les précédents et ils possèdent un temps de calcul au mieux exponentiel lorsque la méthode utilisée est de simplement tester l'ensemble des possibilités. En pratique, ce type de problème est adapté aux ordinateurs quantiques puisqu'ils sont théoriquement dans la capacité de traiter 2^N combinaisons. Les exemples de problèmes sont nombreux et touchent différents domaines, la logistique, la planification, la production, les transports, les télécoms, la finance et surtout la cryptographie.

En 1956 (Kurt Gödel)[21], une polémique se centre sur l'intérêt de savoir si la classe P est égale ou non à la classe NP. Si $P = NP$, cela impliquerait qu'il serait aussi facile de trouver un résultat quand il est possible de simplement le vérifier. A l'heure actuel, aucune démonstration sérieuse ne montre que $P = NP$ et la communauté de la théorie de la complexité pense que cela n'arrivera jamais. La démonstration de cette égalité (ou de sa non-égalité) fait partie de l'un des sept défis mathématiques du *Clay Mathematics Institut* avec pour récompense un prix de 1M$ [63].

Classe NP**-complet** Cette classe de problème est définie comme un sous ensemble des problèmes NP. Une des raisons qui laisse à penser que $P \neq NP$ est l'existence de cette classe. En effet, si un seul problème dans NP-complet est résolu en temps polynomial, alors tous les problèmes dans NP peuvent être résolus en temps polynomial et donc $P = NP$. Mais aucun algorithme polynomial n'a jamais été découvert pour aucun problème dans NP-complet. Les problèmes NP-complets sont, dans un certain sens, les problèmes les plus difficiles à résoudre de NP. Dans cette classe, plus de 3000 problèmes NP-complets sont identifiés dont le problème SAT. Ce problème est défini comme une formule booléenne composée de variables $x_1, ..., x_n$ et de connecteurs (et, ou, non). La question posée est alors existe-t-il une affectation des variables pour laquelle la formule choisie soit vraie ? C'est le premier problème dont la NP-complétude a été démontrée par Cook en 1971 [52]. Théoriquement, il est démontré (théorème de Cook) que si nous pouvions déterminer la solution optimale à un problème de la classe NP-complet, alors, toutes les solutions aux problèmes de cette classe seraient faciles à déterminer (notion de réduction de problèmes).
Il existe d'autres problèmes bien connus : la clique (un graphe donné contient-il un sous-graphe complet ?), le voyageur de commerce (le voyageur de commerce doit faire la tournée d'un ensemble de villes, existe-il un chemin le plus court possible ?), coloriage de graphe (est-il possible

21. Nous savons depuis peu de temps que Gödel a anticipé une partie de la NP-complétude dès 1956

de colorier à l'aide de trois couleurs les sommets d'un graphe de sorte que deux sommets adjacents aient des couleurs différentes ?), le problème du sac à dos (parmi un nombre d'objets prédéfinis, y a t-il une combinaison d'objets tel qu'un sac à dos soit rempli sans dépasser un poids limite ?), etc. Un autre problème bien connu est le démineur, il consiste à localiser des mines cachées sur un terrain avec comme seules indications le nombre de mines dans la zone et le nombre total de mines sur le terrain. Enfin, il est conjecturé que la simulation du repliement de protéines complexes serait un problème NP-complet [92].

De nombreux problèmes ne sont pas des problèmes à proprement dit de décisions, mais bien des problèmes d'optimisation. Pour utiliser ces problèmes dans le cadre de la théorie de la NP-complétude, il suffit le plus souvent de les reformuler sous la forme d'un problème d'optimisation en imposant une borne sur la valeur à optimiser. Nous introduisons la classe NP-difficile (qui sont des problèmes d'optimisation) dans le paragraphe suivant afin de pouvoir présenter au chapitre 3 l'intérêt que peut porter les machines quantiques de type D-Wave censées pouvoir résoudre des problèmes d'optimisation de cette classe.

Classe NP-difficile Cette classe de problèmes correspond au pendant d'optimisation des problèmes NP-complets. Actuellement, la grande majorité des problèmes pratiques se trouve dans cette classe, y compris les problèmes fondamentaux dans de nombreuses disciplines annexes (l'ordonnancement, Sudokus, etc. [18]). Il existe un catalogue des propriétés NP-difficiles [84] qui contient un grand nombre de problèmes datant de 1979 et plusieurs sites Web tiennent à jour les derniers problèmes d'optimisation en date [54]. Cette classe est aussi utilisée pour des problèmes d'optimisation avec comme objectif de chercher un minimum (ou un maximum) global dans un vaste ensemble de solutions. Nous verrons dans le Chapitre sur l'ordinateur quantique adiabatique que cette machine est spécialisée dans la résolution de problème d'optimisation et donc par définition, sur des problèmes de la classe NP-difficile. La classe des problèmes NP-difficile contient aussi de nombreux jeux Nintendo [13] comme Super Mario Bros [22], La Légende de Zelda et même Pokémon.

Pour aller plus loin Certains problèmes NP-complets admettent des cas particuliers qui sont polynomiaux. Pour reprendre l'exemple du voyageur de commerce, cette fois le graphe est limité à un circuit. Alors dans ce problème trivial, il n'y a qu'une solution pourtant le voyageur de commerce est NP-complet. Ainsi, quasiment tous les problèmes NP-complet admettent des cas particuliers polynomiaux. Nous verrons que dans le Chapitre 3 qui présente la machine quantique

22. Bien entendu nous faisons un raccourci en disant que Super Mario est un problème NP-difficile. Précisément, il y a un sous problème du jeu qui peut être généralisé sous la forme d'un problème NP [166].

adiabatique, il y a une limitation topologique sur le graphe, et aujourd'hui personne n'a montré si le problème de référence de la machine sur ce graphe est NP-difficile ou non. Si la complexité du problème de référence sur la topologie de cette machine est bien NP-difficile, alors en théorie, n'importe quel autre problème NP-difficile devrait pouvoir être résolu sur un graphe Chimera.
Bien entendu, il existe d'autres classes de complexité de problèmes qu'il n'est pas nécessaire d'introduire dans ce manuscrit [23] [18].

2.4.2 Classes de complexité quantique

En parallèle, il existe aussi des classes de complexité spécifiques pour les algorithmes quantiques. Nous pouvons alors ajouter une classification des problèmes par niveau de difficulté pour les ordinateurs quantiques même si la correspondance avec les classes ci-dessus est encore tranchée.

Classe BQP (bounded error quantum polynomial time) Cette classe a été introduite dans les années 90 lors de l'apparition des premiers algorithmes quantiques et est définie pour des problèmes qui sont traitables en un temps polynomial avec un ordinateur quantique. Théoriquement, il y aurait sous certaines conditions, une correspondance entre les problèmes BQP et des problèmes P. Les premières analyses ont démontré que la classe P des problèmes polynomiaux est bien dans la classe BQP [25]. C'est dans cette classe qu'est définie une grande partie des algorithmes quantiques bien connus comme Grover et Shor. Ces problèmes sont au centre de la recherche actuelle sur les algorithmes quantiques et sur leurs utilisations. En effet la question est de savoir si $P \neq BQP$? Il est déjà établi que $P \subseteq BQP$ [24] qui lui-même est un sous ensemble (strict ou non) de NP.
Malheureusement, les problèmes connus pour avoir une accélération exponentielle (la factorisation par exemple) sont peut-être dans la classe P. Si ces problèmes n'étaient effectivement pas dans P, alors nous saurions résoudre efficacement et il serait aisé d'affirmer que $BQP \neq P$. De plus, la classe BQP n'appartient pas à NP-complet, elle est entre deux. En fait nous n'avons pas montré que ces problèmes sont dans P (pas d'algorithmes efficaces) et nous n'avons pas montré non plus que se sont des problèmes au moins aussi durs que des problèmes dans NP-complets (intersection vide entre NP-complet et BQP et entre NP-complet et P). Ainsi, le fait qu'il existe l'algorithme de Shor et qu'il soit plus efficace que n'importe quel algorithme classique connu n'est pas une preuve que P est un sous ensemble strict de BQP. En revanche la classe BQP ne contient pas de problèmes des classes NP-difficile/complets et le consensus pense qu'il n'y aura

23. comme les classes L, NL, $EXSPACE$, $NEXPTIME$, EXP, ZPP, SZK, etc.
24. Une autre question ouverte est de savoir si $P \subset BQP$?

jamais de problèmes complets dans BQP. La question de savoir si $BQP = NP$ [1] est quand même ouverte mais reste hautement spéculative.

Une autre question sur laquelle travaillent les théoriciens [1, 144] est de savoir si certains problèmes dans BQP échapperaient à la hiérarchie polynomiale (PH)[25] ce qui démontrerait que les machines quantiques peuvent résoudre des problèmes intrinsèquement inaccessibles aux machines classiques. De récents travaux ont présentés des algorithmes utilisant des oracles qui sont dans BQP mais pas dans PH. Ses travaux tendraient à mettre en évidence que ces algorithmes possèdent un temps de résolution polynomial sur ordinateur quantique mais resteraient insolubles sur un ordinateur classique y compris avec un temps de calcul exponentiel. Comme pour les classes de complexité classique, il existe un grand nombre de classes en quantique [167] (QMA, $QCMA$, BPP, etc.) qu'il n'est pas nécessaire de développer ici.

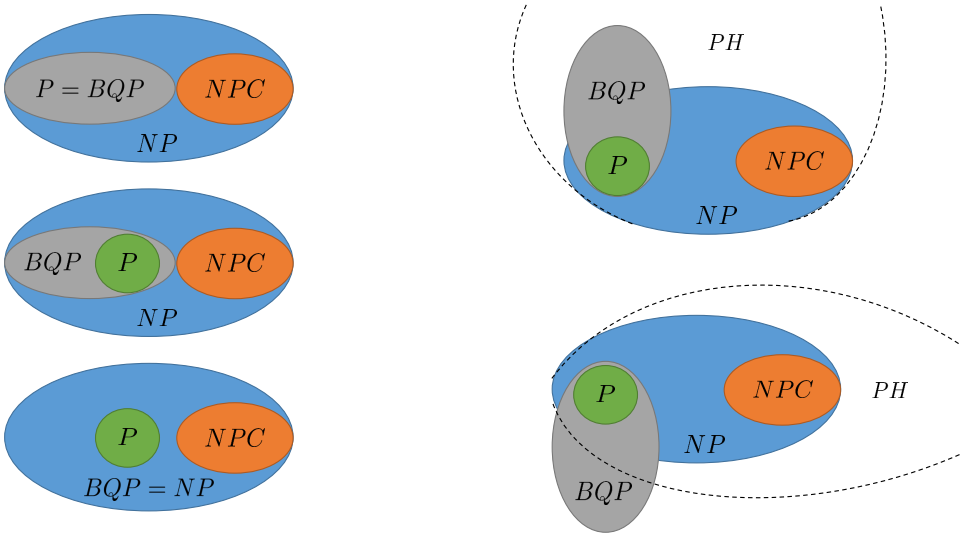

FIGURE 2.2 – Question ouvertes sur les liens entre calcul quantique et théorie de la complexité

Pour aller plus loin Un grand nombre de publications décrivent les limitations des algorithmes et des ordinateurs quantiques. Un problème BQP qui n'est pas dans PH donne un avantage quantique important, mais des problèmes exponentiels pour lesquels l'amélioration apportée par le quantique n'est que quadratique ne modifie pas leur nature initiale [2]. Les problèmes

25. PH signifie "Polynomial Hierarchy" et correspond à la classe des problèmes qui peuvent être résolus par des ordinateurs classiques actuels.

NP-complets et au-delà restent donc inaccessibles aux ordinateurs quantiques. À l'heure où ce manuscrit est rédigé, aucun des deux types d'ordinateurs classique et quantique n'est en mesure de résoudre un problème NP-difficile/complet en un temps raisonnable. Il est plus que probable que les propriétés intrinsèques d'un ordinateur quantique conduisent à des facteurs d'accélérations bien meilleurs pour certains problèmes, même si cela ne se traduit pas par une accélération exponentielle. Par ailleurs, la classe de problème résolu efficacement par des ordinateurs quantiques (les BQP) est connue pour résoudre des problèmes qui ne sont pas (encore) efficacement résolus par les ordinateurs classiques. La factorisation des entiers, par exemple, est connue pour avoir un algorithme efficace sur des ordinateurs quantiques (algorithme de Shor [158]) mais aucun algorithme efficace n'a encore été capable de le résoudre sur des ordinateurs classiques. L'algorithme de Grover [90], recherche et résout en un temps linéaire un ensemble de données (tableau) non triées. En informatique classique il faudrait parcourir l'ensemble du tableau pour déterminer la meilleure solution.

Enfin, la simulation de processus quantique est un autre problème qu'il est possible de résoudre, en théorie, avec des ordinateurs quantiques mais il n'est peut-être pas envisageable de le résoudre avec des ordinateurs classiques. Néanmoins, pour la simulation d'un système quantique, existe-il un algorithme plus efficace que l'algorithme qui consiste à stocker un vecteur dont la dimension est 2^N avec N étant le nombre de qubits ? A l'heure actuelle, les physiciens ne connaissent pas de meilleur moyen algorithmique pour simuler un système quantique que de stoker ce vecteur. Cependant, il pourrait exister (c'est peu vraisemblable mais peut-être) un algorithme qui puisse répondre à la question de la simulation de processus quantique [26]. De plus, à ma connaissance, il n'y a pas de preuve de NP-complétude des questions de décision (un problème dans NP) sur des systèmes quantiques. Il existe peut-être un moyen algorithmique plus efficace de simuler ces problèmes qu'en ayant un algorithme à complexité exponentielle.

2.5 Algorithmes et langages de programmation quantique

L'ordinateur quantique utilise des algorithmes qui devraient être en théorie plus efficaces pour résoudre certains types de problèmes que leurs équivalents traditionnels connus. Mais ces algorithmes sont en nombres restreints et leur performance relative aux algorithmes classiques pas toujours évidente à évaluer. Le fondement même de la création d'algorithmes quantiques

26. Dans l'état des connaissances actuelles, il n'y a pas de preuve qu'il n'existe pas un algorithme qui puisse faire une représentation plus compact de ce vecteur.

(comme celui de Grover développé ci-dessous) est de s'assurer qu'ils soient plus efficaces que leurs équivalents optimisés.

Algorithme de Grover Cet algorithme utilise un oracle pour effectuer une recherche dans un ensemble d'éléments non structurés. Intuitivement, la seule manière de résoudre (efficacement) ce problème est de faire une recherche aléatoire. En pratique, les deux façons de faire requièrent dans le pire cas $O(N)$ itérations avec N le nombre d'éléments dans l'ensemble. Alors si nous cherchons à décrire l'ensemble d'une liste avec n bits, si $N = 2^n$, le temps de calcul croît comme l'exponentiel en $O(2^n)$ dû à la liste non structurée. Pour résoudre plus rapidement, l'algorithme de Grover [90] permet de diminuer le temps de calcul en $O(\sqrt{N})$. Ainsi, avec N assez grand, nous obtenons à l'aide du calcul quantique une accélération quadratique et non exponentielle. De plus, cet algorithme admet une borne inférieure car si le problème n'a pas de structure et est complètement aléatoire, il est alors impossible de le résoudre en un temps inférieur à $O(\sqrt{N})$ sur une machine quantique.
Néanmoins, en prenant le problème SAT (NP–complet d'après le théorème de Cook) à N variables, nous avons donc 2^N affectations possibles pour les variables. L'algorithme de Grover permet de réduire à $\sqrt{2^N}$ affectations. C'est bien une accélération, mais elle est non polynomiale et par rapport à des heuristiques bien choisie pour le problème SAT, Grover est inefficace. En réalité l'algorithme de Grover accélère bien la recherche mais par rapport à une recherche exhaustive [27].
Le principal inconvénient avec l'algorithme de Grover est la complexité exponentielle du problème. S'il existe une exponentielle dans la complexité du problème, Grover pourra être utilisé comme oracle pour tirer une solution quadratiquement plus vite mais ne résoudra pas le problème de l'exponentielle. Contrairement à l'algorithme de Shor qui lui est un schéma algorithmique avec une accélération exponentielle relative aux meilleurs algorithmes connus pour un problème (la factorisation) mais sans borne inferieure.

2.5.1 Classe d'algorithmes quantiques

La classe des algorithmes quantiques est une application directe de l'algèbre linéaire et son utilisation s'appuie sur le calcul matriciel pour modifier l'état des qubits sans pour autant lire leur contenu. Cette propriété intrinsèque au calcul quantique rend difficile la programmation, mais certaines portes quantiques (CNOT, XOR & Co) peuvent simuler (émuler) sur un ordinateur classique le comportement du calcul. Dans le cadre des ordinateurs à recuit, nous verrons que

27. Une recherche exhaustive (ou par force brute) est une méthode algorithmique qui consiste à simplement essayer l'ensemble des solutions possibles (assez peu efficace en pratique).

l'algorithme utilisé est très différent des autres et s'appuie sur un principe physique du *théorème adiabatique* qui est une interpolation entre deux hamiltoniens. De nos jours, quatre catégories d'algorithmes quantiques sont en voie de développement :

- Les algorithmes de recherche qui utilisent ceux de Deutsch-Jozsa [62], Simon et de Grover.
- Les algorithmes basés sur les transformés de Fourier quantiques (QFT) qui utilisent l'algorithme de Shor et développent un phénomène appelé "pompiers-pyromanes" [28]. Actuellement il y a le NIST (National Institute of Standards and Technology) qui développe un concours pour avoir de nouveaux standards de crypto post-quantique [48] et il est vraisemblable que ces standards résisteront aux futures attaques des ordinateurs quantiques bien avant que ceux-ci n'en soient capable.
- Les algorithmes d'optimisation qui cherchent un minimum (ou maximum) dans un système complexe (en recherche opérationnelle notamment).
- Les algorithmes de simulations quantiques qui permettent de simuler les interactions entre des atomes, des interactions dans des structures moléculaires, et d'autres phénomènes quantiques.

Pour la suite du manuscrit nous allons détailler l'un des quatre algorithmes, les autres étant principalement développés pour les ordinateurs quantiques numériques. Le lecteur qui souhaite approfondir les autres algorithmes, peut se référer à plusieurs ouvrages [49, 104, 148]. L'algorithme utilisé dans les ordinateurs quantiques de types analogiques se focalise sur des problèmes d'optimisations et plus particulièrement sur la recherche d'un optimum dans un problème complexe. En théorie, les algorithmes conçus spécifiquement pour les ordinateurs à portes quantiques peuvent être convertis en algorithmes exécutables sur ce type d'ordinateur et réciproquement [7]. Le principe est de faire évoluer dans le temps l'hamiltonien d'Ising suffisamment lentement pour rester dans le cadre du théorème adiabatique mais en changeant le problème d'Ising à un certain instant afin d'être représenté comme une ou plusieurs portes quantiques agissant sur le système [29].

28. Les pyromanes sont ceux qui veulent créer des ordinateurs quantiques capables de casser les clés de sécurité publique type RSA et les pompiers sont les cryptographes qui conçoivent des systèmes de chiffrement post-quantique notamment dans le cadre de la compétition du NIST.

29. Le problème d'Ising peut être retranscrit comme une porte logique ; A partir de ce modèle, nous pouvons faire passer des portes logiques pendant l'évolution dans le temps du système et ainsi modifier l'état du problème étape par étape jusqu'à l'hamiltonien final.

2.5.2 Algorithmes d'optimisation

Il existe une catégorie d'algorithme d'optimisation quantique [30] qui sont conçus pour résoudre des problèmes d'optimisation combinatoire.

Algorithme adiabatique Il convient d'introduire un algorithme d'optimisation qui sera détaillé au Chapitre 3 et au Chapitre 4, le calcul quantique adiabatique. L'idée est de partir d'un système initial facile à préparer et de le faire varier lentement pour obtenir un état fondamental qui code la solution d'un problème. Ce modèle de calcul quantique a été proposé pour la première fois par Farhi et al. [8] comme méthode pour résoudre les problèmes d'optimisation combinatoire dans NP [8, 75]. Les algorithmes quantiques adiabatiques pour les problèmes d'optimisations utilisent généralement des systèmes stochastiques dépendant du temps et sont appelés recuit quantique. Le temps d'exécution asymptotique de ces algorithmes d'optimisation sont difficiles à déterminer et le consensus général sur leurs utilisations n'est pas encore clairement établi [15, 57, 76, 77, 78, 79]. Il convient également de mentionner les articles [124] et [81] sur le recuit quantique, qui était à l'origine une référence à un algorithme d'optimisation classique qui fonctionne en simulant un processus quantique [31].
Dans [146], les auteurs ont montré que les ordinateurs quantiques adiabatiques peuvent exécuter un processus analogue à la l'algorithme de Grover en temps $O(\sqrt{N})$. Les algorithmes quantiques adiabatiques permettant d'obtenir une accélération quadratique pour une classe plus générale de problèmes de processus quantiques [161] sont développés en adaptant des techniques de chaîne de Markov [164]. Enfin, des algorithmes adiabatiques ont été proposés pour plusieurs problèmes spécifiques ; dont le "PageRank" (classification des pages internet) [85] ; (l'apprentissage machine [142] ; la recherche de matrices Hadamard (matrice binaire en calcul quantique numérique) [162] et les problèmes de graphes [82, 83].

Pour implémenter des algorithmes sur des machines quantiques, il faut définir des langages de programmation, des environnements de développement et des logiciels. Les machines quantiques n'étant pas encore totalement opérationnelles, de nombreux industriels (IBM, Microsoft, Rigetti et D-Wave [114]) développent en amont des langages de programmation spécialement destinés pour les futures machines.

30. Comme algorithmes, nous pouvons citer : "Constraint Satisfaction", "Adiabatic Algorithms ", "Gradients, Structured Search, and Learning Polynomials", etc.

31. Nous pouvons faire le parallèle avec le recuit simulé qui est un algorithme d'optimisation classique qui fonctionne en simulant un processus thermique (Voir 4).

Langages de programmation quantique Il existe plusieurs langages de programmation quantique développés à ce jour et bon nombre d'entre eux sont indépendants des architectures matérielles qui existent. Il en existe un très grand nombre, nous ne citerons que les plus connues, QCL (ou Quantum Computation Language) [133], Q Language [26], QFC (et le QPL) [155], QML [14], qGCL (ou Quantum Guarded Command Language) [173], Scaf-fold (de l'Université de Princeton [6]), Quipper [88], QWire [136], etc.
L'offre verticalisée intègre souvent un langage de bas niveau comparable au langage machine, un langage haut niveau le tout dans un *framework open source* exploitable avec des fonctions prédéfinies, un environnement de développement et souvent, une offre d'accès à l'ensemble sur le cloud. À l'heure actuelle, il n'y a que très peu d'industriels qui fournissent un ensemble aussi complet, sauf dans une certaine mesure avec les ordinateurs quantiques de D-Wave. Néanmoins, il existe un grand nombre de plateformes plus ou moins finalisées, notamment "Xanadu quantum cloud", "Liquid" de Microsoft, "Qiskit" de IBM, etc. Dans le Chapitre 3, nous montrerons les outils développés par la société qui commercialise les ordinateurs à recuit quantique.

Chapitre 3

Les ordinateurs à recuit quantique

3.1 Introduction

L'objectif principal de la recherche sur les ordinateurs quantiques comme ceux de D-Wave s'axe sur la capacité de résolutions de problèmes d'optimisation. L'intérêt pratique de ces machines est de pouvoir comparer leurs résultats avec ceux obtenus avec des calculateurs classiques. Jusqu'à présent, très peu de recherches montrent expérimentalement de réels avantages à utiliser les machines quantiques analogiques (hors développement technologiques) et la portée pratique (résolution de problèmes dans certaine classe de complexité) n'est pas encore bien définie. Comme il s'agit d'une toute nouvelle plate-forme de calcul quantique, il reste encore beaucoup à faire avant de vraiment connaître le potentiel pratique d'une telle machine. Néanmoins, il a l'avantage d'exister à une échelle non triviale contrairement aux ordinateurs quantiques numériques et cette gamme d'ordinateurs sont actuellement les seuls calculateurs quantiques commercialisé au monde.

3.1.1 Histoire de D-Wave

La société canadienne D-Wave est implantée à Vancouver et reste la seule société qui commercialise des ordinateurs quantiques analogiques. Bien qu'il s'agisse d'ordinateurs présentant de nombreuses limitations par rapport à leurs homologues universels, ces ordinateurs ont l'avantage d'exister et de permettre de faire avancer la recherche sur les aspects du calcul quantique. De plus, ils permettent de tester une première approche quantique du calcul sur un éventail de problèmes d'optimisation.
La société a été fondée en 1999 et il a fallu 8 ans pour produire un prototype avec une puce de 4 qubits. Deux ans plus tard, elle a vendu son premier ordinateur quantique et a reçu un financement de 1,2 million de dollars d'InQTel (”CIA Investment Fund”) pour poursuivre le développement de leur futur prototype. Entre 2007 et 2009, la société a annoncé trois générations

de prototypes (de 16 qubits [39] à 28 qubits [100]) et en 2010, les paiements de la National Security Agency (NSA) et des Western Intelligence Services (”Western Intelligence Services”) ont permis de finaliser leur toute première puce. Mais c'est en 2011 (en partenariat avec Lockheed Martin), qu'une quatrième génération d'ordinateurs est arrivée sur le marché, avec 128 qubits physiques internes, baptisée ”*D-Wave One*” (”Rainer”) [34]. En 2013, la nouvelle puce ”*D-Wave Two*” (”Vesuvius”) est commercialisée avec une puce de 512 qubits [41], puis en 2015, le ”*D-Wave 2X*” [1] avec 1152 qubits est commercialisé. Enfin en 2017, le ”*D-Wave 2000Q*” [2] est commercialisé (pour un prix de 15 millions de dollars) et possède 2048 qubits physiques utilisables, 5600 coupleurs pour relier les qubits entre eux et près de 128 000 jonctions Josephson [3] afin de lire et stocker la valeur des qubits en sortie. Une prochaine génération [4] est attendue pour 2021 avec plus de 5000 qubits physiques et un décuplement de la connectivité devrait en partie palier les limitations actuelles des connexions entre qubits.

3.1.2 L'ordinateur quantique de D-Wave

Les machines analogiques de D-Wave ont comme principe d'établir des connexions entre des qubits qui sont des boucles supraconductrices en niobium (numéro atomique Z=54) avec des pondérations de couplage basées sur le principe d'un verre de spin, puis de faire évoluer le système en laissant le processus de recuit adiabatique déterminer un état de faible énergie. La modification lente de l'état du système permet au processus de recuit adiabatique de (théoriquement) trouver un minimum énergétique global. Ce minimum doit correspond à la solution recherchée du problème posé et ce processus est dit adiabatique car il s'effectue de façon réversible sans qu'il n'y ait aucun (idéalement) transfert énergétique entre la puce de l'ordinateur et son environnement.

D'un point de vue algorithmique, D-Wave a développé une plateforme logicielle qui contrôle les différents niveaux de leurs machines. Comme solveur nous pouvons citer *qbsolv* et à bas niveau *QMI* qui permet d'intégrer le poids d'un problème sur les qubits [118]. Les ordinateurs D-Wave sont principalement utilisés pour des études de cas d'usage, même si elles restent assez préliminaires et portent le plus souvent sur la vérification de performance. L'étude qui a porté un premier intérêt est celle Google [60] et de la NASA (Figure 3.1) réalisée avec un *D-Wave Two*

1. https://www.dwavesys.com/blog/2015/08/announcing-d-wave-2x-quantum-computer
2. https://www.dwavesys.com/sites/default/files/D-Wave%202000Q%20Tech%20Collateral_0117F_0.pdf
3. Dispositif physique constitué de superconducteurs couplés par une barrière isolante et permettant de démontrer l'effet Josephson.
4. Ce processeur qui n'est pas encore accessible est baptisé ”*Pegasus*”[36] (voir 3.2).

pour résoudre un problème d'optimisation de partitionnement. Le problème qu'ils ont choisi doit répondre à trois critères. 1 : Les solutions répondent au problème posé et le problème doit être pratique. 2 : Le problème est intégrable sur le matériel présent (ou futur). 3 : Le recuit quantique doit offrir un avantage en terme de temps d'exécution par rapport aux autres algorithmes (ici le recuit simulé et le "Quantum Monte Carlo"). Le problème utilisé dans leur article doit déterminer s'il existe une partition d'un ensemble S en deux sous-ensembles tels que la somme des éléments du premier ensemble soit égale à la somme des éléments de l'autre ensemble (problème NP-complet [5]).

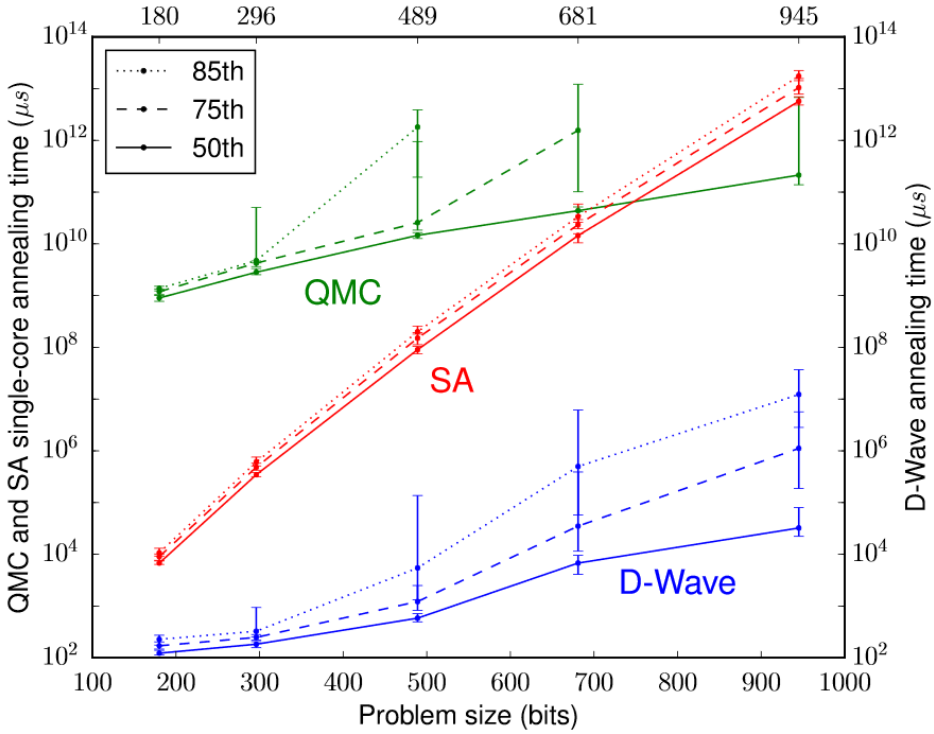

FIGURE 3.1 – Graphe représentant le temps pour trouver la solution optimale avec une probabilité de 99% pour différentes tailles de problèmes. Nous comparons le recuit simulé (SA), le Monte Carlo quantique (QMC) et le DWave 2X. Les 50e, 75e et 85e percentiles sont représentés sur un ensemble de 100 instances. Les barres d'erreur représentent les intervalles de confiance à 95% du bootstrapping. Cette expérience a occupé des millions de cœurs de processeurs pendant plusieurs jours pour régler et exécuter les algorithmes classiques. Les durées d'exécution des quantiles supérieurs pour les problèmes de taille plus importante pour le QMC n'ont pas été calculées parce que le coût de calcul était trop élevé. (Traduction de la figure 4 de l'article [60]).

5. Le problème de partitionnement est l'un des six problèmes fondamentaux de l'ouvrage de Garey et Johnson sur la théorie de la NP-complétude [84]

Malgré des performances estimée 100 millions de fois supérieures à celle d'un recuit simulé et du *QMC* optimisé, d'autres études pointent les limitations des machines et plus particulièrement du recuit quantique. Dans [11], les auteurs affirment que, jusqu'à présent, aucun exemple pratique n'a été trouvé montrant une supériorité d'optimisation avec le recuit quantique [171, 12]. En pratique, trouver des affectations des qubits qui minimisent l'énergie des problèmes d'optimisations combinatoires plus rapidement qu'avec les méthodes classiques est un sujet de recherche et reste une question ouverte. Malgré d'importants efforts pour construire des dispositifs plus denses en terme de connexions entre les qubits, l'idée d'intégrer des problèmes d'optimisation plus importants qui puissent démontrer l'accélération quantique est considéré par le consensus comme difficile à déterminer [106]. En ce sens l'article [11] montre qu'il est essentiel de tester la robustesse du processus d'optimisation face aux erreurs qui réduisent la probabilité d'atteindre l'état fondamental en fin de processus avant d'estimer les performances de ces machines. De plus, dans leur travaux, les auteurs considèrent que la physique sous-jacente du recuit quantique (détaillé à la section suivante) est un processus fixe et fini. En cela, ils soulignent une limitation qui empêche de fonctionner comme un calculateur compétitif avec les algorithmes classiques. Ils en concluent que théoriquement, il faudrait mieux maîtriser le temps de processus en fonction de la taille du problème.
Malgré cette controverse (et d'autres détaillés au Chapitre 4), connaître concrètement les performances de ces machines est devenu une priorité et arriver à résoudre un problème de taille importante avec une bonne qualité de solution est devenu l'enjeu majeur de l'informatique quantique adiabatique.

3.2 Implémentation sur D-Wave

Cette section présente les concepts mathématiques nécessaires et les modèles théoriques avant de pouvoir formuler un problème sur un D-Wave.

3.2.1 Modèle d'Ising

Les systèmes D-Wave sont basés sur un procédé de recuit quantique, une technique d'optimisation qui utilise des phénomènes quantiques afin de parcourir plus efficacement l'espace des solutions [8] et dont le but est de minimiser un hamiltonien d'Ising. Les variables du problème sont ici les spins qui peuvent être soit dans l'état "haut" ($\uparrow$) soit "bas" ($\downarrow$) et correspondent aux valeurs $+1$ et -1. La fonction objectif exprimée sous la forme de l'hamiltonien est la suivante :

$$\mathcal{H}(\mathbf{h}, \mathbf{J}, \boldsymbol{\sigma}) = \sum_i h_i \sigma_i + \sum_{i<j} J_{ij} \sigma_i \sigma_j, \tag{3.1}$$

où le champ externe **h** et la matrice des interactions de couplage de spin **J** sont donnés, et le vecteur des valeurs de spin (ou qubit) $\boldsymbol{\sigma} / \forall i, \sigma_i \in \{-1, 1\}$ est la variable pour laquelle l'énergie du système est minimisée car le processus de recuit adiabatique fait passer le système d'un couplage constant à une superposition de spins [6] à l'hamiltonien final tel que donné par Eq. 3.1. Historiquement parlant, l'hamiltonien d'Ising correspond au cas où seuls les spins voisins les plus proches sont autorisés à interagir (*i.e.* $J_{ij} \neq 0 \iff$ les nœuds i et j sont voisins).

Ces variables de spin doivent prendre la valeur ± 1 pour que les qubits ayant la même direction de courant soient affectés à la même valeur de spin (ou son contraire). Les coefficients linéaires h_i correspondant aux biais des qubits et les coefficients quadratiques $J_{i,j}$ correspondant aux forces de couplage entre deux qubits. Ce biais appliqué prend une valeur scalaire délimitée par le matériel dû à la résolution et la force de couplage entre deux variables de spin détermine la connexion entre les qubits. Les limites de précision du biais et des couplages sont particulièrement importantes pour certains problèmes qui dépendent d'un fragile équilibre des forces entre les variables de spin. D-Wave liste le biais et les couplages sur une plage de valeurs dans $[-1; +1]$ pour J et $[-2; +2]$ pour les h [118]. Lorsque les biais et les couplages sont normalisés dans cette plage, il n'y a plus de distinction entre les valeurs. L'Eq. 3.1 décrit le problème de l'hamiltonien H qui définit les quantités pour une configuration donnée de variables spins. La programmation d'un tel problème implique de régler précisément les valeurs de h et de J pour chaque qubit et pour chaque couplage sur l'appareil. Après réalisation de l'évolution adiabatique, le résultat théorique final devrait être l'état fondamental de l'hamiltonien H. En pratique, cela dépend de l'écart d'énergétique minimum, le temps d'évolution, les effets de décohérence environnementale ou intrinsèque et de la taille du domaine de cohérence des qubits sur la puce. Le spin de chaque qubit σ est mesuré à la fin de ce processus, et cet ensemble de spins de qubits définit l'état d'énergie le plus bas trouvé.

6. L'hamiltonien initial est proportionnel à $\sum_{i,j} \sigma_i^x \sigma_j^x$, donc basé sur les vecteurs propres de l'opérateur $\widehat{\sigma^x}$ (sur l'axe x) alors que le moment de spin sur Eq. 3.1 est un état propre de $\widehat{\sigma^z}$ (sur l'axe z) pour lequel les états propres de $\widehat{\sigma^x}$ sont des états superposés. Le théorème adiabatique permet de passer de l'état ferromagnétique initial sur l'axe x à un état propre de l'hamiltonien de Eq. 3.1 sur l'axe z et, espérons-le, à la plus faible énergie de celui-ci.

3.2.2 Modèle QUBO

Le problème généralisé d'Ising, pour lequel toute paire de spins dans le système est autorisée à interagir, est facilement transformable en un problème d'optimisation binaire bien connu appelé QUBO ("Quadratic Unconstrained Binary Optimization"). Les problèmes QUBO sont des modèles utilisés depuis longtemps en informatique et sont simplement une autre façon de définir les verres de spin d'Ising. Les variables sont des états "VRAI" ou "FAUX" qui correspondent aux valeurs 1 et 0. Un problème QUBO est défini à l'aide d'une matrice triangulaire haute Q de taille $N \times N$ de poids réels et d'un vecteur x de variables binaires pour minimiser la fonction :

$$O(\mathbf{Q}, \mathbf{x}) = \sum_i Q_{ii} x_i + \sum_{i<j} Q_{ij} x_i x_j, \tag{3.2}$$

dans laquelle la matrice $\boldsymbol{Q}$ est constante et le but de l'optimisation est de déterminer le vecteur de variables binaires $\forall i, x_i \in \{0, 1\}$ qui minimise (ou maximise) la fonction objectif $O(\boldsymbol{Q}, \boldsymbol{x})$ de l'Eq. 3.2. Les termes diagonaux $Q_{i,i}$ sont les coefficients linéaires (biais pour la variable de spin x_i) et les termes non nuls hors diagonale sont les coefficients quadratiques $Q_{i,j}$ (force de couplage entre les variables de spin x_i et x_j). Pour le problème de minimisation (un changement de signe pour le problème de maximisation), il est trivial de voir que le problème d'Ising généralisé et le problème QUBO soit équivalent : étant donné $\forall i, Q_{ii} = h_i, \forall i, j, i \neq j, Q_{ij} = J_{ij}$ et $\forall i, \sigma_i = 2x_i - 1$. Ceci peut être exprimé de manière plus compacte comme :

$$\min_{\forall x \in \{0,1\}^n} x^T Q x \tag{3.3}$$

Par conséquent, si le recuit quantique peut atteindre une configuration d'énergie minimale, alors le vecteur d'état associé résout le problème QUBO équivalent en même temps. Les principes de mécanique quantique (par exemple, l'effet tunnel) peuvent aider à atteindre la configuration d'énergie minimale, ou du moins une approximation de celle-ci, dans plus de cas qu'avec un recuit simulé. En effet, lorsque le recuit simulé ne repose que sur des températures (simulées) pour franchir les barrières de potentiel, le recuit quantique peut avoir un meilleur comportement car l'effet tunnel est plus efficace pour franchir les barrières énergétiques (voir Chapitre 2).

3.2.3 Fonction objectif

Le processeur dans un ordinateur quantique adiabatique est conçu pour trouver des solutions de faible coût, et théoriquement la solution optimale au problème de minimisation d'Ising [117]. Ce problème peut être redéfini par un graphe $G = (V, E)$, dont les sommets sont les qubits et les

arêtes les couplages dans le processeur. Soit $h : V \to \mathbb{Z}$ et $J : E \to \mathbb{Z}$ de V et E. Le système tente alors de minimiser l'Eq 3.4.

$$E(s) = \sum_{v \in V} h(v)s_v + \sum_{e \in E} J(e)s_e \tag{3.4}$$

où les variables du problème s_v et les s_e, sont les spins des qubits en $\{-1, +1\}$.

Les reformulations des problèmes combinatoires sous forme de QUBO sont connues pour préserver la structure sous-jacente de la fonction objective [109]. En théorie, tous les problèmes NP-complets peuvent être convertis en un problème QUBO ou Ising [117], et en pratique de nombreux problèmes peuvent être convertis sans trop être déstructurés, tels que le QAP, les variantes de SAT [121], etc. Les formulations mathématiques du théorème adiabatique [8] et du problème de référence fournissent alors la base pour résoudre de tels problèmes sur un ordinateur quantique analogique.

3.3 Principes d'ordinateur quantique adiabatique

Le modèle "*Adiabatic Quantum Computing*" (AQC) est une architecture de calcul quantique ayant des similitudes avec les algorithmes d'optimisation du type recuit simulé. Cette section décrit la théorie qui sous-jacente de l'AQC et les bases de la programmation d'un ordinateur quantique D-Wave [97] avec les concepts de physique essentiels à la compréhension du processus qui régit le recuit quantique.

L'hamiltonien et le spectre énergétique Un hamiltonien classique est par définition une description mathématique d'un système physique en matière d'énergie (voir rappels mathématiques au Chapitre 2). Il est alors facile de définir n'importe quel état particulier du système, et l'hamiltonien nous donne l'énergie pour cet état. Pour la plupart des hamiltoniens, trouver l'état d'énergie minimal est un problème NP-difficile, et les ordinateurs classiques ne peuvent pas (encore) déterminer efficacement celui-ci. Dans le cadre de la mécanique quantique, nous pouvons définir un opérateur hamiltonien $\hat{H}$ associé à l'énergie du système physique modélisé. Les états propres de cet opérateur hamiltonien forment un ensemble d'états qui correspondent aux modes stables et observables de l'énergie possible du système. Les valeurs propres associées forment ce qui est appelé le spectre énergétique du système.
L'hamiltonien d'un système quantique est décrit comme une matrice H de dimension $n \times n$, alors son spectre représente l'ensemble de toutes ses valeurs propres pour tous les états possibles Ψ du système :

$$H\Psi = \lambda\Psi \tag{3.5}$$

L'état fondamental est l'état avec le minimum d'énergie (le minimum global) et il est habituellement définit comme $\lambda_0 = 0$. Il est important de noter que contrairement à la mécanique classique le niveau fondamental ne correspond pas tout à fait avec le minimum du hamiltonien à cause du principe d'incertitude d'Heisenberg. Le premier état excité d'un hamiltonien est le premier état au-dessus du niveau fondamental, avec sa valeur énergétique correspondante λ_1 [7]. Quand le système est dans un état propre de l'hamiltonien, son énergie est alors bien définie et est appelée énergie propre. Lorsque le système est dans un autre état (transitoire), son énergie est indéterminée. La physique des basses énergies des ordinateurs D-Wave [9] est donnée par un hamiltonien de la forme :

$$\mathcal{H}(t) = A(t)\mathcal{H}_0 + B(t)\mathcal{H}_P \tag{3.6}$$

où les fonctions $A(t)$ et $B(t)$ doivent satisfaire $B(t = 0) = 0$ et $A(t = \tau) = 0$ de sorte que, lorsque l'évolution de l'état $t = 0$ passe à $t = \tau$, le hamiltonien $H(t)$ est "recuit" sous une forme purement classique. Ainsi, l'état fondamental $\mathcal{H}(0) = \mathcal{H}_0$ évolue vers un état $\mathcal{H}(\tau) = \mathcal{H}_P$, les mesures effectuées à l'instant τ nous donnent les états de faible énergie de l'hamiltonien d'Ising (Eq. 3.1). Le théorème adiabatique stipule que si l'évolution temporelle est suffisamment lente (*i.e.* τ est assez grand), alors la solution optimale (globale) $\epsilon(\sigma)$ du système peut être obtenue avec une forte probabilité. $\mathcal{H}_0 = \sum_i \sigma_i^x$ donne les effets quantiques, et $\mathcal{H}_P = \sum_i h_i\sigma_i^z + \sum_{(ij)} J_{i,j}\sigma_i^z\sigma_j^z$ est donné pour encoder le problème de l'instance Ising :

$$\min \epsilon(\sigma) = \min \left\{ \sum_i h_i\sigma_i + \sum_{i,j} J_{i,j}\sigma_i\sigma_j \right\} \tag{3.7}$$

L'Hamiltonien 3.6 est une somme de deux termes, un hamiltonien initial et un Hamiltonien final :

— H_0 (premier terme) ; L'état d'énergie le plus bas de H_0 est atteint quand tous les qubits sont superposés sur σ_z, il est purement quantique [8] et est appelé un hamiltonien intriqué et s'écrit sous la forme : $-1/2\sum_i \sigma_x^i$.

— H_P (deuxième terme) ; L'état d'énergie le plus bas est la solution au problème initialement posé. L'état final est un état purement classique, et inclut les biais des qubits et les couplages entre eux. Ce terme dans l'hamiltonien global est l'hamiltonien d'Ising de la

7. Même si l'ensemble énergétique peut être continu, un hamiltonien a un ensemble d'états discrets. Aucun état n'est possible entre l'état fondamental et le 1^{er} était excité (et ainsi entre chaque état).

8. Car le système est quantique, autrement, il s'agit d'un simple état propre de σ_x, qui s'exprime comme une superposition d'état de σ_z car les opérateurs de Pauli ne sont pas observables simultanément (non commutables).

forme : $\sum_i h_i \sigma_z^i + \sum_{i<j} J_{ij} \sigma_z^i \sigma_z^j$. Avec $\sigma_z^{(i,j)}$ les matrices de Pauli qui sont définies sur le qubit q_i. h_i et $J_{i,j}$ sont les poids (les biais et forces de couplages) des qubits. Par définition ces poids n'ont pas de bornes, ils sont définis en amont par le problème à mettre sur la machine. Mais nous verrons par la suite que l'interface de D-Wave impose une limitation sur ces poids due au champ qui est appliqué pour contrôler les qubits.

Dans le recuit quantique, nous commençons dans l'état propre le plus bas de l'hamiltonien initial, tous les qubits utilisés sont intriqués. Le recuit va introduire progressivement l'hamiltonien d'Ising, qui contient les biais et les coupleurs, et abaisser l'influence (la superposition) de l'hamiltonien initial. À la fin du recuit, le système est alors dans un état propre du problème de l'hamiltonien d'Ising et idéalement le système reste dans l'état d'énergie le plus faible tout au long du processus. En sortie, l'état d'énergie est minimal et le système donne une (bonne) réponse au problème posé.

Etats de faible énergie Le système commence dans l'état le plus bas énergétiquement et au fur et à mesure que l'hamiltonien est introduit, des niveaux d'énergie excités peuvent se rapprocher de l'état fondamental. Plus l'écart entre ces deux niveaux est faible, plus la probabilité que le système passe de l'état d'énergie fondamental à l'un des états excités devient élevée. Il y a un moment dans le spectre où le premier état excité, avec la plus faible énergie (hormis l'état fondamental), s'entrecroise étroitement avec l'état fondamental, puis s'écarte à nouveau. Certains facteurs peuvent faire passer le système de l'état fondamental à un état d'énergie supérieur. Il existe donc une distance minimale entre les deux états. Les principales sources sont alors, la fluctuation thermique qui existe dans tous les systèmes physiques et le processus de recuit exécuté bien trop rapidement. Le processus de recuit qui ne subit aucune interférence des sources d'énergie extérieures et qui évolue assez lentement pour laisser le système se stabiliser est appelé le processus adiabatique [9]. Étant donné qu'aucun ordinateur, quantique ou non, n'est jamais totalement isolé de l'extérieur, le recuit quantique peut être vu comme l'équivalent pratique de l'informatique quantique adiabatique idéal (théorique). En réalité, pour certains problèmes, la probabilité de rester dans l'état fondamental peut parfois être assez faible. Ainsi, pour chaque problème, il existe un hamiltonien avec un spectre propre lui correspondant et les problèmes les plus difficiles à résoudre (en matière de recuit quantique), sont généralement ceux qui présentent les écarts énergétiques les plus faibles.

9. C'est de là que vient le nom d'informatique quantique adiabatique.

3.3.1 Théorème adiabatique

Le système évolue dans le temps dt selon l'équation de Schrödinger [73] (voir rappels mathématiques au Chapitre 2) et le théorème adiabatique stipule : " *pour un système donné, si ce système évolue assez lentement, alors il restera toujours dans le même état d'énergie propre*". Pour exploiter ce théorème, il faut commencer par un système quantique donné par un hamiltonien initial H_0, avec un état fondamental facile à déterminer. Si le système tend vers un hamiltonien H_f (description d'un ensemble d'états énergétiques à partir duquel le minimum global est souhaité) en évoluant assez lentement vers un état d'énergie, à la fin de l'évolution, le système restera encore dans l'état fondamental, et donc le minimum global de H_f est finalement déterminé.
Un système quantique a une énergie donnée par l'hamiltonien H_0 à un temps initial t_0 et le système est dans un état propre décrit par $\Psi(x, t_0)$. Le système évolue selon l'équation de Schrödinger, afin d'atteindre l'état final $\Psi(x, t_f)$ ce qui donne un hamiltonien final $H_f(t_f)$ à un temps t_f. Le théorème adiabatique indique que la modification du système dépend du temps par la relation $t_a = t_f - t_0$. Pour un processus adiabatique, $t_a \to \infty$, alors, l'état final $\Psi(x, t_f)$ est un état propre de l'hamiltonien final, avec une configuration $|\Psi(x, t_f)|^2 \neq |\Psi(x, t_0)|^2$.
Le degré auquel un changement donné approche un processus adiabatique dépend à la fois de la séparation énergétique g_{min} entre les états énergétiques, et le rapport entre l'intervalle t_a de l'évolution de $\Psi(x, t_0)$ pour un hamiltonien indépendant du temps. Inversement, dans la limite $t_a \to 0$ on a un passage infiniment rapide (ou diabatique) ; la configuration de l'état reste inchangée : $|\Psi(x, t_f)|^2 = |\Psi(x, t_0)|^2$. Ce théorème s'applique seulement lorsque l'évolution se fait assez lentement et le temps total requis pour une évolution adiabatique (un seul recuit), t_a, est donné par le théorème adiabatique :

$$t_a \gg \frac{1}{g_{min}^2} \tag{3.8}$$

où g_{min} est l'écart d'énergie minimale [10] tel que défini par ;

$$g_{min} = \min_{0 \leq s \leq 1} (\lambda_1(s) - \lambda_0(s)) \tag{3.9}$$

Avec $s = t/t_f$. Cette relation implique que, plus l'écart énergétique minimal est faible, plus l'évolution doit durer longtemps pour qu'elle reste dans l'état fondamental. Malheureusement, comme pour beaucoup de processus quantiques, le calcul de l'écart énergétique minimal est généralement un processus difficile et équivaut en réalité à résoudre le problème lui-même.

10. l'écart le plus petit entre l'état fondamental et le 1^{er} état excité de l'hamiltonien à un moment donné

3.3.2 Fonctionnement interne

Les qubits matérialisés par des boucles microscopiques [129] composent le QPU ("Quantum Processor Unit") d'un ordinateur D-Wave. Pendant le calcul, la direction du courant électrique est dans un état superposé, mais lorsque le système décohére (en raison de l'observation ou du bruit), le courant circule dans un sens ou dans l'autre. Pour effectuer des calculs efficaces, il est nécessaire de soigneusement contrôler la distribution de probabilité de la rotation du qubit. Ceci est réalisé en programmant le *biais* et les *couplages* de chaque qubit. Le biais est un champ magnétique programmable appliqué à la boucle du qubit qui va effectuer la rotation du courant dans la direction souhaitée à la valeur choisie. Chaque qubit est également connecté à d'autres qubits à l'aide de couplages ferromagnétiques/antiferromagnétiques programmables, qui incitent les qubits à se polariser préférentiellement dans le même sens ou respectivement dans le sens inverse des qubits auxquels ils sont connectés.

Le processus physique qui rentre en jeu peut être représenté évolue dans le temps suivant trois étapes : à l'initialisation le qubit possède un unique puits (représentant l'état 0 ou 1) avec un seul minimum énergétique (un état superposé). Au début du processus de recuit, le puits se délocalise en deux états bien distincts (correspondant à l'état $|0\rangle$ et l'état $|1\rangle$ du qubit). Ce processus transforme le diagramme énergétique en un potentiel à double puits. Le qubit doit se retrouver dans l'une de ces vallées à la fin du recuit pour sortir de son état quantique et de tomber dans un état classique.
La probabilité que le qubit se retrouve à l'état $|0\rangle$ où $|1\rangle$ est alors égale à 50%. Il est cependant possible de contrôler cette probabilité en appliquant un champ magnétique externe au qubit. L'application de ce champ fait basculer le potentiel du double puits dans un unique puits, augmentant fortement la probabilité que le qubit se retrouve dans le puits le plus bas énergétiquement. La constante programmable pour contrôler le champ magnétique externe est la polarisation. La polarisation introduit un minimum d'énergie préférentiel pour une des deux orientations qui permet par exemple de contrôler la probabilité d'une orientation plutôt que l'autre et faire passer celle-ci de façon continue de 0% à 100%.

L'architecture de ce dispositif est alors conçu pour associer plusieurs qubits en les couplant mutuellement afin qu'ils puissent s'influencer. Contrairement aux ordinateurs quantiques numériques qui utilisent une porte quantique à deux qubits pour coupler les qubits entre eux, l'AQC utilise directement des biais et des coupleurs. L'architecture de D-Wave associe à chaque qubit et à chaque couplage un poids que nous programmons en fonction du type de problème posé en amont.

3.4 Architecture

La topologie (graphe d'interconnexion des qubits physiques) de D-Wave est assez contraignante pour traduire une fonction objectif QUBO (ou Ising) dans un format que le système D-Wave peut résoudre. Lors de la formulation d'un problème sur le D-Wave, nous associons les valeurs des biais (poids sur les qubits) et des coupleurs (poids de connexions entre deux qubits) sur la topologie d'interconnexion des qubits et théoriquement, plus il y a de qubits et de connexions utilisés, plus les problèmes qui y sont associés deviennent denses et complexes à résoudre. Comme nous le verrons dans la suite de cette section, la topologie mise en œuvre dans les architectures D-Wave est quadratiquement moins dense que la densité maximale que nous pouvons avoir sur un problème.

3.4.1 Connectivité des qubits dans le graphe Chimera

Cellule unitaire Une cellule unitaire est composée de 2×4 qubits sur un graphe biparti (encadrée en rouge sur la figure 3.2). Chaque qubit est relié au plus à quatre autres qubits pour un total de connexions au plus de 6.

Graphe 4×4 Les interactions qubit à qubit, sont limitées par son graphe d'interconnexion C_q contenant $8q^2$ qubits. Le réseau d'interconnexion est délimité par les cellules unitaires de qubits connectés entre elles pour former une topologie globale en grille, le Chimera (la topologie Chimera pour 128 qubits est illustrée sur la figure 3.2).
Les nœuds et les arrêtes du graphe permettent de définir la fonction objectif. Ainsi chaque qubit physique du graphe est associé à un "qubit logique" (variable diagonale du problème QUBO) dans le graphe de la fonction objectif. Par exemple, Choi et al [51] montrent qu'un graphe complet K_q peut être intégré dans le triangle supérieur du graphe Chimera C_q au prix de l'utilisation de plusieurs qubits physiques comme qubits logiques. Ce principe implique une réduction drastique du nombre de qubits logiques disponibles dans le système. Cela crée un écart (important) entre le nombre de qubits physiques et la taille (nombre de variables) des problèmes. De plus, la recherche de bons algorithmes et/ou d'heuristiques pour trouver de telles correspondances reste un problème ouvert. Malheureusement et comme nous le verrons par la suite, une variable logique du problème doit souvent être associée à plusieurs qubits physiques.

La propriété la plus importante du graphe de Chimera est qu'il s'agit d'un graphe peu dense ; chaque qubit est couplé avec au plus six autres qubits. Cela signifie que tous les problèmes intégrables sur un graphe de variables qui ne correspondent pas exactement à la structure de

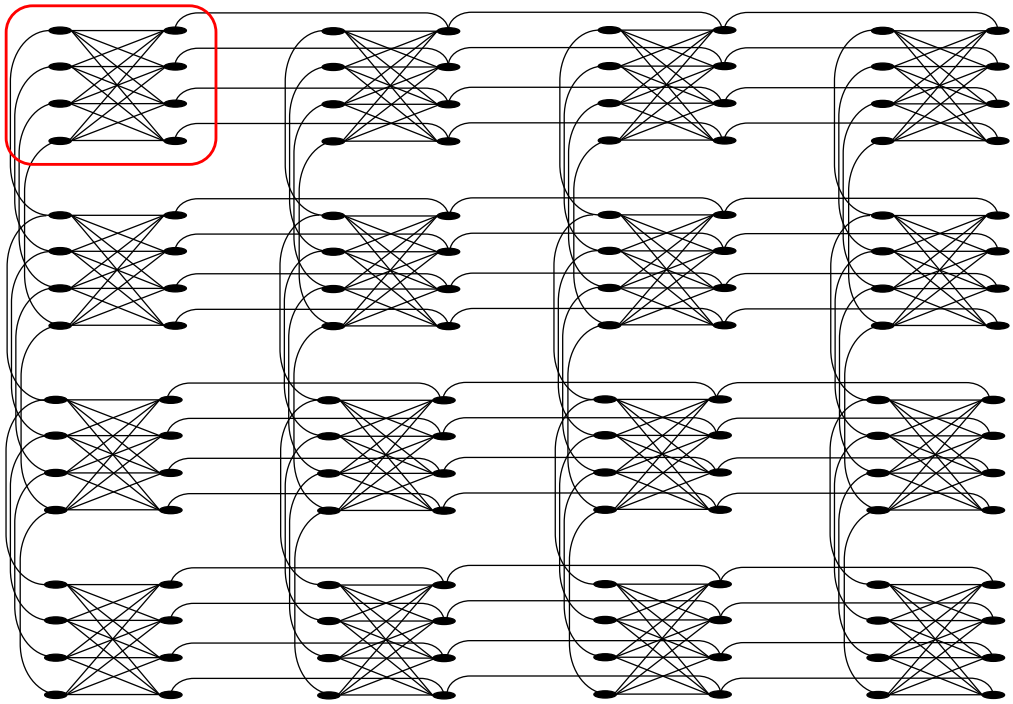

FIGURE 3.2 – Représentation du graphe Chimera avec 4×4 cellules unitaires (encadrée en rouge), chacune d'elles définit un graphe biparti pour un total de 128 qubits physiques

Chimera devront être reformulés. L'incorporation des problèmes se réfère à une transformation de graphe pour se conformer à la structure du graphe Chimera. Deux problèmes entrent en jeu. Le premier est le cas des graphes isomorphes au graphe Chimera. Trouver un isomorphisme est un problème difficile [82] et nous ne savons pas s'il est dans la classe NP. Des heuristiques de prétraitements doivent être développés en amont pour assurer l'isomorphisme du problème sur le graphe Chimera. Deuxièmement, si le graphe n'est pas isomorphe, alors la solution la plus adaptée pour venir intégrer le graphe sur le Chimera est de dupliquer plusieurs qubits pour représenter une variable. En théorie des graphes, *"si le graphe du problème QUBO n'est pas un sous-graphe (isomorphe) au graphe Chimera, alors il ne peut pas être directement intégré"*. Pour une résolution exacte, Il faut alors recourir au problème d'incorporation de graphe et utiliser plusieurs qubits physiques comme un unique qubit logique.
Pour pallier ce problème d'isomorphisme de graphe, une des façons de faire est de mettre le même poids sur plusieurs qubits physiques de façon à ce que la chaîne totale représente une seule et unique variable. Avec cette méthode, la taille maximale du graphe qui peut être intégrée (en supposant une intégration parfaite) est souvent très inférieure à la taille réelle du problème

(voir Chapitre 7). Une grande partie de la théorie sous-jacente se trouve dans Choi [51] qui est une heuristique (**minor-embedding algorithm** [11]) pour faire l'intégration des graphes fournis dans le cadre de l'interface de programmation D-Wave [44].

3.4.2 Connectivité des qubits dans le graphe Pegasus

La prochaine génération de D-Wave pourra atteindre plus de 5000 qubits interconnectés dans une topologie de graphe appelé Pegasus [58, 36]. Cette topologie englobera l'ancienne topologie Chimera comme sous-graphe. La topologie Pegasus possède, en plus des coupleurs externes (qui connectent des paires de qubits sur deux bipartis différents) et internes (qui connectent des paires de qubits dans un biparti) de Chimera, un troisième type de coupleur, les coupleurs impairs. Les coupleurs impairs connectent des paires de qubits parallèles dans des colonnes adjacentes (schématisé à la figure 3.3). Dans la topologie de Chimera, un qubit peut connecter au plus 4 qubits internes et 2 qubits externes pour avoir au plus 6 connexions entre chaque qubit. Dans un Pegasus, un qubit peut connecter 12 qubits internes et 3 externes pour un total de 15 connexions entre chaque qubit.

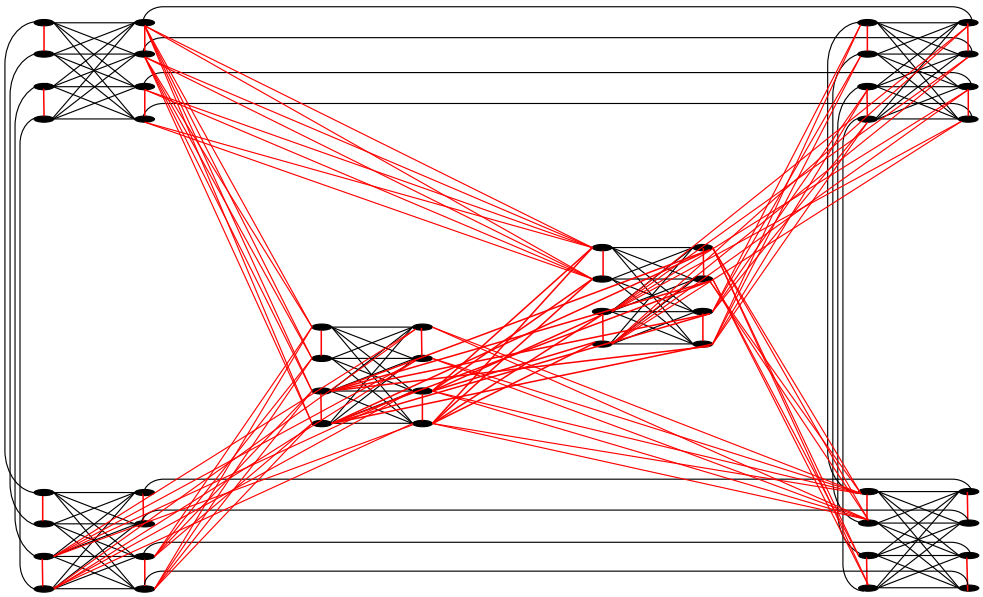

FIGURE 3.3 – Représentation du graphe Pegasus superposé sur le Chimera de 2×2 cellules unitaires. Les arrêtes en rouge représentent les connexions supplémentaires incorporées dans une topologie Pegasus pour un total de 48 qubits.

11. L'heuristique détermine le plus grand sous-graphe en procédant par construction itérative partant d'un sommet jusqu'à couvrir le maximum de sommet disponible dans le graphe.

Ainsi, un Pegasus de taille M noté P_M contient au total $24M(M-1)$ qubits. Comparativement à un Chimera C_4 de taille 4×4 avec 128 qubits, un Pegasus P_4 possède 288 qubits (augmentation de 2 par rapport au nombre de qubits du Chimera). De plus, les heuristiques utilisées pour déterminer les sous-graphes isomorphes au graphe Chimera pourront être facilement réadaptées car la nouvelle architecture englobe le Chimera [172]. L'avantage prédominant dans cette nouvelle topologie est que les chaînes de qubits (duplication des qubits pour représenter une variable d'un problème) seront nettement plus courtes grâce à la densité de connexion 2.5 fois supérieure. Enfin, comme des chaînes de qubits augmentent le risque d'erreur (mauvaises duplications en sortie) dans l'intégration d'un problème sur le graphe, la densité devrait réduire la longueur des chaînes et permettre d'améliorer la résolution des problèmes.
Dans [58], D-Wave montre des premiers résultats de leurs tests sur leur Pegasus et en particulier deux points importants ; le premier pour trouver des sous-graphes dans un $P6$ (avec 680 qubits et 4484 coupleurs). Il détermine que l'architecture Pegasus obtient systématiquement une réduction de 50 à 60% de la longueur des chaînes par rapport à Chimera, donnant l'avantage de venir intégrer des problèmes plus denses. En second point, il confirme que la topologie peut supporter des barrières énergétiques plus importantes et tendrait à confirmer l'utilisation des principes de la physique quantique controversés. Cette nouvelle technologie offrirait plus de facilité à observer les phénomènes (effet tunnel notamment) qu'avec un Chimera et mettrait fin au débat (voir Chapitre 4) sur l'utilisation ou non de ces phénomènes. Enfin, l'utilisation de leur nouvelle puce avec ces nouvelles propriétés permettrait d'accélérer les temps de calcul et la qualité des solutions.
Bien que les machines basées sur la technologie Pegasus ne soient pas encore commercialisées, la chaîne d'outils logiciels supporte déjà cette nouvelle topologie d'interconnexion et permet de réaliser des expériences préliminaires (section 7.2.1).

3.4.3 Limites de la connectivité

La disposition du graphe Chimera du calculateur quantique entraîne certaines limitations sur la taille des problèmes qui peuvent être intégrer directement. Les deux plus grandes limitations sont le nombre de qubits et leurs connexions.

Nombre de qubits Le nombre de qubits dans les ordinateurs quantiques D-Wave a été jusqu'à présent relativement faible. Pour un graphe Chimera de taille $M \times N$, avec M le nombre de colonnes et N le nombre de lignes, il y a un total de $2 \times L \times M \times N$ qubits avec L le nombre de cellules unitaires. Actuellement disponible, l'ordinateur avec le plus grand nombre de qubits est le D-Wave 2X avec 2048 qubits et seulement 1152 connexions entre les qubits. Cela signifie que

seuls les problèmes QUBO intégrables de taille 2048 avec seulement 1152 variables non nulles et non diagonaux représentant les poids de couplage sur le Chimera (termes de couplages $Q_{i,j}$) peuvent être disposés sur le D-Wave. En utilisant les 2048 termes diagonaux comme biais sur les qubits, le nombre total des termes non diagonaux non nuls directement intégrable sur le Chimera ne représente que 0.05% du nombre total de termes non nuls dans la matrice complète. Pour contourner cette limitation, l'article de D-Wave [118] suggère qu'il n'y a pas d'autre choix que de diviser le problème en sous problèmes plus petits, de les résoudre et de reconstruire la solution du problème initial. Nous verrons dans le chapitre 5 qu'il existe une autre façon de procéder mais elle aussi possède ses limitations.

Connectivité entre qubits Chaque nœud du graphe Chimera est adjacent à au plus six autres nœuds. Cette contrainte signifie que, si un problème a des variables qui sont connectées les unes aux autres d'une manière qui n'existe pas dans ce graphe, alors il ne peut pas être intégré sur le matériel sans être reformulé. En théorie des graphes : *si le problème a un graphe qui n'est pas un sous-graphe du graphe de Chimera alors nous avons besoin de reformuler le problème afin de résoudre le problème initial*. Pour contourner cette limitation, nous utilisons plusieurs qubits physiques pour représenter un unique qubit dans notre problème. Par définition, *un qubit physique* est un qubit qui compose le matériel quantique, et *un qubit logique* est la représentation d'une variable du problème. Pour un graphe de Chimera avec les paramètres L, M, N, la documentation de D-Wave [118] indique que le graphe complet le plus grand avec n sommets et pour $n = 1 + L.min(M, N)$. Ainsi, la faible connectivité implique un nombre de variables (nombre de qubits) et de connectivités (nombre de couplages) limités pour intégrer le plus grand graphe possible du problème. Intégrer un graphe directement sur l'architecture revient souvent à redéfinir un graphe directement isomorphe.

Chapitre 4

Présentation du problème et état de l'art

4.1 Recuit Simulé vs Recuit Quantique

L'heuristique du recuit simulé se base à l'origine sur la simulation du recuit des matériaux [107]. En physique de la matière condensée, appliquer un recuit sur solide est un processus physique permettant de chauffer le solide jusqu'à obtenir une température maximale pour laquelle toutes les particules se déplacent aléatoirement jusqu'à être dans la phase liquide. Le liquide est ensuite refroidi en abaissant lentement la température. Ainsi, les particules retombent dans l'état fondamental (état de plus faible énergie) correspondant au solide. La condition nécessaire et suffisante est que la température maximale doit être initialement élevée et que le refroidissement s'effectue suffisamment lentement. La phase de refroidissement est alors donnée pour chaque valeur de température, et le solide doit atteindre l'équilibre thermique caractérisé par une probabilité d'être dans un état énergétique donné par la distribution de Boltzmann :

$$P = \frac{1}{Z(T)} \exp\left\{\frac{-E}{k_b T}\right\} \tag{4.1}$$

Où $Z(T)$ est un facteur de normalisation de la fonction de partition dépendant de la température T et k_B la constante de Boltzmann, E l'état énergétique du système. Lorsque la température est abaissée, la distribution de Boltzmann fait tendre les états vers l'énergie la plus basse possible ($\lim_{t\to\infty} \exp^{-t} = 0$) et lorsque la température avoisine le zéro, seuls les états ayant la plus faible énergie ont une probabilité d'exister. En revanche, si le refroidissement est effectué trop rapidement et que l'équilibre thermique n'a pas le temps d'être atteint pour chaque valeur de température, alors des défauts peuvent apparaître dans le solide et la structure cristalline d'énergie la plus faible n'est pas atteinte.

Du point de vue informatique, pour simuler l'évolution vers l'équilibre thermique d'un solide à valeur constante T, Metropolis et al. ont proposé une méthode de Monte Carlo [123], qui

génère des séquences d'états du solide. Étant donné l'état à un moment donné du solide, caractérisé par les positions de ses particules, une légère perturbation générée de façon aléatoire est appliquée par le déplacement d'une particule choisie aléatoirement. Après stabilisation, si la différence d'énergie, ΔE, entre l'état actuel et l'état légèrement perturbé est négative (énergie plus faible), le processus se poursuit avec ce nouvel état. Si $\Delta E \geq 0$, alors la probabilité d'accepter l'état perturbé est donnée par $\exp\left\{\frac{-\Delta E}{k_b T}\right\}$. Cette règle d'acceptation pour les nouveaux états d'énergie est le critère Metropolis. Selon ce critère, le système va évoluer vers un équilibre thermique [1] et atteindra l'état de plus bas énergie possible du système.

L'algorithme d'optimisation a lui été développé indépendamment par Kirkpatrick et al. [46, 107] dans les années 80. L'algorithme de Metropolis est également utilisé pour générer des séquences de configurations d'un problème d'optimisation combinatoire. Dans ce cas, les configurations vont jouer le rôle des états du solide tandis que la fonction de coût et le paramètre de contrôle jouent les rôles de l'énergie et de la température. L'algorithme de recuit simulé est alors considéré comme une séquence d'algorithmes de Metropolis évaluée par les valeurs décroissantes du paramètre de contrôle. En utilisant cette méthode, il est possible de sortir d'un minimum local avec une probabilité élevée dépendant de la température. Ainsi à très basse température, les états les plus probables d'être atteints sont les états d'énergie très faible et sont les solutions au problème d'optimisation.
Le principe de l'heuristique du recuit simulé consiste à initialiser l'algorithme avec une température élevée, puis, de maîtriser la loi de décroissance de la température dans l'algorithme de Metropolis. De cette façon, il existe plusieurs lois de décroissance ([69, 70], etc.) qui possèdent toutes des caractéristiques différentes et qui offrent de plus ou moins bonnes solutions au problème considéré. De la même façon qu'il existe plusieurs lois de décroissances, il existe un grand nombre de ce type d'algorithmes a cause de l'ajustement des paramètres sur lesquels il possible d'influer (valeur initiale de la température, fonction de décroissance de la température, nombre de paliers par niveau, critère d'arrêt de l'algorithme, etc.). Alors, l'efficacité et la performance de l'heuristique sont intrinsèquement liées aux choix de ces paramètres de contrôle.

4.1.1 Performance

L'analyse des performances d'un algorithme non déterministe porte sur la qualité des solutions obtenues et le temps d'exécution requis par l'algorithme. Pour le recuit simulé, ces quantités dépendent de la taille de l'instance du problème et du nombre d'itérations par pallier. Il

1. Après un grand nombre de perturbations, en utilisant le critère d'acceptation, la distribution de probabilité des états se rapproche de la distribution de Boltzmann, donnée par eq. 4.1

existe également une distribution de probabilité sur l'ensemble des solutions possibles que l'algorithme peut obtenir pour un problème donné. Ainsi, la moyenne[2] fait référence à l'espérance mathématique du coût sur l'ensemble des solutions de l'instance du problème. En étudiant les performances du recuit simulé, il est possible de quantifier la qualité des solutions obtenues par rapport à la solution optimale.

Sasaki et al. [150] ont réussi à déterminer que dans certains cas, le recuit simulé ne résout pas certains problèmes polynomiaux en temps polynomial, par exemple, le problème de couplage biparti de cardinalité maximale présenté dans le Chapitre 6. Plus précisément, certains cas particuliers nécessitent un nombre exponentiel d'itérations pour atteindre la solution optimale. D'autres études [5, 94] ont approfondi la question de la convergence du recuit vers l'optimum global et ont montré théoriquement qu'il existe une condition nécessaire et suffisante pour que le recuit converge en probabilité vers l'optimum global, et ce en faisant un nombre exponentiel d'itérations. Cette condition est alors valable pour une grande classe de types de recuit indépendamment de la loi de décroissance choisie. Il nous est alors impossible de garantir exactement la convergence du recuit, même pour un problème polynomial.
En dépit des inconvénients que nous venons de voir, le recuit simulé reste une heuristique largement utilisée en optimisation car elle permet très souvent d'obtenir des solutions sous-optimales mais utilisables en un temps de calcul qui reste raisonnable.

4.1.2 Comparaison Recuit Simulé vs Recuit Quantique

L'optimisation combinatoire est un domaine de l'informatique ayant pour fonction de déterminer le minimum d'une fonction de coût possédant un grand nombre de variables discrètes. Comme nous l'avons vu, la théorie de la complexité classe la difficulté d'une telle tâche selon le type de problème en fonction du temps de calcul et de la mémoire utilisée par les algorithmes. Lorsque la fonction de coût peut être vue comme une énergie, les configurations optimales correspondent aux états fondamentaux et amènent à des procédures de recuit inspirées de la physique afin de déterminer l'état fondamental du système.

Le recuit simulé Cette heuristique exploite en particulier les fluctuations thermiques pour atteindre l'état fondamental, la température peut être lentement abaissée jusqu'à atteindre une température proche de zéro. Si cette baisse est suffisamment lente et contrôlée, le système reste

2. Pour une distribution de probabilité du coût sur l'ensemble des solutions possibles que l'algorithme peut obtenir pour un problème donné.

proche de l'équilibre thermique et en fin de processus, les configurations sont les états fondamentaux du système.

Depuis la fin des années 90, les algorithmes de type recuit ont été utilisés avec succès pour résoudre efficacement des problèmes [47]. L'intérêt d'utiliser cette heuristique plutôt que des solveurs exacts réside dans la simplicité de l'algorithme, un comportement type boîte noire [3] et une grande variété de paramètres internes réglables. Théoriquement, l'heuristique doit approcher la solution optimale du problème par le biais d'une marche aléatoire (processus de Markov avec une bonne loi stationnaire) au propriétés bien choisies (règle probabiliste qui permet d'accepter ou non une solution qui est liée à la loi de Boltzman) dans tous l'espace des solutions. D'un point de vue pratique, le réglage des paramètres internes permet d'accélérer la convergence vers une solution proche de l'optimale. Le recuit simulé est aussi adapté pour résoudre les problèmes d'optimisation continue [4] [159]. A contrario, l'inconvénient majeur est le temps de résolution même pour des instances relativement petites. En effet, il faut tout de même explorer un grand nombre de solutions dans tout l'espace pour ne pas tomber et rester bloqué dans un minima local.

Le recuit quantique Les fluctuations thermiques sont définies comme des fluctuations quantiques [103] afin de rapidement explorer l'espace de configuration vers la configuration de base. Ce recuit peut être vu comme une métaheuristique pour trouver le minimum global d'une fonction objective sur un ensemble de solutions candidates [81]. L'idée est alors de faire évoluer le système décrit par l'équation de Schrödinger, avec un hamiltonien interpolant un hamiltonien plus simple dont l'état de base est facile à préparer et l'hamiltonien encodant la fonction de coût que nous souhaitons minimiser. Si cette interpolation est suffisamment lente et donc si la condition d'adiabaticité est bien maîtrisée, le système initialement préparé dans l'état fondamental de l'hamiltonien simple finira dans l'état fondamental recherché de la fonction de coût. Cette procédure mise en œuvre dans un ordinateur quantique reste très coûteuse sur un ordinateur classique. En effet la dimension de l'espace d'Hilbert quantique augmente de façon exponentielle avec le nombre de variables.

3. Représentation du système sans utiliser son fonctionnement interne et sans avoir besoin de calculer la fonction économique.

4. Dans les problèmes d'optimisation continue, les variables peuvent prendre l'ensemble des valeurs réels.

4.2 Analyse bibliographique

L'état de l'art qui est présenté dans ce chapitre fait le point sur l'état actuel des techniques en matière de calcul quantique adiabatique et sur ses applications.

4.2.1 Calcul Quantique Adiabatique (AQC)

Le modèle adiabatique a été décrit pour la première fois par Farhi et al. [8] dans les années 2000. Ils ont l'intuition d'appliquer l'évolution adiabatique d'un système quantique comme algorithme pour résoudre certains problèmes d'optimisation combinatoire. Par définition, pour qu'un tel algorithme soit efficace, son temps d'exécution pour résoudre un problème doit augmenter au pire polynomialement avec le nombre de qubits. Ce laps de temps d'exécution dépend de l'écart spectral minimal (g_{min}) entre les états d'énergies des hamiltoniens.
Des travaux plus approfondis des mêmes auteurs en 2001 [75] ont montré que les simulations du calcul quantique adiabatique étaient capables de résoudre des problèmes NP-complets générés aléatoirement. Ils ont aussi montré une faible augmentation du temps de fonctionnement du recuit en fonction du nombre de qubits, mais seulement pour un petit nombre de qubits (en raison des contraintes d'espace des systèmes quantiques de simulation classique). De plus, ils ont montré que l'évolution temporelle des hamiltoniens pouvait être approchée à l'aide d'opérateurs unitaires discrets dans le modèle informatique quantique universel. Récemment, l'algorithme adiabatique a été appliqué à un problème spécifique NP-complet comme : "3-bit Exact Cover" [75] et 3-SAT ("Three-Satisfiability") [108]. Il a été constaté que l'algorithme quantique réussit en un temps qui semble croître seulement quadratiquement en fonction de l'entrée.

Comme l'AQC peut être simulé par des ordinateurs quantiques universels, sa puissance de calcul ne pourra pas dépasser celle du modèle conventionnel. C'est Aharonov et al. [7] qui ont d'abord prouvé que la réciproque est également vraie ; le modèle de calcul quantique universel pouvait être simulé efficacement sur le recuit adiabatique (avec un surcroît de complexité polynomiale). Une autre question qui fait encore débat est l'étude du passage d'une éventuelle différence de complexité d'un problème avec des ordinateurs quantiques à portes sur des ordinateurs à recuit quantique de D-Wave. D'après *Adiabatic Quantum Computation is Equivalent to Standard Quantum Computation* [7] de Dorit Aharonov et al. il n'y aurait aucune différence (à condition de considérer des hamiltoniens plus généralisés), la complexité est identique.
Enfin, Childs et al. [50] ont prouvé que le modèle adiabatique présentait des tolérances de défauts inhérents qui n'étaient pas présentes dans le calcul quantique universel. Intuitivement, comme les performances dépendent de l'écart spectral g_{min}, la décohérence environnementale pourrait

être minimisée en utilisant un dispositif à très basse température par rapport au g_{min}. De plus, le calcul tolèrerait également des erreurs introduites par la mise en oeuvre matérielle du problème, en supposant que ces erreurs (bruits et erreurs de précisions [87]) varient lentement pour éviter d'influencer le processus de recuit.

Ces résultats ont alors mené au premier brevet majeur de D-Wave Systems dans Amin et Steininger (2006) [16] et finalement à leurs travaux dans Johnson et al. (2011) [102], dans lesquels ils ont décrit la conception et la validation du processeur "Rainier" de D-Wave One. L'architecture permettait essentiellement d'utiliser un modèle programmable de l'hamiltonien d'Ising avec des champs magnétiques contrôlables. La contribution de Johnson et al. a été une avancée significative, elle introduisait pour la première fois un ordinateur quantique adiabatique qui pouvait être programmé et offrir une solution pour des problèmes de taille raisonnable. Les ordinateurs quantiques adiabatiques précédents étaient soit incapables d'être programmés (car c'était des prototypes destinés à la recherche en physique fondamentale), soit incapables de supporter plus de deux ou trois qubits. Les expériences décrites dans l'article de 2011 ont montré de quelle manière, un seul qubit et une seule cellule du processeur, possédaient des propriétés liées au recuit quantique. Un résultat qui a ensuite été étendu à 108 qubits par une équipe indépendante de Boixo et al. et ils ont rapidement constaté que, la faible connectivité entre les qubits (le graphe Chimera, tel que décrit au chapitre 3) avait pour implication que le processeur n'était pas encore un ordinateur quantique universel.

Depuis Johnson et al. (2011), plusieurs groupes indépendants ont effectué des analyses pour déterminer la rentabilité réelle des ordinateurs de D-Wave en terme de capacité de calcul. McGeoch et Wang (2013) [121] ont été en mesure de démontrer une accélération du temps de calcul significative pour certain problèmes sur le nouveau processeur D-Wave Two par rapport aux solutions logicielles classiques existantes. Ces résultats ont été rapidement réfutés par d'autres, et un consensus scientifique général sur la nature quantique des ordinateurs D-Wave n'a pas encore été validé. Ce débat controversé a déjà conduit à des gains d'efficacité significatifs dans le calcul classique, avec des algorithmes classiques hautement optimisés démontrés par Selby (2013) [154]. Ils ont été capable d'égaler et même de surpasser les performances des machines D-Wave sur les problèmes types QAP ("Quadratic Assignment Problem").
Malgré tout, cela n'a pas empêché la NASA, Lockheed Martin et Google (ainsi que l'Université de Californie) d'acquérir leurs propres D-Wave pour le développement d'algorithmes et de demeurer optimiste quant à leurs applications futures.

4.2.2 Controverse sur la nature des calculs dans les machines D-Wave

Il y a une controverse importante sur la question de la manifestation effective de phénomènes quantiques dans le calculateur D-Wave et plus particulièrement s'il présente des avantages en terme de gain de temps, d'efficacité de résolution de problèmes par rapport aux ordinateurs classiques actuels.

Plusieurs études ont suggéré que l'ordinateur D-Wave est effectivement un calculateur qui utilise les principes de mécanique quantique. L'article de 2011 dans *Nature* [102] stipule que les puces D-Wave possèdent des propriétés de mécanique quantique (intrication et effet tunnel) requises pour l'informatique quantique et dans *Science* [108] la même année, les auteurs ont montré qu'un D-Wave "*utilise au moins un minimum de mécanique quantique*".
En 2014, un article compare les données expérimentales de D-Wave avec trois applications de physique classique et une de physique quantique. Ils affirment que le modèle quantique est plus efficace pour traiter les données expérimentales [12]. En mai, un article contenant des résultats expérimentaux suggère bien l'utilisation de l'intrication entre les qubits dans le D-Wave [112].
Enfin en 2015, Google a publié un article comparant le calculateur D-Wave avec des algorithmes de recuit simulé et du Monte Carlo quantique. Ils ont découvert que le recuit quantique est plus rapide et plus efficace que ses homologues. Ils ont également découvert que le recuit quantique à l'échelle de D-Wave est similaire aux algorithmes de Monte Carlo quantiques, et est environ 108 fois plus rapide [60].

Malgré tout, d'autres études majeures suggèrent que l'ordinateur quantique D-Wave n'utilise pas vraiment les propriétés de la mécanique quantique.
En 2013 (et en réponse à l'article de McGoech et Wang [121] sur le résultat montrant qu'un D-Wave est plus rapide que quelques algorithmes classiques) les auteurs [154, 143] ont démontré que d'autres algorithmes classiques sont plus rapides que le calculateur D-Wave. En janvier 2014, d'autres équipes ont publié un modèle classique pour expliquer les données expérimentales de D-Wave, suggérant que ce n'est peut-être pas un ordinateur quantique [156], mais un calculateur qui se voudrait être quantique. Enfin, en juin 2014, une étude publiée dans *Science* a révélé que D-Wave ne produisait pas d'accélération quantique et qu'il n'y avait pas non plus de preuve que les machines utilisaient des phénomènes quantiques. Ceci inclut le fait que le recuit simulé sur un ordinateur classique était plus rapide que le recuit quantique de D-Wave [147].

4.3 Analyse et critiques

4.3.1 Hardware quantique

Les limites du calculateur quantique existent, en particulier, la connectivité limitée qui impose une augmentation significative du nombre de qubits requis pour résoudre des problèmes concrets (Chapitre 3). De plus, l'architecture de l'ordinateur n'est pas un graphe complet mais un graphe quadratiquement moins dense que les problèmes qui y sont définis. Enfin, en raison de certains problèmes dus au matériel, certains qubits peuvent être désactivés et non utilisables (tous les qubits ne sont pas parfaits et il y a une fraction d'eux 5 à 10 % qui sont inutilisables [28]). Idéalement, toute méthode mise au point doit résister à de légers changements dans la structure du graphe. Par conséquent, nous devons nous assurer de formuler le problème d'une manière qui nous permette de trouver une intégration du graphe rapidement et facilement. Ainsi, nous devons savoir si H est un sous-graphe isomorphe au graphe G. Cette question relativement récente, s'inspire de l'incorporation de graphe dans le hardware de D-Wave [44] et, à l'écriture du manuscrit, reste encore ouverte.

Controverse Deux questions englobent la controverse : le D-Wave utilise-t-il réellement les principes de la mécanique quantique et peut-il réellement résoudre des problèmes complexes plus rapidement que les ordinateurs classiques ?

Pour la première question, les études de l'état de l'art permettent d'affirmer avec une confiance raisonnable que les puces D-Wave sont capables au moins localement sur quelques qubits d'utiliser les principes de mécanique quantique. En revanche, nous ne savons pas si l'effet quantique est toujours présent lorsqu'un grand nombre de qubits sont utilisés. Le résultat le plus probant qui suggère que cela pourrait être vrai est celui de décembre 2015 [60], où le recuit quantique est bien meilleur à grande échelle que le recuit simulé. Cependant, il est toujours du même ordre de grandeur que d'autres algorithmes classiques comme le Monte Carlo quantique. En outre, l'augmentation de la taille des instances, et donc du nombre de qubits requis sur D-Wave, pourrait potentiellement nécessiter une certaine correction d'erreur en raison des difficultés d'ingénierie des puces quantiques avec de nombreux qubits.

Pour la deuxième question, il est important de se rendre compte qu'il y a deux aspects à comparer : "*Le premier est le temps d'exécution réel des machines, et le second est comment le temps d'exécution augmente à mesure que la taille d'entrée augmente*". Une comparaison du temps d'exécution réel sur D-Wave et sur un ordinateur classique peut nous dire s'il y a actuellement une accélération qui

peut être obtenue pour résoudre des problèmes. La plupart des recherches suggèrent que même si un D-Wave peut être plus rapide que des algorithmes spécifiques et des solveurs généralisés, il y a certains algorithmes classiques qui restent plus performants que le D-Wave.

4.3.2 Applications

Bien qu'il existe de nombreux groupes de recherche qui conçoivent et fabriquent actuellement des ordinateurs quantiques, les processeurs D-Wave sont actuellement les seuls commercialisés possédant un nombre important de qubits [68, 125]. Cela signifie que la capacité de développer et de tester des applications commerciales avec ces machines se trouve principalement dans les quelques entreprises disposant d'un accès à D-Wave (Google, la NASA et Lockheed Martin).
La NASA, dans le cadre du laboratoire d'intelligence artificielle quantique (QuAIL), a concentré ses recherches sur l'évaluation de performances des machines quantiques et les partenaires de Lockheed Martin de l'Université de Californie du Sud (USC) se concentre sur l'évaluation de leurs comportements. Ils ont également proposé une méthode pour effectuer de l'apprentissage machine sur un processeur adiabatique [142], mais aucune mise en œuvre directe publiée sur le matériel D-Wave.
De plus, Lockheed Martin se concentre sur les applications logicielles [89]. Dans Neven et al. [128], Google a fait la démonstration d'un algorithme d'apprentissage automatique pour la classification des images appliqué à un premier prototype de D-Wave [142]. Ils ont montré que "QBoost" (algorithme de Quantum Boosting) était capable de fournir un classement d'images avec une meilleure précision qu'une recherche classique, malgré le faible nombre de qubits disponibles. Cependant, leur algorithme ne fonctionnait pas aussi bien qu'un algorithme classique spécialisé. Ils ont attribué cela aux limites du matériel D-Wave ainsi qu'à la faible connectivité entre les qubits.

Avec l'augmentation du nombre de qubits dans le calculateur D-Wave (de 1000 à 2000 qubits), cela conduit à élargir le champs de recherche. Dans Neven et al. [127], QBoost a montré qu'il produisait des classements généralisés et exigeait moins de performance de calcul que l'algorithme d'amplification spécialisé. Dans Smelyanskiy et al. [160], une étude conjointe NASA/D-Wave Systems/USC a exploré diverses applications d'algorithmes quantiques pour différents problèmes spécifiques au vol spatial. Dans le même registre que Lucas (2014), Smelyanskiy et al. ont développé plusieurs "mappings" permettant, à partir d'un problème posé d'arriver aux formulations d'Ising. Toutefois, aucune étude ne décrit les mises en œuvre réelles ni les résultats sur le matériel D-Wave.

Controverse Toutes ces expérimentations sur les machines D-Wave montrent l'engouement pour cette branche de l'informatique quantique mais restent encore assez restrictives. Les résultats obtenus et publiés ont souvent une portée minime (voire moindre) que les algorithmes classiques qui sont développés par la suite. De plus, l'intégration des problèmes sur la machine reste assez succincte pour une vérification expérimentale des résultats obtenus. Dans cette optique, notre intérêt se porte sur la mise en œuvre pratique d'un problème (arbitraire ou non) sur une telle machine (Chapitre 6 et Chapitre 7). La question qui est posée ici est : comment intégrer en pratique un problème sur un D-Wave ?

4.3.3 Problèmes adressables

La capacité des ordinateurs quantiques à pouvoir utiliser un nombre exponentiel d'états a conduit à des algorithmes quantiques dont l'exécution s'est avérée plus rapide que celle des algorithmes classiques pour certains problèmes (la factorisation en entier de Shor et la recherche dans la base de données non triée de Grover [90, 158]).

Les ordinateurs de D-Wave sont basés sur un modèle programmable de verre de spin d'Ising. Ce problème particulier a fait l'objet d'études approfondies dans la littérature, et sa complexité informatique a été décrite pour la première fois en 1982 [23]. Les formulations mathématiques d'Ising de nombreux problèmes NP-complets et NP-difficiles sont décrites dans [117], ainsi que les techniques pour imposer les contraintes du problème dans les spins auxiliaires.
Plusieurs équipes ont essayé d'appliquer des variations des problèmes d'Ising au calcul quantique adiabatique afin de mieux comprendre les performances des machines D-Wave. Dans Boixo et al. [32] et McGeoch et Wang [121], les auteurs ont montré non seulement le comportement quantique des processeurs D-Wave, mais aussi une accélération du temps de calcul potentiellement significative. Dans Crosson et al. [56], les auteurs ont simulé des cas générés au hasard de problèmes d'Ising et ont constaté que des probabilités d'obtenir la meilleure solution plus élevées que des solveurs classiques.

Controverse Ces résultats montrent un intérêt certain pour l'expérimentation et l'exploitation des ordinateurs quantiques analogiques. Néanmoins, la comparaison des performances de ces machines avec les solveurs classiques n'est pas réellement équivalente. En effet, les études donnent les performances d'algorithmes optimisés (tel que CPLEX, qbsolv, etc.) par rapport à un algorithme quantique de type recuit simulé. Les qualités des solutions obtenues entre ces deux catégories de solveurs ne sont clairement pas comparable. La question centrale qui fait l'objet de ce manuscrit est de connaître les performances (au-delà du problème adressable) en terme

de temps d'exécution et de qualité de solutions d'un recuit quantique par rapport à un recuit simulé. Ces deux algorithmes décrits dans la section précédente présentent l'avantage d'être du même type (l'un utilisant des fluctuations de température et l'autre quantique) et ainsi être comparable en terme de coût d'une solution pour un problème donné. Dans la suite de ce manuscrit, la partie contribution fournit des éléments de réponses expérimentaux à cette question : le recuit quantique possède-t-il une accélération quantique par rapport au recuit simulé et obtient-il une meilleure qualité de solution que son homologue ?

Deuxième partie

Contributions

Chapitre 5

Etude des limites de la topologie d'interconnexion Chimera

Résumé

Ce chapitre examine la possibilité d'utiliser un ordinateur quantique analogique tel que commercialisé par D-Wave pour résoudre des problèmes QUBO de grande taille au moyen d'une seule invocation du recuit quantique. En effet, cette machine résout un problème de verre de spin avec des coefficients réglables mais soumis à des restrictions topologiques assez fortes sur l'ensemble des coefficients non nuls. Plutôt que de faire correspondre les variables du problème à de multiples qubits, une approche qui nécessite de nombreuses invocations du recuit pour résoudre des problèmes de petite taille, il est tentant d'étudier l'existence de relaxations creuses, conformes à la topologie d'interconnexion des qubits de la machine, donc exploitable en une seule invocation de l'oracle du recuit, mais fournissant néanmoins des solutions de bonne qualité au problème d'origine. Ce chapitre propose un dispositif expérimental qui vise à déterminer si de telles relaxations pratiques existent ou non ou, plutôt, si elles sont faciles à trouver. Nos expériences suggèrent que ce n'est pas le cas et, par conséquent, que la résolution de problèmes arbitraires, même de taille modérée, avec un seul appel à un recuit quantique n'est pas possible, du moins dans les limites de la topologie dite Chimera. Nous concluons le chapitre avec un certain nombre de perspectives que ces résultats impliquent sur la conception d'heuristiques tirant profit d'un oracle de recuit quantique pour résoudre des problèmes à grande échelle.

5.1 Introduction

L'informatique quantique suscite un regain d'intérêt avec les récentes annonces de plusieurs acteurs. La raison la plus évidente est que certains algorithmes quantiques peuvent résoudre en temps polynomial les mêmes tâches qui sont actuellement considérées comme non polynomiales

sur les ordinateurs classiques, comme la factorisation. Pourtant, d'un point de vue pratique, l'émergence d'ordinateurs quantiques capables de rivaliser avec les performances des ordinateurs classiques les plus puissants reste très spéculative dans un avenir proche. Une nouvelle approche consiste à construire des machines quantiques analogiques comme celles actuellement commercialisées par D-Wave. D'un point de vue abstrait, un tel ordinateur peut être considéré comme une machine spécialisée dans la résolution d'un problème d'optimisation de verre de spin en utilisant un algorithme similaire à un recuit simulé mais bénéficiant peut-être d'un facteur d'accélération quantique. Dans ce chapitre, nous recherchons des chemins de transformation polynomiaux qui permettent de convertir le plus efficacement possible des problèmes d'optimisation en type de problème traité par les machines quantiques analogiques. La première étape consiste à comprendre le fonctionnement d'une telle machine et à déterminer dans quelle mesure elle contribue efficacement à la résolution de problèmes complexes.

Dans ce contexte, ce chapitre est une étude expérimentale qui vise à déterminer si un ordinateur quantique analogique, topologiquement contraint, peut ou non être utilisé de manière directe pour résoudre des problèmes QUBO ou, de manière équivalente, des problèmes Ising avec un nombre de variables égal à son nombre de qubits. Cela suppose que des instances arbitraires QUBO admettent des relaxations creuses définies sur un graphe isomorphe à la topologie d'interconnexion des qubits et que ces relaxations soient faciles à trouver.

Contraintes topologique Les contraintes liées à la réalisation de l'architecture du processeur D-Wave [97] imposent une topologie d'interconnexion des qubits qui ne définit pas un graphe complet mais un graphe quadratiquement moins dense, dit graphe Chimera, qui correspond à un réseau de graphes bipartis 4×4. Lorsque le graphe engendré par les coefficients non nuls du problème QUBO n'est pas isomorphe à ce graphe d'interconnexion, le problème ne peut être directement résolu sur la machine. Dans ce cas, la difficulté est de rentabiliser au mieux le (faible) nombre de qubits sous les contraintes induites par cette topologie d'interconnexion.

Dans ce contexte, la contribution de ce chapitre est une première étude expérimentale qui vise à déterminer si un ordinateur quantique analogique avec une topologie Chimera peut ou non être utilisé directement pour résoudre des problèmes QUBO aléatoire avec un nombre de variables égaux au nombre de qubits physiques. Cela suppose que les instances doivent être relaxées pour être intégrées sur un graphe isomorphe à la topologie d'interconnexion des qubits. La deuxième question qui en découle est de savoir si de telles relaxations sont faciles à déterminer en pratique.

5.2 Positionnement du problème

Dans cette étude, nous choisissons volontairement l'une des premières puces à 128 qubits pour lesquelles nous supposons qu'il n'y a pas de qubits défectueux. Dans ce cas, le recuit est effectué sur un graphe matériel H qui est un sous-graphe de C_8 et les cas qui ne sont pas directement intégrés sur H ne sont pas directement résolus par le processus. Choi et al [51] montrent qu'un graphe complet K_q peut-être intégré dans le triangle supérieur du graphe Chimera C_q au prix de l'utilisation de plusieurs qubits physiques comme qubits logiques. Ce principe implique une réduction drastique du nombre de qubits logiques disponibles dans le système. De plus, un écart (important) peut se créer entre le nombre de qubits physiques et la taille (nombre de variables) des problèmes à mettre sur le graphe Chimera. La recherche de bons algorithmes et d'heuristiques pour trouver de telles correspondances reste un problème aussi en cours d'étude.

Deux grands types d'approches de résolution existent dans la littérature : décomposition du problème et recomposition d'une solution [35] ; résolution (une ou plusieurs fois) de relaxations du problème conformes aux contraintes de la topologie d'interconnexion [116] que nous allons détailler dans la section suivante.

5.2.1 Approches de résolution existantes

Plusieurs groupes indépendants ont mené des analyses pour déterminer la pertinence pratique des ordinateurs quantiques analogiques. Deux principaux types d'approches de résolution ont été envisagés :

— Décomposition du problème et reconstruction de la solution [35]. Cette méthode implique plusieurs difficultés majeures : il faut représenter un qubit logique (variable du problème) par plusieurs qubits physiques. Cette solution conduit à résoudre un problème moins dense en coefficients de couplage avec potentiellement une déstructuration importante du problème initial. Malgré de premiers résultats expérimentaux encourageants en matière de qualité des solutions obtenues, la rentabilité d'utilisation des qubits reste (très) faible au sens où elle engendre un grand nombre de sous problèmes très petits et autant d'appels au processeur quantique.

— Résolution de problèmes satisfaisants aux contraintes de la topologie d'interconnexion [116] avec plusieurs utilisations du recuit. Dans ce cas, venir intégrer le problème directement, enlève la contrainte topologique, mais malheureusement la résolution de ce genre de problème reste inexploitable. Le problème n'a pas d'application concrète et est souvent

défini pour cette étude. Néanmoins, il a l'avantage de pouvoir tester l'efficacité du recuit sur l'ensemble de l'architecture.

Dans [121], les auteurs utilisent des instances QUBO qui après transformations peuvent être résolues directement sur le matériel. Leurs expérimentations comprennent 100 instances avec des coefficients générés aléatoirement, chacune d'entre elles avec des tailles de problème prédéfinies : $n \in \{32, 119, 184, 261, 261, 361, 349, 439\}$ correspondant parfaitement aux sous graphes du Chimera. Les poids sont ensuite répartis uniformément en $\{-1, +1\}$. Ils indiquent également que l'ordinateur fonctionne pendant environ une demi-seconde et obtiennent la solution optimale dans 97% des cas pour les 600 instances (pour toutes les tailles) qu'ils ont générés en utilisant la plateforme D-Wave Two. En outre, ils affirment que certains cas faiblement connectés dans leurs instances peuvent être directement intégrés sur le matériel. Néanmoins, comme évoqué précédemment, les contraintes liées à la réalisation de l'architecture du processeur D-Wave [97] conduisent à une topologie d'interconnexion qui ne définit pas un graphe complet mais qui est quadratiquement moins dense. Lorsque le graphe généré par les coefficients non nuls du problème QUBO n'est pas isomorphe à ce graphe d'interconnexion, le problème ne peut pas être résolu directement sur la machine.

La première approche de Lewis et Glover [116] utilise des techniques de prétraitement pour réduire initialement la taille du problème. Ils considèrent des problèmes définis sur 1000 nœuds avec 5000 et 10000 arêtes et avec des coefficients uniformément répartis entre -10 et 10 et un petit nombre de valeurs aberrantes d'une amplitude comprise entre 25 et 250. Pour cela, ils introduisent 5 règles pour réduire la taille du problème et utilisent des transformations de graphe et l'ensemble de ces règles est combiné dans un préprocesseur appelé *QPro*. Ils concluent qu'avec les caractéristiques particulières de leur QUBO, ce qui prédomine pour obtenir une réduction efficace est la gamme d'éléments uniformément distribué ainsi que le nombre et les amplitudes des valeurs de J_{ij}. De plus, les pourcentages de réduction sont de l'ordre de 18% pour un problème à 10000 arêtes et de 20% pour un problème à 5000 arêtes.

Dans [35], les auteurs ont développé plusieurs algorithmes de type *divide-and-conquer* afin de résoudre des problèmes de plus grandes tailles. Pour résoudre un problème avec plus de variables qu'il n'y a de qubits disponibles, il faut diviser le problème en sous problèmes, les résoudre, puis construire à partir de ceux-ci une solution au problème original. *qbsolv* [1] est un exemple de ces solveurs. Il détermine une valeur minimale d'un grand problème QUBO en le

1. qbsolv algorithm en open source https://github.com/dwavesystems/qbsolv

divisant en morceaux et en le résolvant soit sur un système D-Wave soit via un solveur Tabu classique. En outre, ce logiciel propose également de représenter un qubit logique par plusieurs qubits physiques. Cette solution permet de résoudre un problème moins dense en coefficients de couplage avec une déstructuration potentiellement importante du problème initial.

Notre démarche se situe dans la seconde catégorie ; nous choisissons d'affecter directement les qubits logiques (variables du problème) aux qubits physiques mais au prix d'une perte de coefficients de couplage (ceux qui correspondent à des arrêtes non présentes dans le graphe Chimera). La relaxation doit alors être aussi dense que possible pour préserver au mieux la structure du problème et être généré par un algorithme rapide. Pour tester cette approche, nous engendrons plusieurs relaxations à l'aide d'algorithmes à démarrage multiple que nous résolvons ensuite à l'aide d'un recuit simulé classique, comme simulation d'un recuit quantique. Les solutions obtenues sont ainsi confrontées avec la fonction économique du problème QUBO initial.

5.2.2 Représentation des contraintes

Contrairement aux travaux précédents, nous essayons de prendre les problèmes QUBO qui ne sont pas isomorphes à la topologie d'interconnexion du graphe et de les prétraiter, en forçant à zéro certains coefficients qui ne peuvent pas être intégrés sur le graphe. De cette façon, nous obtenons des relaxations définies sur un graphe isomorphe à la topologie Chimera. Nos matrices ont un nombre de coefficients non nuls et non diagonaux J_{ij} qui est beaucoup plus élevé que le nombre d'arêtes disponibles dans le graphe Chimera. Tenter de traiter de telles matrices présente un potentiel pratique intéressant. En effet, si nous sommes capables de trouver une solution presque optimale en ne considérant que des sous-ensembles de la matrice à coefficients non nuls, nous pouvons théoriquement atteindre des tailles d'instance plus importantes, jusqu'au nombre de qubits de la machine. Ainsi, nous pourrions être en mesure de résoudre des problèmes QUBO plus complexes que ceux qui peuvent être résolus à l'heure actuelle.

En outre, une variable QUBO peut être définie sur n'importe quels qubits car la numérotation des variables n'est pas une contrainte fixe. Par conséquent, nous recherchons une bijection entre l'ensemble des variables QUBO et l'ensemble des qubits. Lorsque le couplage de deux qbits indiquent un coefficient non nul dans la matrice QUBO, leur couplage est défini comme ce coefficient. Par construction, ce processus donne une relaxation du problème QUBO original qui est conforme par construction à la topologie Chimera. Ensuite, notre objectif est de trouver des bijections qui conduisent, après un recuit quantique, à des assignations de variables QUBO et d'obtenir une solution de la fonction économique initiale de plus bas coût possible.

Pour que ces relaxations soient utiles en pratique, elles doivent à la fois exister et être faciles à trouver avec des algorithmes simples à mettre en œuvre. En effet, il n'est pas légitime d'exécuter un algorithme classique complexe comme étape de prétraitement pour un algorithme quantique, plus rapide, puisqu'un algorithme classique complexe pourrait être directement appliqué au problème initial dans la première étape. Néanmoins, nous tentons de déterminer expérimentalement si de telles relaxations pratiques existent ou non et si nous pouvons nous limiter à un algorithme de faible complexité pour les trouver.

5.3 Relaxation des coefficients

Nous allons tout d'abord définir deux relaxations, nommé $relax_1$ et $relax_2$ qui prennent en compte les contraintes topologiques partiellement ou en totalité.

Dans un premier temps, $relax_1$ est définie de la manière suivante : nous avons écarté les contraintes de topologie afin de voir si des relaxations creuses, c'est-à-dire des relaxations dans lesquelles nous gardons autant de coefficients non nuls de la matrice QUBO que le nombre d'arêtes (m) disponibles dans le graphe Chimera, mais sans tenir compte de savoir si le sous-ensemble de coefficients sélectionné est isomorphe à ce dernier. Comme nous l'avons déjà dit, nous ne nous limitons pas encore à des algorithmes de faible complexité, car notre premier objectif est de caractériser si une telle relaxation des coefficients existe. Nous essayons de le faire au moyen d'un algorithme de recherche locale qui commence par un sous-ensemble de coefficients m sélectionnés au hasard. Puis de manière itérative, nous permutons aléatoirement un coefficient sélectionné avec un coefficient non sélectionné. Enfin nous gardons ce changement si l'affectation des variables trouvée par un recuit simulé classique est meilleure que la précédente au bout de 1000 itérations sur la nouvelle fonction économique. Notre recherche locale implique donc l'exécution d'un recuit simulé pour calculer la fonction économique du problème relaxé. Cette méthode malgré le fait qu'elle soit coûteuse en appel au recuit n'est pas un problème puisque, nous cherchons à déterminer, si de telles relaxations existent. Le pseudo-code 1 illustre l'algorithme mis en œuvre pour $relax_1$.

Dans un deuxième temps, $relax_2$ (voir l'agorithme 2) est définie de la manière suivante : nous avons cette fois-ci inclus la topologie Chimera dans la définition de la relaxation. Pour ce faire, nous avons également utilisé le même type d'algorithmes mais la recherche locale commence par une affectation aléatoire des variables QUBO aux qubits. Ensuite, nous procédons à la permutation aléatoire de l'affectation de deux variables directement intégrable et nous gardons ce

Algorithm 1 $relax_1$

```
1:  procedure INITIALISATION(V1,V2,Chim)        ▷ nb variable Vi, nb arêtes Chim
2:      V1 = Chim,V2 = Chim                     ▷ Sélection aléatoire
3:      Sol : RS1, RS2                          ▷ Recuit simulé
4:      if RS1 ≤ RS2 then                       ▷ Sélection de la meilleure solution
5:          Sol : RS = min(RS1, RS2)
6:      for i = 0 to 1000 do
7:          Cin SWITCH Cout                     ▷ Sélection d'un coefficient mis à 0
8:          RS3
9:          if RS1 ≤ RS3 then
10:             Sol : RS = min(RS1, RS2)
11:     return RS                               ▷ Retourne la meilleure solution
```

changement, si l'affectation des variables trouvée en exécutant un recuit simulé sur l'instance topologiquement isomorphe induit par l'affectation (comme expliqué dans la section précédente) conduit à une meilleure solution que la précédente.

Algorithm 2 $relax_2$

```
1:  procedure INITIALISATION(V1,Chim,Iq)        ▷ nb variable V1, nb arêtes Chim, Iq qubits
2:      V1 = Chim and V1 ∈ Iq                   ▷ Sélection d'un sous graphe ismorphe au Chimera
3:      Sol : RS1                               ▷ Recuit simulé
4:      for i = 0 to 1000 do
5:          Cin ∈ Iq SWITCH Cout ∈ Iq           ▷ Sélection d'un coefficient mis à 0
6:          RS2
7:          if RS1 ≤ RS2 then
8:              Sol : RS = min(RS1, RS2)
9:      return RS                               ▷ Retourne la meilleure solution
```

Dans les deux cas, nous supposons [2] que si le type de relaxation qui nous intéresse existe et est relativement facile à trouver, alors le type d'algorithmes complexes ci-dessus devrait être capable de trouver ces dernières. Dans le cas contraire, cela donnerait un résultat négatif impliquant qu'il est difficile de résoudre des cas de QUBO dense avec un seul appel au recuit quantique. Sinon, cela indiquerait que la conception d'algorithmes rapides pour trouver ces relaxations est un problème intéressant.

2. Nous faisons l'hypothèse qu'avec le moyen algorithmique mis en œuvre par $relax_1$, il doit facilement nous convaincre que ces relaxations existent.

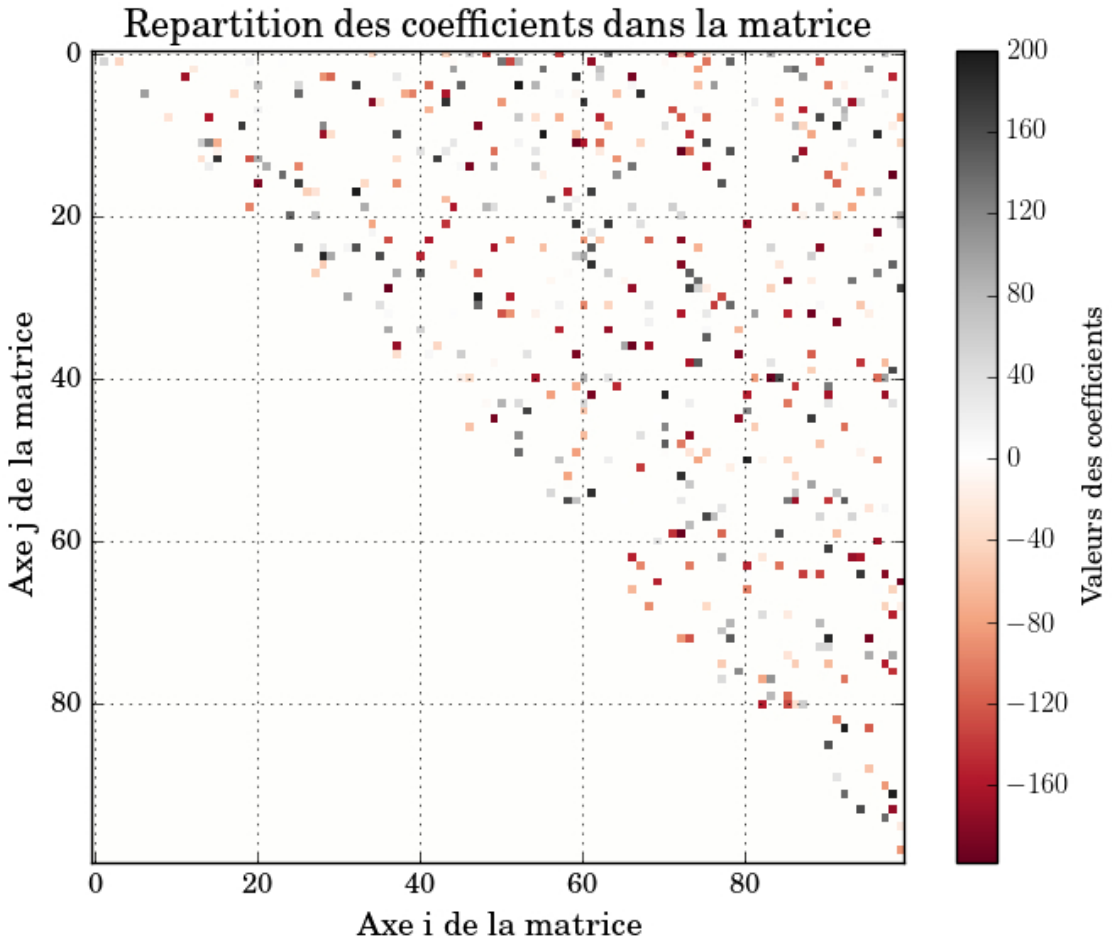

FIGURE 5.1 – Matrice bqp100-1 initiale

Problème bqp100-1 Les problèmes que nous avons traités et qui sont présentés dans le tableau à la section suivante sont des BQP (”Binary quadratic program”) disponibles avec le solveur *qbsolv* et des problèmes créés artificiellement. Pour illustrer notre algorithme de recherche, nous utilisons le problème *”bqp100-1”* présenté à la figure 5.1.

La figure 5.1 représente la matrice Q initiale avec 11 coefficients non nuls Q_{ii} sur la diagonale et 464 coefficients non nuls Q_{ij} non diagonaux. Après utilisation de $relax_1$ nous pouvons représenter toutes les grandeurs utilisées pour analyser notre solution sur la figure 5.2.

La figure 5.2 décrit l'ensemble des solutions obtenues avec plusieurs algorithmes détaillés ci-dessous. Sur l'axe des ordonnées, nous avons le coût de la solution obtenue et sur l'axe des abscisses nous avons le nombre de coefficients qui sont enlevés. 0 correspondant à aucun coefficient

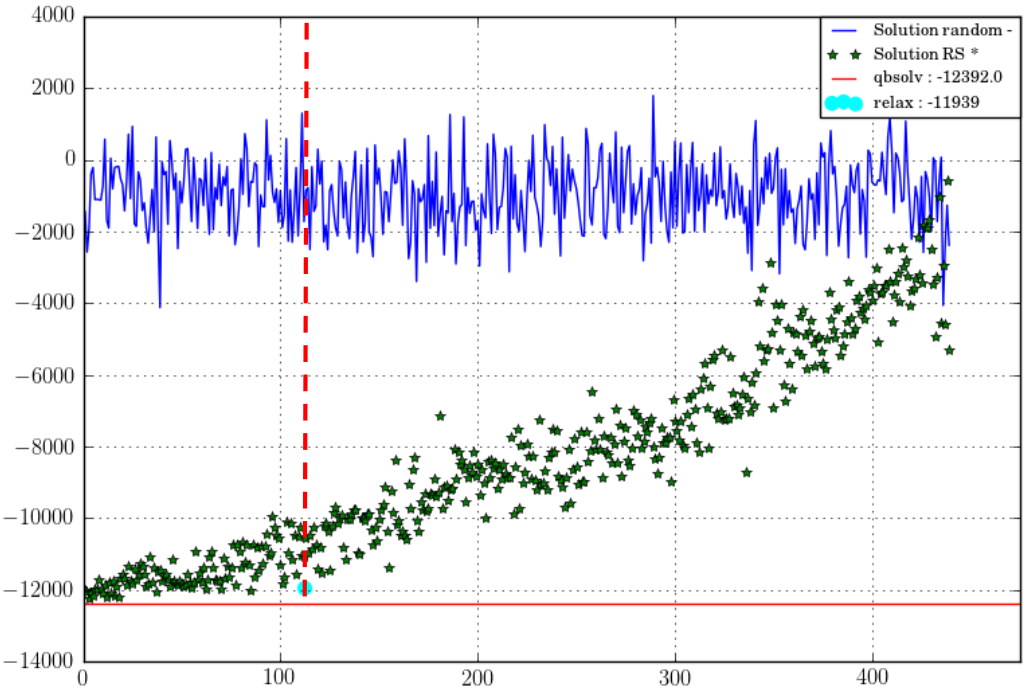

FIGURE 5.2 – Graphe de l'ensemble des solutions obtenues pour la matrice bqp100-1. L'axe des ordonnées donne le coût de la solution et l'axe des abscisses, le nombre de coefficients enlevés (0 étant la matrice initiale 5.1 et 464 la matrice vide). La courbe bleu donne les solutions avec un solveur aléatoire, les points verts avec un recuit, la droite rouge la solution obtenue avec le solveur *qbsolv* et le point bleu la meilleure solution avec notre algorithme $relax_1$. La ligne verticale en pointillés rouge représente la délimitation du nombre de couplages disponibles dans le D-Wave.

enlevé (matrice initiale 5.1) et 464 la matrice vide (uniquement des coefficients non nuls sur la diagonale). 3 solveurs ont été utilisés pour comparer notre solution $relax_1$:

- Random : Cette solution est un simple tirage aléatoire des variables ; c'est notre étalon haut. Si l'algorithme ne peut pas faire mieux qu'un simple tirage aléatoire, il nous faut redéfinir notre façon de creuser les matrices ;
- Recuit simulé (RS) : Cette solution est obtenue par un recuit simulé, il nous permet de quantifier notre algorithme par rapport à un recuit classique ;
- qbsolv : Cette solution est déterminée à partir de la matrice initiale $bqp100-1$ et correspond à notre étalon bas (solution obtenue et donné par D-Wave). L'objectif d'utiliser $qbsolv$ était de se rapprocher au mieux de cette solution ;
- $relax_1$: C'est la solution de l'algorithme mis en œuvre pour déterminer s'il est possible de creuser avec une perte de qualité de la solution maitrisée [3] notre matrice et d'obtenir une solution se rapprochant de l'étalon bas $qbsolv$. Le point bleu "**optim**" est situé au nombre d'arrêtes dans le graphe Chimera mais sans tenir compte de cette topologie.

Sur la figure 5.2, nous pouvons constater plusieurs tendances, la solution aléatoire reste quasiment constante au vu du tirage aléatoire des 0 et 1 de la solution QUBO, la solution RS augmente progressivement vers l'aléatoire quand le nombre de coefficients non nuls dans la matrice commence à être trop faible. Cette tendance de la courbe à tendre vers l'aléatoire en fonction du nombre de coefficients enlevés est assez intuitive, le nombre de coefficients non nuls dans la fonction économique devenant de plus en plus faible, le recuit simulé fait un tirage aléatoire. Enfin, pour $relax_1$, nous pouvons constater que la solution obtenue est très proche de celle de $qbsolv$ (3% d'écart). Nous pouvons affirmer que notre algorithme ne dégrade pas significativement la solution de plus bas coût connue.
Malgré tout, il est possible que la sélection initiale des coefficients de la matrice influence la bonne sélection des coefficients et que $relax_1$ ne puisse pas trouver une bonne solution (Le nombre d'itérations n'étant pas infini) et il sera très difficile pour $relax_1$ de déterminer un meilleur sous-ensemble de coefficients non nuls. Pour pallier cela, nous avons refait l'expérimentation sur 30 matrices creusées différemment afin de confirmer nos résultats sur bqp100-1. Nous pouvons en conclure qu'il serait facile de fournir un algorithme qui détermine un sous-ensemble de coefficients donnant une bonne qualité des solutions. Néanmoins, la densité de coefficients non nuls initialement présents dans le problème bqp100-1 n'est que de 10% du nombre total de coefficients qu'il est possible d'avoir dans une matrice triangulaire de taille 100 [4].

3. Si nous dégradons pas significativement la qualité de notre solution, alors il est possible de creuser efficacement la matrice.
4. Une matrice QUBO entièrement connectée possède 4900 coefficients non nuls non diagonaux

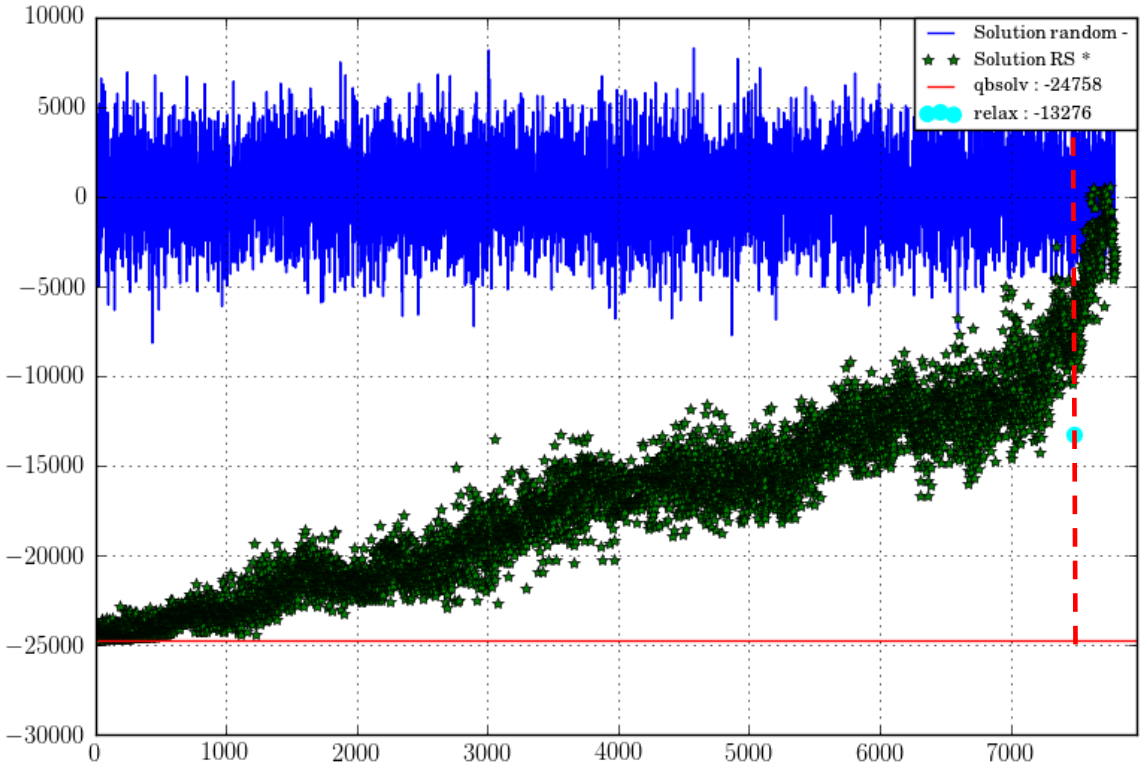

FIGURE 5.3 – Graphe de l'ensemble des solutions obtenues pour la matrice Rand-15. L'axe des ordonnées donne le coût de la solution et l'axe des abscisses, le nombre de coefficients enlevés. La courbe bleue donne les solutions avec un solveur aléatoire, les points verts avec un recuit, la droite rouge la solution obtenue avec le solveur *qbsolv* et le point bleue la meilleure solution avec notre algorithme $relax_1$. La ligne verticale en pointillés rouge représente la délimitation du nombre de couplages disponibles dans le D-Wave.

Problème Rand-15 Nous pouvons reproduire le même procédé mais cette fois avec une matrice possédant une densité de coefficients non nuls de 90% afin de comparer les solutions obtenues et de confirmer notre analyse sur bqp100-1.

Nous pouvons observer que l'aléatoire (random) et le recuit simulé (RS) se comportent de la même façon qu'avec le problème rand-15, mais notre algorithme n'arrive pas à déterminer une solution proche de *qbsolv* (écart d'environ de 54%). Cela implique qu'il est difficile de déterminer un bon sous-ensemble de coefficients non nuls malgré le nombre d'itérations effectués. Alors, plus la densité de coefficients est importante, plus il sera difficile de sélectionner un bon sous-ensemble de coefficients à garder et d'obtenir une solution proche de la meilleure solution connue.

Afin de confirmer ces analyses, nous avons reproduit ce protocole pour plusieurs matrices de densités de coefficients variables présentées dans la section résultats.

5.4 Résultats expérimentaux

QUBO	Size	Edges	Density (%)	Qbsolv	$relax_1$	$relax_2$	*rand*
factoring	127	703	10	-767	-538	-202	13
bqp100-1	100	464	10	-12392	-11967	-5356	-548
bqp100-5	100	459	10	-9629	-9062	-3764	-557
Rand-1	128	509	8	-5912	-5158	-1811	-85
Rand-2	128	522	7	-5458	-4932	-1938	258
Rand-3	128	499	7	-6413	-6056	-2872	112
Rand-4	128	1496	18	-13159	-9901	-6263	-102
Rand-5	128	1570	19	-9940	-6524	-3565	23
Rand-6	128	1503	19	-12269	-9071	-5029	-570
Rand-7	128	3874	48	-16814	-8125	-6280	1175
Rand-8	128	3845	47	-18205	-11378	-5707	507
Rand-9	128	3819	47	-16443	-9609	-6994	-213
Rand-10	128	5769	71	-19175	-9174	-7775	429
Rand-11	128	5833	72	-21317	-12801	-8669	-555
Rand-12	128	5881	72	-22498	-12879	-10236	-2034
Rand-13	128	7866	97	-27762	-13984	-10921	-1057
Rand-14	128	7842	96	-23323	-11958	-6979	-2004
Rand-15	128	7839	96	-24758	-13276	-9947	-676

TABLE 5.1 – Tableau des résultats expérimentaux sur les différentes matrices

Le tableau 5.1 fournit les résultats expérimentaux obtenus en exécutant les deux algorithmes sur trois instances QUBO tirées de la littérature ainsi que sur un certain nombre d'instances aléatoires et compare leurs solutions à celle d'un recuit simulé classique (nous avons de nouveau utilisé le logiciel *qbsolv* de D-Wave comme oracle de substitution du recuit quantique). Nos matrices aléatoires ont été générées avec des coefficients non diagonaux tirés au hasard dans l'ensemble $\{-100, \ldots, +100\}$ et sont appelées de Rand-1 à Rand-15. Afin d'observer l'évolution des solutions obtenues par les deux algorithmes en matière de densité d'instance, nous avons fixé un pourcentage de coefficients non nuls de 10% à 90%.

Les colonnes du tableau 5.1 fournissent respectivement : la taille de la matrice QUBO (Size), son nombre de coefficients non nuls (Edges), sa densité (Density), le coût obtenu en utilisant un

recuit classique simulé sur le problème (qbsolv), le coût obtenu par nos deux algorithmes de recherche ($relax_1$ et $relax_2$) et, enfin, le coût d'une solution choisie aléatoirement au QUBO ($rand$). Nous pouvons observer une tendance non surprenante entre la solution de recuit simulé (notre étalon) et la solution $relax_1$: plus la matrice QUBO est dense, plus la différence entre les solutions est importante. Nous pouvons également comparer la solution $relax_2$ avec les solutions $qbsolv$ et $relax_1$, et, dans ce cas, la différence entre les solutions est très importante (environ 50%). Cela est principalement dû au fait que l'algorithme $relax_2$ n'est pas capable de faire correspondre un nombre significatif de coefficients non nuls aux arêtes du graphe Chimera (30-40% du nombre total). Ainsi, les solutions $relax_2$ sont représentatives d'une relaxation de densité extrêmement faible. Néanmoins, la tendance est inversée par rapport à $relax_1$, plus la matrice est dense, plus la solution est proche de $relax_1$. Afin de rendre les solutions $relax_2$ plus caractéristiques des solutions $relax_1$, nous avons procédé à une nouvelle sélection et à une nouvelle relaxation mais cette fois en fonction du nombre de coefficients non nuls intégrés avec succès par $relax_2$. Nous constatons que leurs différences sont plus faibles (12%) et cet écart nous explique pourquoi nous obtenons une différence globale entre nos résultats. Ainsi, la topologie du graphe Chimera apparaît très contraignante, même pour les trois premières instances non aléatoires. Sans cette contrainte topologique, nos résultats seraient plus encourageants.

Deux observations peuvent donc être faites : premièrement, s'il était possible d'obtenir facilement un graphe isomorphe au graphe Chimera et d'avoir une solution proche de l'optimum connu, nous aurions dû pouvoir le trouver avec l'algorithme $relax_2$. Nous devons donc, soit redéfinir le graphe Chimera et connecter les qubits d'une manière différente, soit invoquer le recuit quantique plusieurs fois. Deuxièmement, en effectuant un nombre important d'exécutions (2000) sur les différentes matrices, il semble que nous puissions obtenir des solutions proches (écart entre 10 et 15 %) des solutions les plus connues sur nos problèmes.

5.5 Discussion

Dans ce chapitre, nos résultats suggèrent que les problèmes arbitraires de QUBO admettent des relaxations relativement creuses, ils n'admettent cependant pas les relaxations qui sont à la fois creuses et isomorphes à la topologie Chimera. En matière de conception d'algorithmes, cela suggère que de multiples invocations de l'oracle du recuit quantique sont nécessaires pour résoudre des cas pratiques de ces problèmes, soit en suivant une approche de décomposition (comme cela se fait déjà dans la littérature), soit en trouvant de nouvelles façons de combiner les solutions à plusieurs relaxations isomorphes en une solution de bonne qualité pour le problème

global. Les résultats préliminaires montrent que nous arrivons à obtenir des solutions proches des meilleures solutions connues moyennant un nombre important d'essais qu'il convient de mieux caractériser avant de conclure en terme de rentabilité.

Même si le nombres d'instances testées restent limités, nos expériences suggèrent qu'en effectuant une relaxation sur différentes densités de coefficients dans les matrices QUBO, la résolution de problèmes arbitraires, même de taille modérée, avec un seul appel au recuit quantique n'est pas possible, du moins dans les limites de la topologie. Par ailleurs, cette topologie elle-même peut être remise en question car d'autres topologies d'interconnexions, éventuellement dépendantes du problème dans une certaine mesure, peuvent avoir de meilleures propriétés de relaxation. En effet, une telle topologie devrait également être confrontée aux contraintes physiques de la conception du matériel de recuit quantique.

Chapitre 6

Performances du recuit quantique sur des instances ”difficiles” de couplage biparti

Résumé

Ce chapitre étudie expérimentalement le comportement des ordinateurs quantiques analogiques tels que ceux commercialisés par D-Wave lorsqu'ils sont confrontés à des cas de problèmes de couplage biparti de cardinalité maximale spécifiquement conçus pour être difficiles à résoudre au moyen d'un recuit simulé. Nous comparons un ”Washington” (2X) de D-Wave avec 1098 qubits utilisables sur différentes tailles d'instances et nous observons que pour tous ces cas, sauf les plus triviaux, la machine ne parvient pas à obtenir une solution optimale. Ainsi, nos résultats suggèrent que le recuit quantique, du moins tel qu'il est mis en œuvre dans un dispositif D-Wave, tombe dans les mêmes pièges que le recuit simulé et fournit donc des preuves supplémentaires suggérant qu'il existe des problèmes polynomiaux qu'une telle machine ne peut pas résoudre efficacement pour atteindre l'optimalité.

6.1 Introduction

Outre les analogies formelles entre le recuit simulé et le recuit quantique, il y a une analogie entre l'état actuel de la technique et celui du recuit simulé lors de son introduction. Il est donc utile de rappeler quelques faits sur cet algorithme. En effet, le recuit simulé a été introduit au milieu des années 80 [107, 46] et ses innombrables succès pratiques l'ont rapidement établi comme une méthode fiable pour résoudre approximativement les problèmes d'optimisations combinatoires difficiles. Ainsi, la communauté de l'informatique théorique a étudié en profondeur ses propriétés de convergence pour tenter de comprendre le comportement de la méthode dans le pire des cas. À cet égard, ces travaux, qui ont été réalisés à la fin des années 80 et au début des années 90, ont permis d'obtenir plusieurs résultats. Premièrement, lorsqu'il s'agit de résoudre

des problèmes d'optimisation combinatoire pour obtenir l'optimum, il est nécessaire (et suffisant) d'utiliser une règle de décroissance de la température logarithmique [86, 93, 132] conduisant à une convergence en temps exponentielle dans le pire des cas (un fait peu surprenant puisque nous savons que $P \neq NP$ dans le cadre du modèle de l'oracle boîte noire [22]). Deuxièmement, des cas particuliers de problèmes combinatoires ont été conçus pour exiger spécifiquement un nombre exponentiel d'itérations afin de parvenir à une solution optimale, par exemple sur le problème NP-difficile de coloriage [132] et, plus important pour ce chapitre, le problème polynomial de couplage bipartie de cardinalité maximale [150]. Enfin, une veine de travaux, toujours active aujourd'hui étudie le comportement asymptotique des problèmes combinatoires difficiles [43, 99, 151]. En particulier, il a été montré que sur certains problèmes combinatoires (QAP par exemple), le rapport de coût entre les solutions les plus avantageuses et les moins avantageuses pour les instances aléatoires tend (assez rapidement) vers 1 lorsque la taille des instances tend vers $+\infty$. Ces derniers résultats ont fourni des indices sur les raisons pour lesquelles des heuristiques simples telles que le recuit simulé semble fonctionner assez bien sur des instances de grande taille. Ainsi que sur le fait que les méthodes de résolution exacte de type "branch-and-bound" ont tendance à souffrir d'un effet de traîne (trouve rapidement la solution optimale mais n'arrive pas à prouver rapidement que c'est la solution optimale). Ces méthodes de résolution peuvent trouver des solutions optimales ou quasi-optimales relativement rapidement mais ne peuvent pas prouver leur optimalité dans un délai raisonnable.

Maintenant que ces résultats sont désormais établis, ils peuvent également contribuer à la compréhension en terme performance et être un étalon des machines adiabatiques [75] et déterminer s'ils offrent ou non un avantage quantique par rapport à certaines classes de calculs classiques. Cependant, comme il est peu probable qu'un paradigme en informatique quantique puisse conduire à des algorithmes efficaces pour résoudre des problèmes NP-difficiles, il est intéressant de savoir si l'informatique adiabatique présente ou non un avantage par rapport à l'informatique classique. Pourtant, en tant qu'analogue quantique du recuit simulé, tenter de démontrer un avantage quantique des algorithmes adiabatiques par rapport au recuit simulé semble être une question d'un grand intérêt. Au moment de la rédaction du manuscrit, ce problème fait l'objet de nombreux travaux qui, malgré des affirmations d'accélérations exponentielles dans des cas spécifiques [74] (qui ont également conduit au développement de la métaheuristique classique prometteuse du recuit quantique simulé [55]), qui suggère une condition de décroissance logarithmique de l'analogue de la température du recuit quantique, mais avec des constantes plus petites [149], ne donne qu'un avantage en $O(1)$ du recuit quantique sur le recuit simulé dans le cas général [10].

Le chapitre contribue à l'étude de la question du recuit quantique par rapport au recuit simulé en confrontant expérimentalement un ordinateur de type D-Wave aux cas pathologiques du problème de couplage bipartie de cardinalité maximale proposés par Sasaki et Hajek [150] afin de montrer que le recuit quantique était effectivement incapable de résoudre certains problèmes polynomiaux en temps polynomial. Démontrer une capacité à résoudre ces cas de manière optimale avec un recuit quantique laisserait entrevoir un avantage du recuit quantique dans le pire des cas par rapport au recuit simulé, dans le cas contraire, cela tendrait à démontrer que le recuit quantique reste soumis aux mêmes limitations que le recuit simulé et est donc incapable de résoudre efficacement certains problèmes polynomiaux.

Limitations de l'architecture de D-Wave Il convient de préciser que dans le cas des architectures actuelles, la liberté de choisir les constantes de couplage J_{ij} est fortement limitée par la topologie d'interconnexions des qubits dans le matériel. En particulier, cette topologie dite *Chimera* avec un nombre maximum de couplages inter spin limité à 6 par qubit. La Figure 3.2 du Chapitre 3 illustre le graphe Chimera avec 128 qubits, $T = (N_T, E_T)$, où les nœuds N_T sont des qubits et représentent les variables du problème avec des poids programmables (h_i), et les arêtes E_T sont associées aux couplages J_{ij} entre qubits ($J_{ij} \neq 0 \implies (i,j) \in E_T$).

Ainsi, si le graphe induit par les couplages non nuls n'est pas isomorphe au graphe Chimera, alors il faut recourir à plusieurs types de reformulation, parmi lesquels la duplication des qubits logiques sur plusieurs qubits physiques est la plus pertinante si le problème peut être intégrer sur le dispositif.

Ensuite, un recuit quantique va minimiser l'énergie de l'hamiltonien d'Eq. (3.1) en associant des poids (h_i) aux spins des qubits (σ_i) et des couplages (J_{ij}) aux couplages entre les spins des deux qubits connectés (σ_i et σ_j). À titre d'exemple, le système D-Wave 2X que nous avons utilisé possède 1098 qubits opérationnels et 3049 coupleurs opérationnels.

Comme nous l'avons déjà mentionné, un certain nombre de contraintes ont un impact sur l'efficacité pratique de ce type de machines. Dans [29], les auteurs soulignent quatre facteurs : la précision qui est limitée par les paramètres **h** et **J** dont les plages de valeurs sont également limitées. La plage de $h_i \in [-2, +2]$ et $J_{i,j} \in [-1, +1]$ est une limitation pour toutes les valeurs des variables. Si les valeurs de h_i et $J_{i,j}$ sont en dehors de leurs plages respectives, alors elles ne sont pas utilisables et ne sont pas intégrées. Si les problèmes à résoudre ne correspondent pas à la structure de l'architecture du graphe T, alors ils ne peuvent pas être intégrés et résolus

directement. Dans [28], les auteurs montrent que l'utilisation de grands écarts d'énergie dans la représentation d'Ising du modèle à optimiser peut grandement atténuer certaines des limites intrinsèques du matériel comme les précisions sur le couplage et les bruits dans les mesures de spin. Ils suggèrent également d'utiliser le couplage ferromagnétique d'Ising entre les qubits (c'est-à-dire de faire une duplication des qubits) pour atténuer les problèmes liés à la densité du graphe Chimera. Toutes ces suggestions peuvent être considérées comme des recommandations intéressantes (que nous avons fait de notre mieux pour suivre) pour utiliser la machine et résoudre des problèmes d'Ising ou QUBO avec des probabilités plus élevées d'obtenir la meilleure solution malgré les limitations du matériel et de l'architecture.

Pour palier ces facteurs, des algorithmes de prétraitement sont nécessaires pour adapter le graphe d'un problème au matériel. Les machines quantiques analogiques sont limitées par le nombre de variables (y compris la duplication) qui peuvent être intégrer sur le matériel. Les graphes de plus grande taille nécessitent le développement d'approches hybrides (classiques et/ou quantiques) ou la reformulation du problème pour l'adapter à l'architecture. Par exemple, pour une matrice de 128 $\times$128, le nombre de coefficients possibles J_{ij} est de 8128 dans le pire des cas, alors que le graphe Chimera qui associe 128 qubits (4×4 cellules unitaires) a seulement 318 de coupleurs. La topologie ne représente donc que $\sim$ 4% du nombre total de couplages nécessaires pour intégrer une matrice de 128×128 dans le pire cas. Bien que des études préliminaires (voir chapitre 5 et [169]) nous avons montré qu'il est possible d'obtenir des solutions proches des minimas connus pour des matrices $\mathbf{Q}$ avec des densités supérieures à celles autorisés par le graphe en éliminant certains coefficients. De plus, au chapitre 5, nous avons montré qu'appliquer des relaxations sur des matrices QUBO de manière isomorphe à la topologie est difficile à réaliser en pratique et nous n'obtenons pas de bon résultats (écart de plus 50% avec la solution donnée par D-Wave). Il s'ensuit que la résolution d'instances QUBO denses et de grandes tailles nécessite un pré et un post-traitement non négligeable ainsi qu'un nombre important d'invocations du recuit quantique.

Dans un premier temps, nous allons nous étudier expérimentalement les performances d'un D-Wave "Washington" (2X) avec 1098 qubits opérationnels sur différentes tailles d'instances pathologiques du problème de couplage de cardinalité maximale. Ce chapitre est organisé comme suit : la section 6.2 examine le problème de couplage de cardinalité maximale et introduit la famille de graphes G_n sur lesquels sont définis nos instances pathologiques. Nous allons aussi détailler dans cette section comment nous construisons les instances QUBO à intégrer à partir

de ces instances. Ensuite, la section 6.3 détaille notre approche expérimentale et nos résultats et, enfin, la section 6.4 conclut sur une discussion des résultats et un certain nombre de perspectives.

6.2 Résolution du problème de couplage biparti sur un D-Wave

Pour cette étude, nous nous sommes intéressés à un problème bien connu, le problème de couplage biparti de cardinalité maximale. Contrairement au chapitre précédent où nos matrices étaient factices, dans cette section nous allons développer les étapes nécessaires pour venir intégrer sur un D-Wave le problème de couplage.

6.2.1 Couplage biparti de cardinalité maximale et famille de graphes G_n

Étant donné un graphe (non orienté) $G = (V, E)$, le problème de couplage de cardinalité maximale demande de trouver $M \subseteq E$ tel que $\forall (e, e') \in M^2$, $e \neq e'$, nous avons $e \cap e' = \emptyset$ et tel que $|M|$ est maximum. C'est un problème polynomial bien connu, traité dans presque tous les livres sur l'optimisation combinatoire (par exemple [110]). L'algorithme pour le résoudre dans le cas général, l'algorithme d'Edmond, est un chef-d'œuvre algorithmique. En outre, lorsque G est biparti, c'est-à-dire lorsqu'il existe deux sous-ensembles collectivement exhaustifs et mutuellement exclusifs de E, dans A et dans B, de telle sorte qu'aucune arêtes n'a ses deux sommets en A ou en B, le problème devient un cas particulier du problème du flot maximum et peut être traité avec plusieurs algorithmes plus simples [110].

Il est donc intéressant qu'une méthode apparemment aussi efficace que le recuit simulé puisse être piégée par des cas spéciaux de ce dernier problème plus simple. En effet, dans un article de 1988 [150], Sasaki et Hajek, ont étudié la famille de cas particuliers du problème de couplage biparti de cardianlité bipartie. Soit G_n désignant le graphe (non orienté) avec les sommets $\bigcup_{i=0}^{n} A^{(i)} \cup \bigcup_{i=0}^{n} B^{(i)}$ où chacun des $A^{(i)}$ et des $B^{(j)}$ ont une cardinalité $n+1$ (la numérotation des sommets va de 0 à n), où le sommet $A_j^{(i)}$ est connecté au sommet $B_j^{(i)}$ et où le sommet $B_j^{(i)}$ est connecté aux sommets de $A^{(i+1)}$ (pour $i \in \{0, \ldots, n\}$ et $j \in \{0, \ldots, n\}$). Ces graphes sont clairement bipartis car ils n'ont ni deux sommets dans $\bigcup_{i=0}^{n} A^{(i)}$ ni deux sommets dans $\bigcup_{i=0}^{n} B^{(i)}$ connectés entre eux. Ces graphes présentent donc une structure très particulière qui alterne entre des sous-ensembles de sommets faiblement et densément connectés, comme l'illustre la figure 6.1 pour G_3.

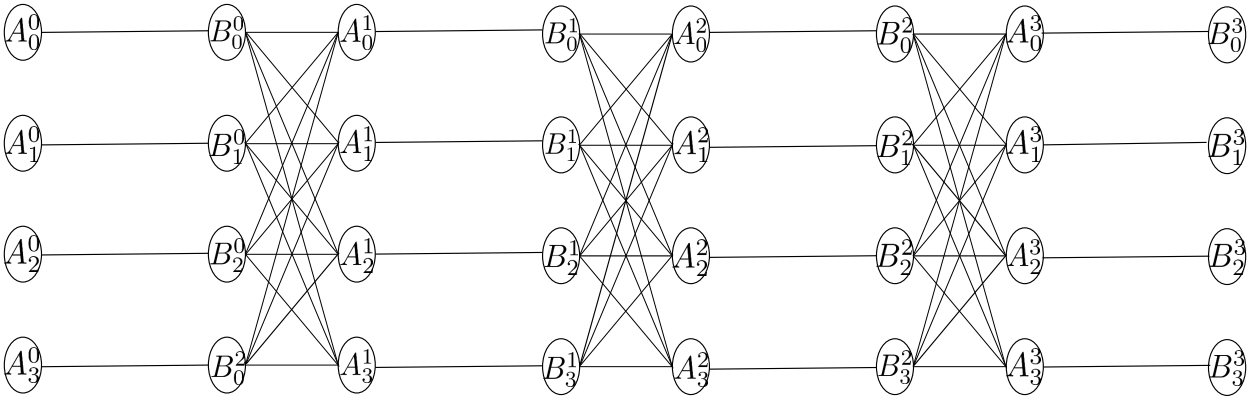

FIGURE 6.1 – Graphe associé à l'instance G_3

Dans le cas particulier du problème de couplage biparti, le cardinal maximum sur G_n peut être résolu par n'importe quel algorithme. Il est facile de se convaincre qu'un couplage est maximal sur G_n en sélectionnant simplement toutes les arêtes reliant les sommets de $A^{(i)}$ aux sommets de $B^{(i)}$ (pour $i \in \{0, \ldots, n\}$). En outre toutes les arêtes des sous-ensembles de sommets faiblement reliés donne la solution optimale et c'est l'unique moyen de l'obtenir. Cela conduit donc à un couplage maximale de cardinalité en $(n+1)^2$ (voir figure 6.2 pour G_3).

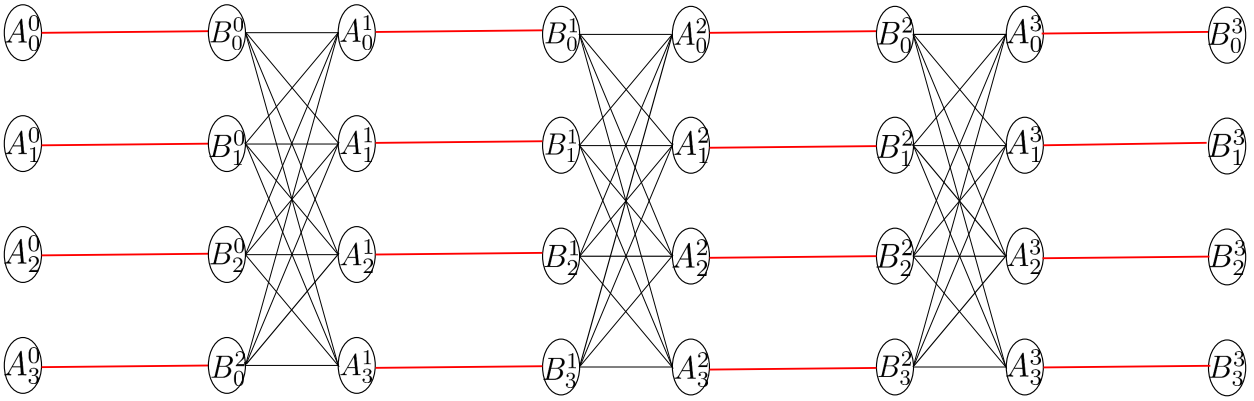

FIGURE 6.2 – Graphe à l'instance G_3 avec la sélection des arêtes (en rouge) de la solution optimale

Nous avons donc un cas spécial et simplifié d'un problème polynomial, mais le résultat fondamental de Sasaki et Hajek indique que l'espérance mathématique du nombre d'itérations requis

par une grande classe d'algorithmes de type recuit (classique) pour atteindre un couplage maximum sur G_n est en $O(e^n)$. La famille G_n constitue donc un "terrain de jeu" intéressant pour étudier le comportement du recuit quantique sur ces instances difficiles pour un recuit classique.

6.2.2 Instances QUBO

Afin que nos résultats soient reproductibles, nous décrivons ci-dessous la conversion des exemples du problème de couplage biparti en problème d'optimisation binaire quadratique sans contrainte (QUBO) dont les machines D-Wave ont besoin comme instances d'entrées.

Soit $G = (V, E)$ qui désigne le graphe non orienté pour lequel un couplage maximum est souhaité. Nous désignons $x_e \in \{0, 1\}$, pour $e \in E$, la variable qui indique si e est dans le couplage.

Nous devons donc maximiser,

$$\sum_{e \in E} x_e,$$

sous réserve que chaque arête v soit couverte au maximum une fois, c'est-à-dire $\forall v \in V$,

$$\sum_{e \in \Gamma(v)} x_e \leq 1, \tag{6.1}$$

où $\Gamma(v)$, dans les notations standard de la théorie des graphes, désigne l'ensemble des arêtes incidentes à v.

Afin de transformer ce problème en un QUBO, nous devons déplacer les contraintes ci-dessus dans la fonction économique, par exemple en maximisant,

$$\begin{aligned}
& \sum_{e \in E} x_e - \lambda \sum_{v \in V} \left(1 - \sum_{e \in \Gamma(v)} x_e \right)^2 \\
= & \sum_{e \in E} x_e - \lambda \sum_{v \in V} \left(1 - 2 \sum_{e \in \Gamma(v)} x_e + \sum_{e \in \Gamma(v)} x_e \sum_{e' \in \Gamma(v)} x_{e'} \right) \\
= & \sum_{e \in E} x_e - \lambda |V| + \sum_{v \in V} \sum_{e \in \Gamma(v)} 2\lambda x_e - \sum_{v \in V} \sum_{e \in \Gamma(v)} \sum_{e' \in \Gamma(v)} \lambda x_e x_{e'}.
\end{aligned}$$

Nous pouvons négliger le terme constant $-\lambda|V|$ qui n'affecte pas la fonction économique et considérer,

$$\sum_{e\in E} x_e + \sum_{v\in V}\sum_{e\in\Gamma(v)} 2\lambda x_e - \sum_{v\in V}\sum_{e\in\Gamma(v)}\sum_{e'\in\Gamma(v)} \lambda x_e x_{e'}$$

Nous devons enfin réagencer les coefficients pour construire une véritable matrice QUBO. Soit $e = (v, w)$, la variable x_e a un coefficient à 1 au premier terme, 2λ au deuxième terme (pour v) puis encore 2λ au deuxième terme (pour w) puis $-\lambda$ au troisième terme (pour v et $e' = e$) et $-\lambda$ au troisième terme (pour w et $e' = e$). Ainsi, les termes diagonaux de la matrice QUBO sont,

$$Q_{ee} = 1 + 4\lambda - 2\lambda = 1 + 2\lambda.$$

Ensuite, si deux arêtes distinctes e et e' ont un sommet commun, le produit des variables $x_e x_{e'}$ a un coefficient en $-\lambda$, au troisième terme, lorsque v correspond à un sommet partagé entre les deux arêtes, et ce, deux fois. Ainsi, pour $e \neq e'$,

$$Q_{ee'} = \begin{cases} -2\lambda & \text{if } e \cap e' \neq \emptyset, \\ 0 & \text{sinon.} \end{cases}$$

Alors, en prenant $\lambda = |E|$ la limite supérieure pour le coût de tout couplage, toute solution qui enfreint au moins une des contraintes (6.1) ne peut être optimale. Par exemple pour G_1 le plus petit graphe, nous obtenons ainsi un QUBO à 8 variables avec la matrice suivante,

	0	1	2	3	4	5	6	7
0	17	0	−16	−16	0	0	0	0
1	0	17	0	0	−16	−16	0	0
2	0	0	17	−16	−16	0	−16	0
3	0	0	0	17	0	−16	0	−16
4	0	0	0	0	17	−16	−16	0
5	0	0	0	0	0	17	0	−16
6	0	0	0	0	0	0	17	0
7	0	0	0	0	0	0	0	17

Pour lesquels, par exemple, un couplage maximum a un coût de 68, la deuxième meilleure solution 53 et la pire solution (qui consiste à sélectionner toutes les arêtes) a un coût de -56.

6.3 Résultats expérimentaux

6.3.1 Implementation sur D-Wave

Dans cette section, nous détaillons les étapes nécessaires pour intégrer et résoudre concrètement les instances QUBO associées à G_n, $n \in \{1, 2, 3, 4\}$, sur un DW2X exploité par l'Université de Californie du Sud.

Les matrices QUBO définies dans la section précédente ne sont pas directement transposables sur la topologie d'interconnexion Chimera et, par conséquent, nous devons recourir à la duplication des qubits. Bien que le "pipeline" logiciel de D-Wave automatise ce processus de duplication, ce besoin de duplication (ou, de manière équivalente la faible densité) restreint fortement la taille des instances que nous avons pu intégrer. Nous avons dû nous limiter à G_4, avec 125 variables qui ont nécessité d'utiliser 951 des 1098 qubits disponibles. Le tableau 7.1 indique le nombre de qubits requis pour chacune de nos quatre instances.

	#var.	#qubits	average dup.	max. dup.
G_1	8	16	2.0	6
G_2	27	100	3.7	6
G_3	64	431	6.7	18
G_4	125	951	7.6	18

TABLE 6.1 – Nombre de qubits requis pour traiter les instances QUBO associées à G_1, G_2, G_3 et G_4.

En outre, les figures 6.3, 6.4, 6.5 et 6.6 fournissent l'histogramme du nombre de duplications pour G_1, G_2, G_3 et G_4.

Au final, la duplication des qubits conduit à un QUBO avec plus de variables et une fonction économique qui comprend un ensemble supplémentaire de contraintes de pénalité pour favoriser les solutions pour lesquelles les qubits représentent la même variable. Il faut que tous les qubits physiques se retrouvent effectivement avec la même valeur pour représenter un qubits logique. Plus précisément, chaque paire de qubits distincts q et q' (associée à la même variable QUBO) ajoute un terme de pénalité de la forme :

$$\varphi q(1 - q')$$

Où la constante de pénalité φ est choisie comme étant le coût de la pire solution possible du QUBO initial, obtenu pour un vecteur rempli de 1. C'est-à-dire une solution qui sélectionne toutes les arêtes du graphe et qui maximise les violations hautement pénalisées des contraintes de cardinalité. Ainsi nous garantissons qu'une solution n'est pas optimale si elle viole au moins une de ces contraintes de cohérence. A noter que nous passons d'un problème de maximisation dans la section 6.2.2 à un problème de minimisation tel que requis par la machine.

Enfin, la duplication des qubits conduit à un QUBO plus grand, en taille et en nombre de coefficients non nuls, le graphe associé doit être isomorphe à la topologie. Il peut être intégré sur le dispositif après une renormalisation de ses coefficients pour s'assurer que les termes diagonaux Q soient en $[-2, 2]$ et les non diagonaux en $[-1, 1]$.

6.3.2 Résultats expérimentaux sur G_1, G_2, G_3, G_4.

Dans cette section, nous présentons les résultats obtenus sur le D-Wave pour nos instances G_1, G_2, G_3, G_4. Comme nous l'avons déjà souligné, en raison de la faible densité de couplage dans la topologie d'interconnexion des qubits, nos instances de QUBO n'étaient pas directement intégrées sur la machine D-Wave et nous avons dû recourir à des duplications de qubits. Ce besoin de duplication de qubits nous a limité à G_4 avec 125 variables binaires, qui est un problème combinatoire de taille non négligeable. Pourtant, pour le résoudre, nous avons dû mobiliser environ 87% des 1098 qubits de la machine. Les résultats ci-dessous ont été obtenus en faisant tourner 10 000 fois, le recuit quantique avec un temps de recuit de 20 μs (bien que nous ayons également expérimenté avec 200 et 2000 μs, cela n'a pas affecté les résultats de manière significative). Le tableau 6.2 résume les principales statistiques des résultats obtenus. Les paragraphes suivants traitent chaque cas plus en détail.

	opt.	best sol.	worst sol.	mean	median	stdev
G_1	-68	-68	-6	-67.4	-68	3.2
G_2	-495	-495	-89	-402.9	-388	47.8
G_3	-2064	-1809	-549	-1460.8	-1549	136.4
G_4	-6275	-5524	-2109	-4492.4	-4525	391.8

TABLE 6.2 – résultats obtenus sur le D-Wave pour les instances G_1, G_2, G_3, G_4

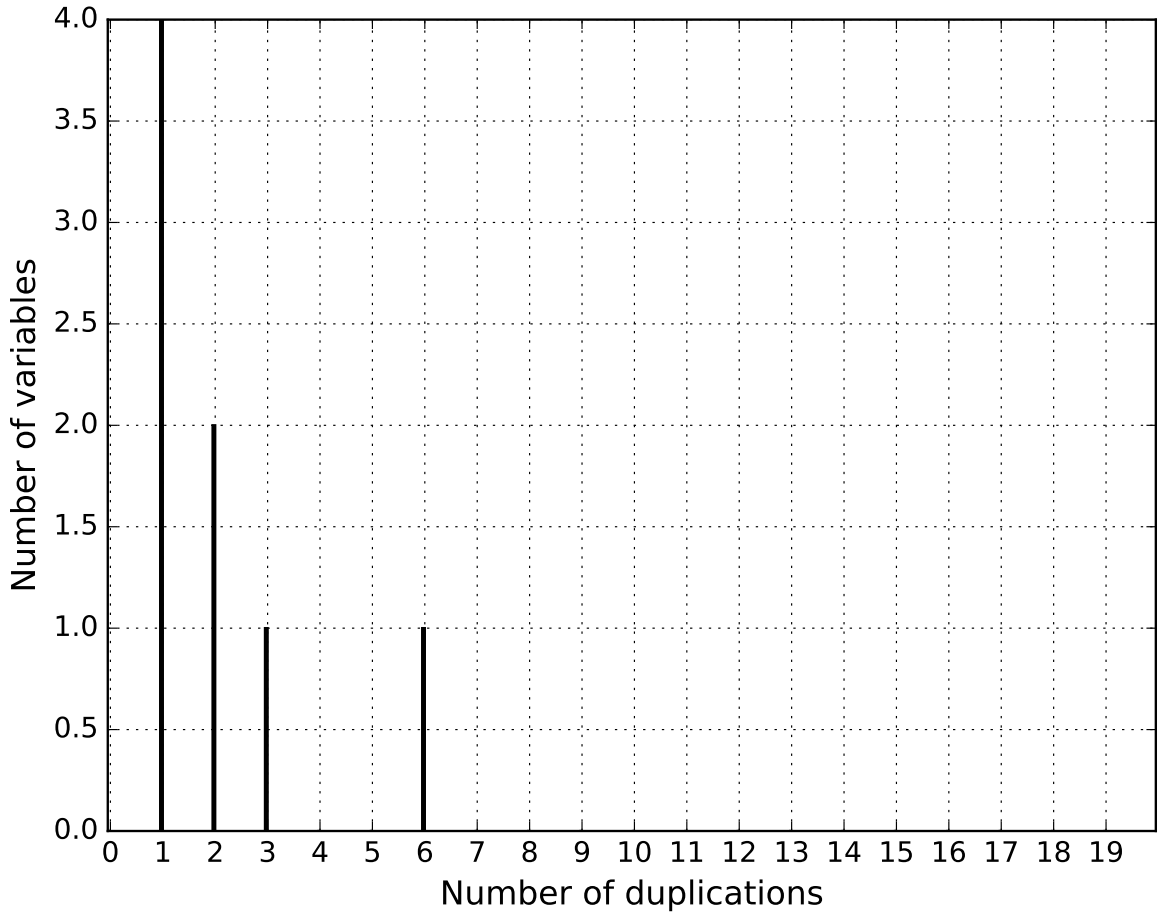

FIGURE 6.3 – Histogramme du nombre de duplications pour G_1. La duplication maximale est de 6 qubits.

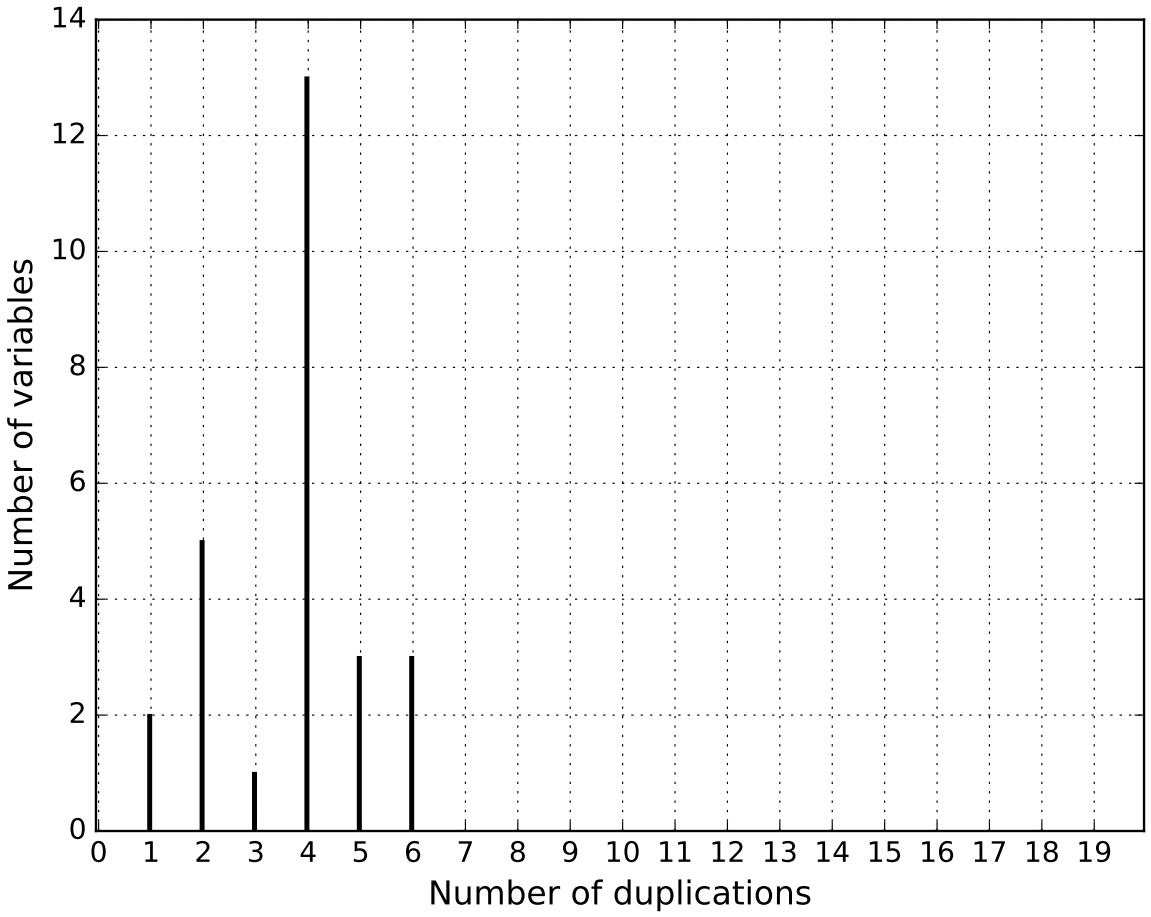

FIGURE 6.4 – Histogramme du nombre de duplications pour G_2. La duplication maximale est de 6 qubits.

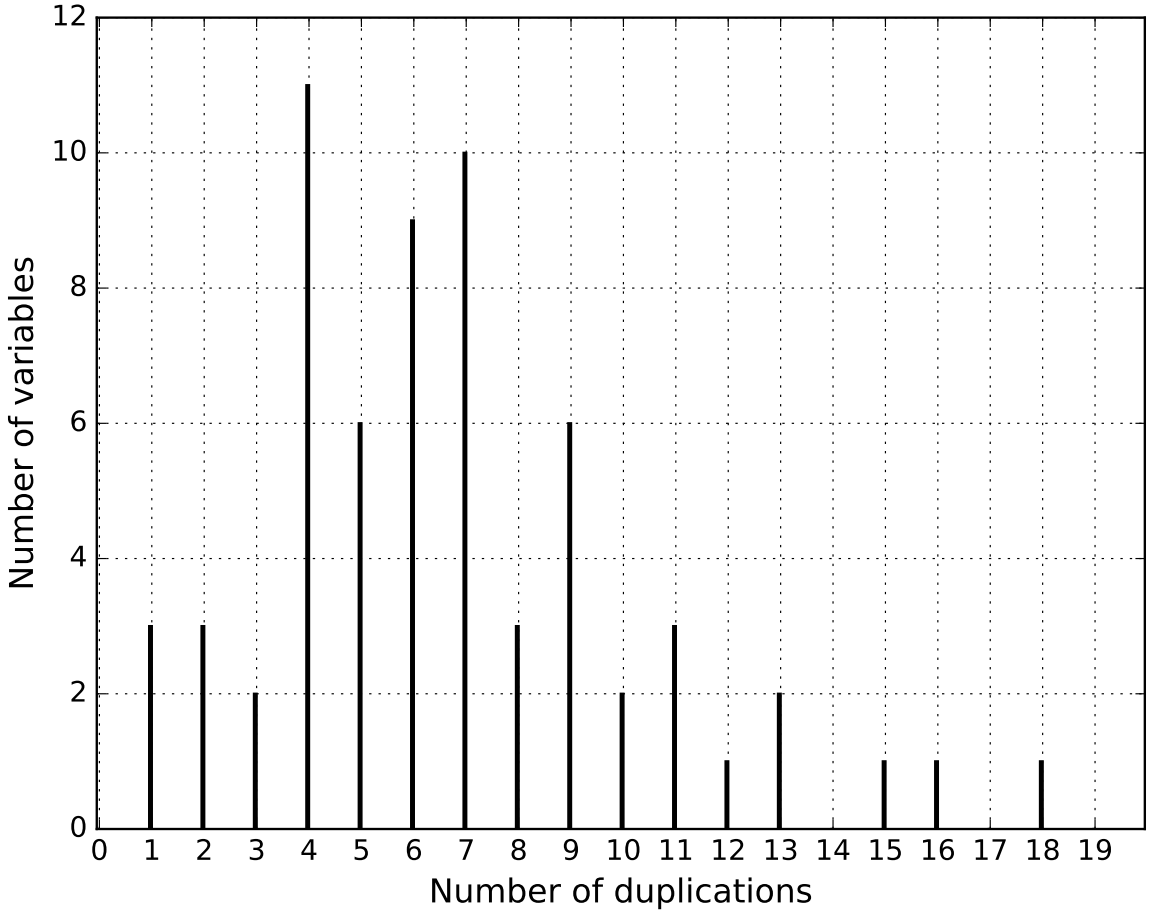

FIGURE 6.5 – Histogramme du nombre de duplications pour G_3. La duplication maximale est de 18 qubits.

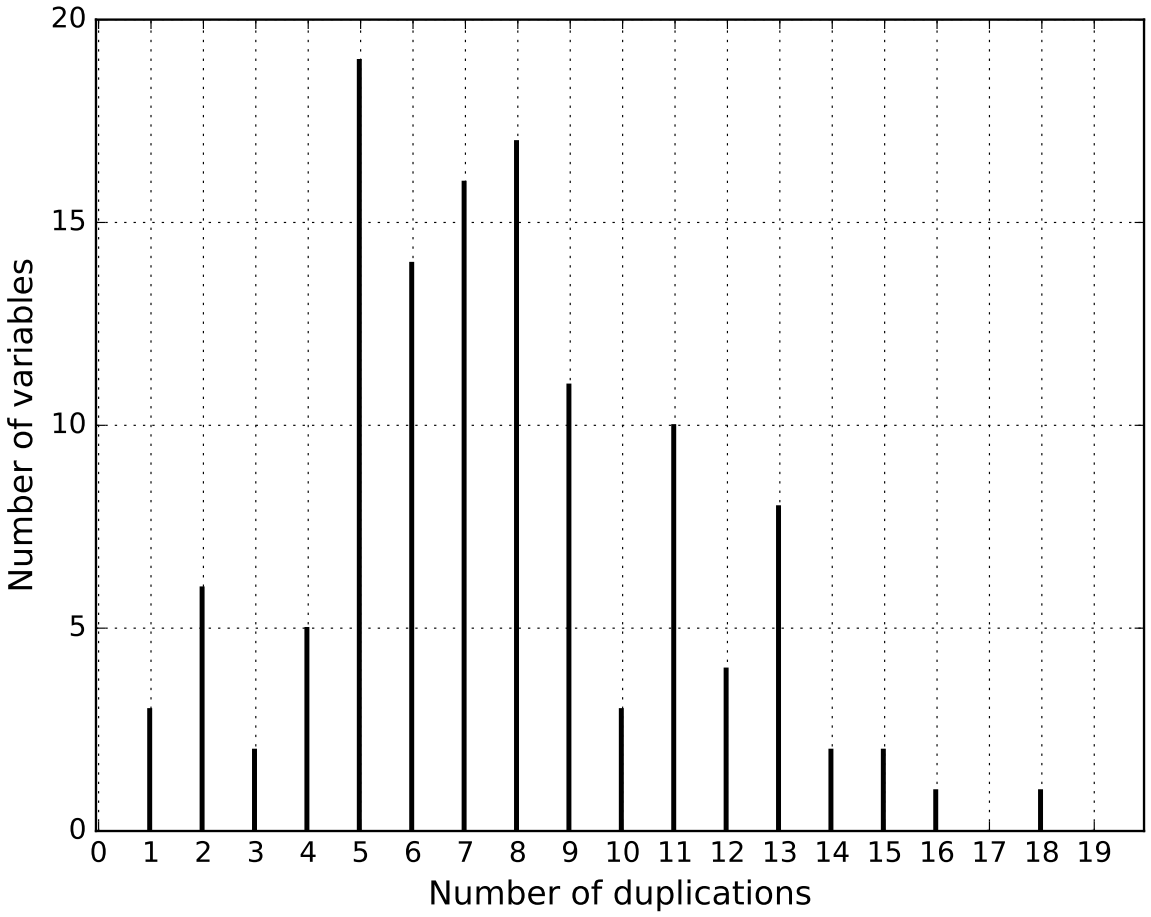

FIGURE 6.6 – Histogramme du nombre de duplications pour G_4. La duplication maximale est de 18 qubits.

Instance G_1 : Cette instance conduit à un graphe avec 8 sommets et 8 arêtes, avant duplication le QUBO associé possède 8 variables et 12 coefficients non diagonaux non nuls [1]. Mettre ce QUBO sur la machine D-Wave a nécessité 16 qubits comme le montre la figure 6.7. Sur 10 000 exécutions du recuit quantique, la solution optimale a été obtenue 9673 fois. Le tableau 6.3 et la figure 6.8 illustrent les meilleures (avec un coût de -68) et les pires solutions (avec un coût de -6) obtenues pour G_1 (la solution médiane est identique à la meilleure solution). Il est intéressant de noter que la pire solution obtenue viole les contraintes de duplication car les 6 qubits représentant la variable 6 n'ont pas tous la même valeur en sortie (4 d'entre eux sont à 0, donc dans ce cas particulier, le post-traitement de la solution au moyen du vote majoritaire donne la solution optimale). La figure 6.9 montre l'histogramme de la fonction économique des solutions données par le D-Wave (mais renormalisée) pour les 10 000 exécutions du recuit que nous avons effectuées. En outre, comme certaines des solutions obtenues sont incohérentes en ce qui concerne la duplication, la figure 6.10 montre l'histogramme de la fonction économique pour les solutions dans lesquelles les incohérences de duplication ont été corrigées par un vote majoritaire.

qubits	variable	best	worst
1040	0	1	1
1041	4	0	0
1042	5	0	0
1044	5	0	0
1045	4	0	0
1047	5	0	0
1048	1	1	1
1050	6	0	0
1051	3	1	1
1052	7	0	0
1053	2	1	1
1054	6	0	0
1055	3	1	1
1137	6	0	0
1143	6	0	1
1146	6	0	0
1151	6	0	1

TABLE 6.3 – Sélection des solutions (les meilleures et les pires) pour le QUBO étendu associé à G_1. La meilleure et la médiane des solutions sont identiques (et optimales) dans le cas de G_1 et ont coûté -68. La pire solution a coûté 6. Dans cette dernière solution, les 6 qubits représentant la variable 6 n'ont pas les mêmes valeurs.

1. Dans la topologie Chimera les coefficients diagonaux ne sont pas contraignant car il n'y a pas de limitation sur les qubits auto-couplés.

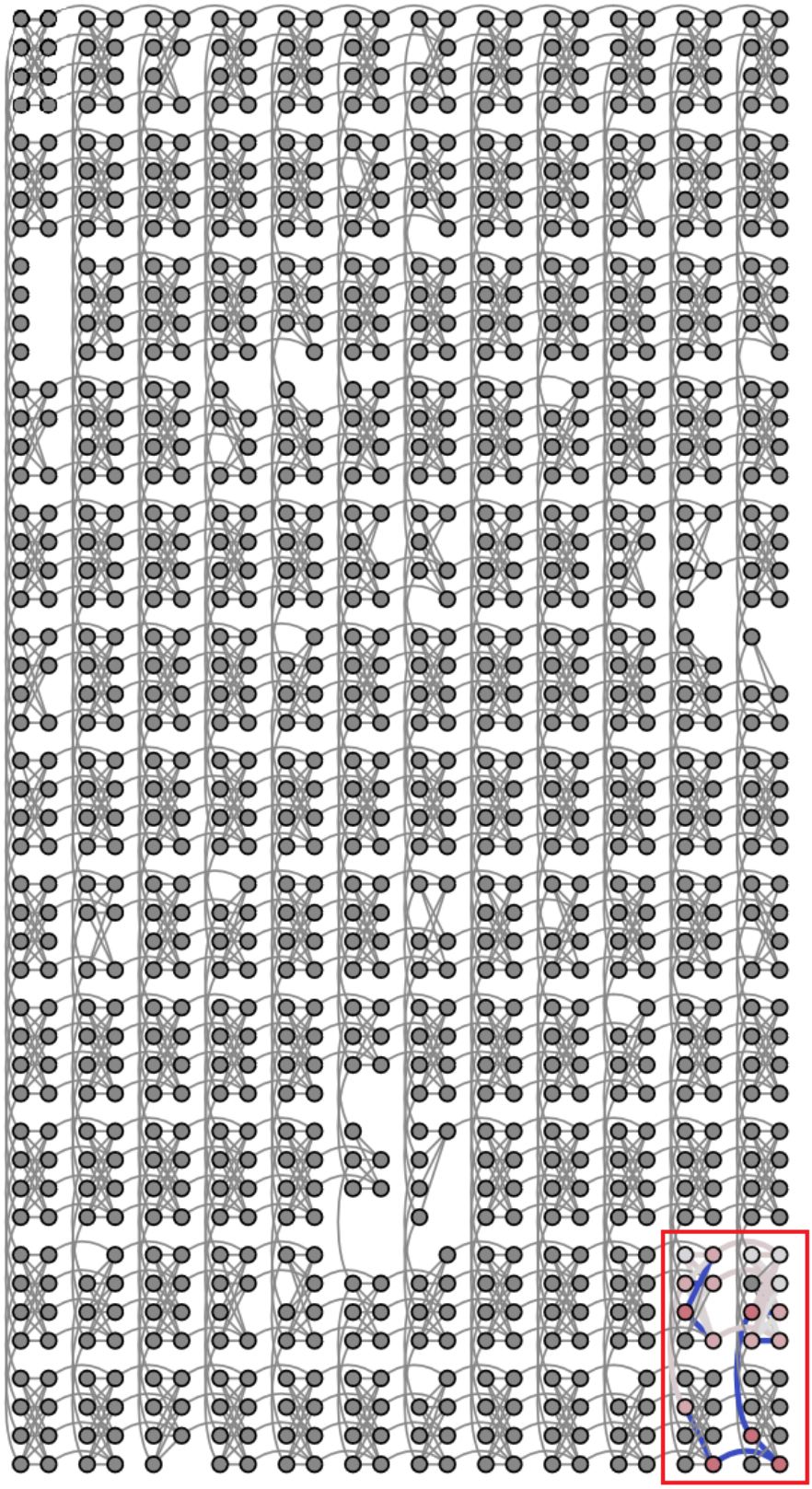

FIGURE 6.7 – Intégration de l'instance QUBO associée à G_1 sur le DW2X, la variable 0 étant intégré aux qubits $\{1040\}$, 1 à $\{1048\}$, 2 à $\{1053\}$, 3 à $\{1055, 1051\}$, 4 à $\{1041, 1045\}$, 5 à $\{1044, 1042, 1047\}$, 6 à $\{1137, 1143, 1054, 1151, 1050, 1146\}$ et 7 à $\{1052\}$.

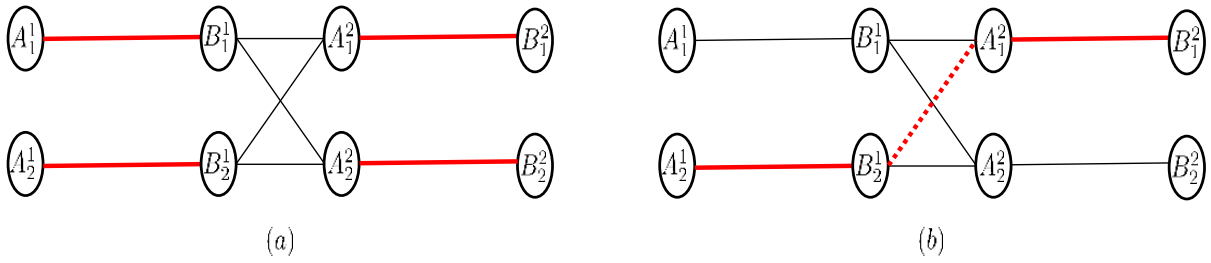

FIGURE 6.8 – Représentation graphique des solutions dans le tableau 6.3. Les lignes pointillées indiquent les mauvaises duplications.

Instance G_2 : Cette instance conduit à un graphe avec 18 sommets, 27 arêtes, puis à un QUBO avec 27 variables et 72 coefficients non nuls et non diagonaux. Mettre ce QUBO sur la machine D-Wave a nécessité 100 qubits comme le montre la figure 6.11. Sur 10 000 exécutions, la solution optimale n'a été obtenue que 662 fois (soit une probabilité de réussite de 6%). La figure 6.12 montre les résultats graphiques de la meilleure, médiane et pire solutions obtenues (respectivement avec un coût de -495, -389 et -89). Bien que la meilleure solution obtenue soit optimale, la solution médiane ne conduit pas à un couplage valide puisque quatre sommets sont couverts 3 fois[2] Comme pour G_1, nous observons également que les pires solutions présentent des problèmes de cohérence de duplication. La figure 6.13 montre l'histogramme de la fonction économique (mais renormalisée) pour les 10 000 exécutions de recuit que nous avons effectuées. En outre, comme certaines de ces solutions sont incohérentes en ce qui concerne la duplication, la figure 6.14 montre l'histogramme de la fonction économique pour les solutions dans lesquelles les incohérences de duplication ont été corrigées par un vote majoritaire (ce qui a entraîné un déplacement marginal vers la gauche du coût moyen de la solution de -402,9 à -404,8, la médiane étant inchangée).

Instance G_3 : Cette instance conduit à un graphe avec 32 sommets, 64 arêtes et un QUBO avec 64 variables et 240 coefficients non nuls et non diagonaux. Mettre ce QUBO sur la machine D-Wave a nécessité 431 qubits (39% des qubits disponibles) comme le montre la figure 6.15. Pour les 10 000 exécutions, la solution optimale n'a jamais été obtenue. Néanmoins, la figure 6.16 fournit

2. La fixation de cette solution nécessiterait une étape de post-traitement pour produire des couplages valides. En effet, cela n'est pas pertinent pour un problème polynomial, mais un tel post-traitement serait donc nécessaire lors de l'utilisation opérationnelle de la machine pour résoudre des problèmes non artificiels.

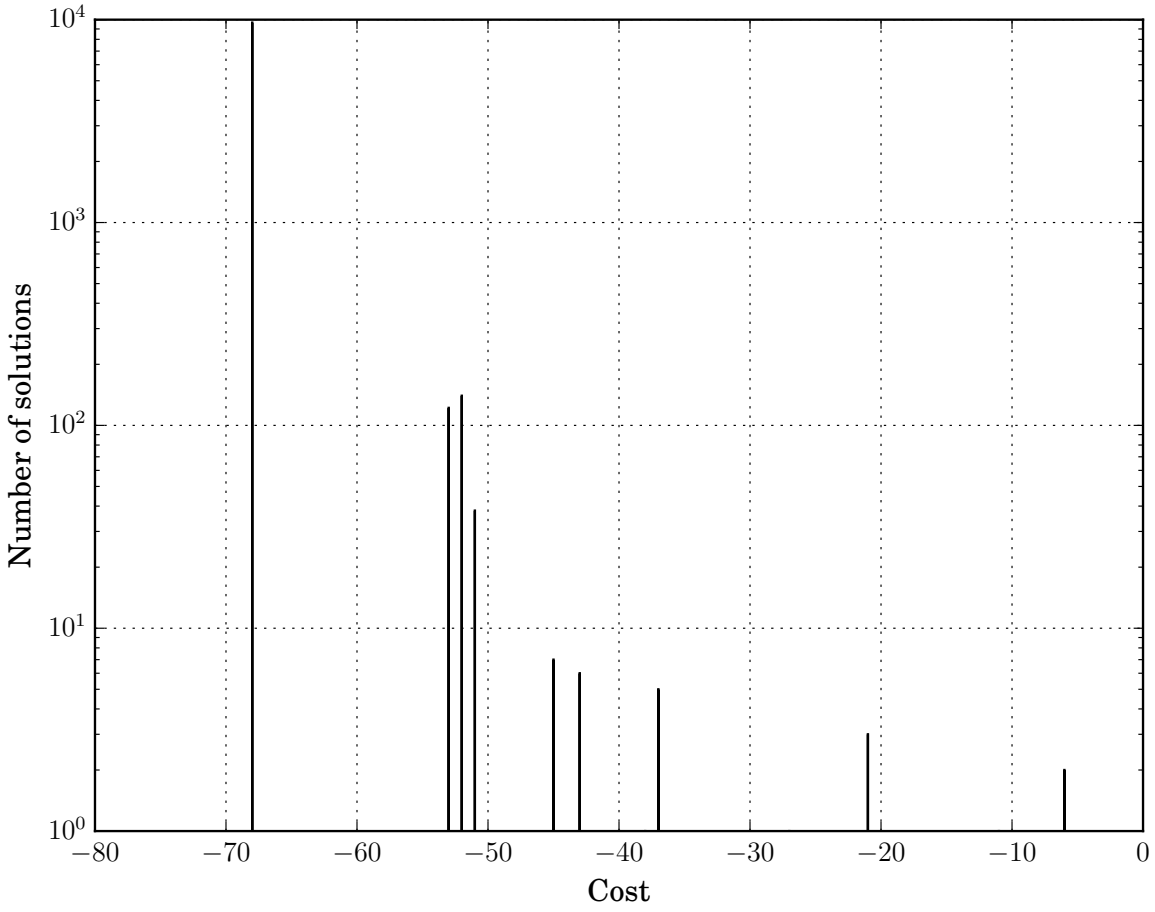

FIGURE 6.9 – Histogramme de la fonction économique sur 10 000 recuits pour G_1.

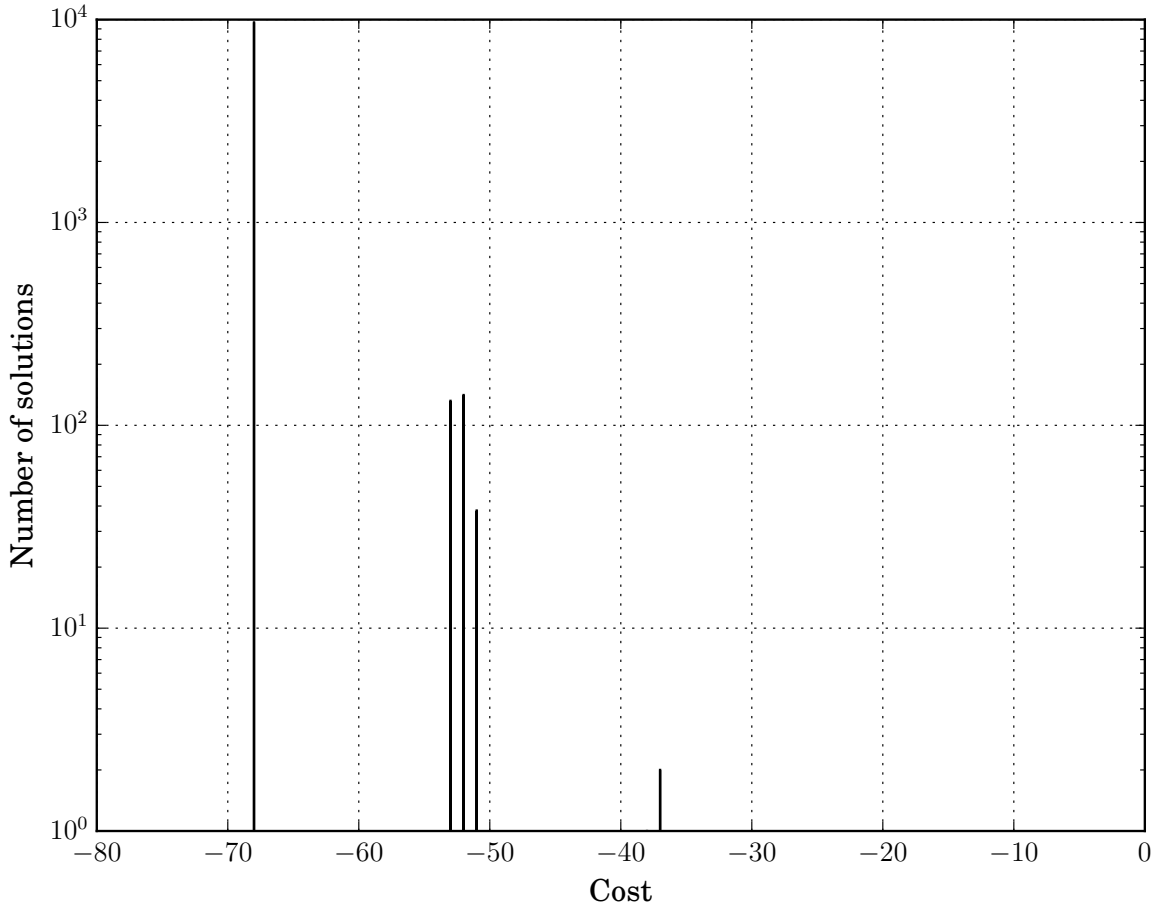

FIGURE 6.10 – Histogramme de la fonction économique sur 10 000 recuits pour G_1 (avec les mauvaises duplications fixées par le vote majoritaire).

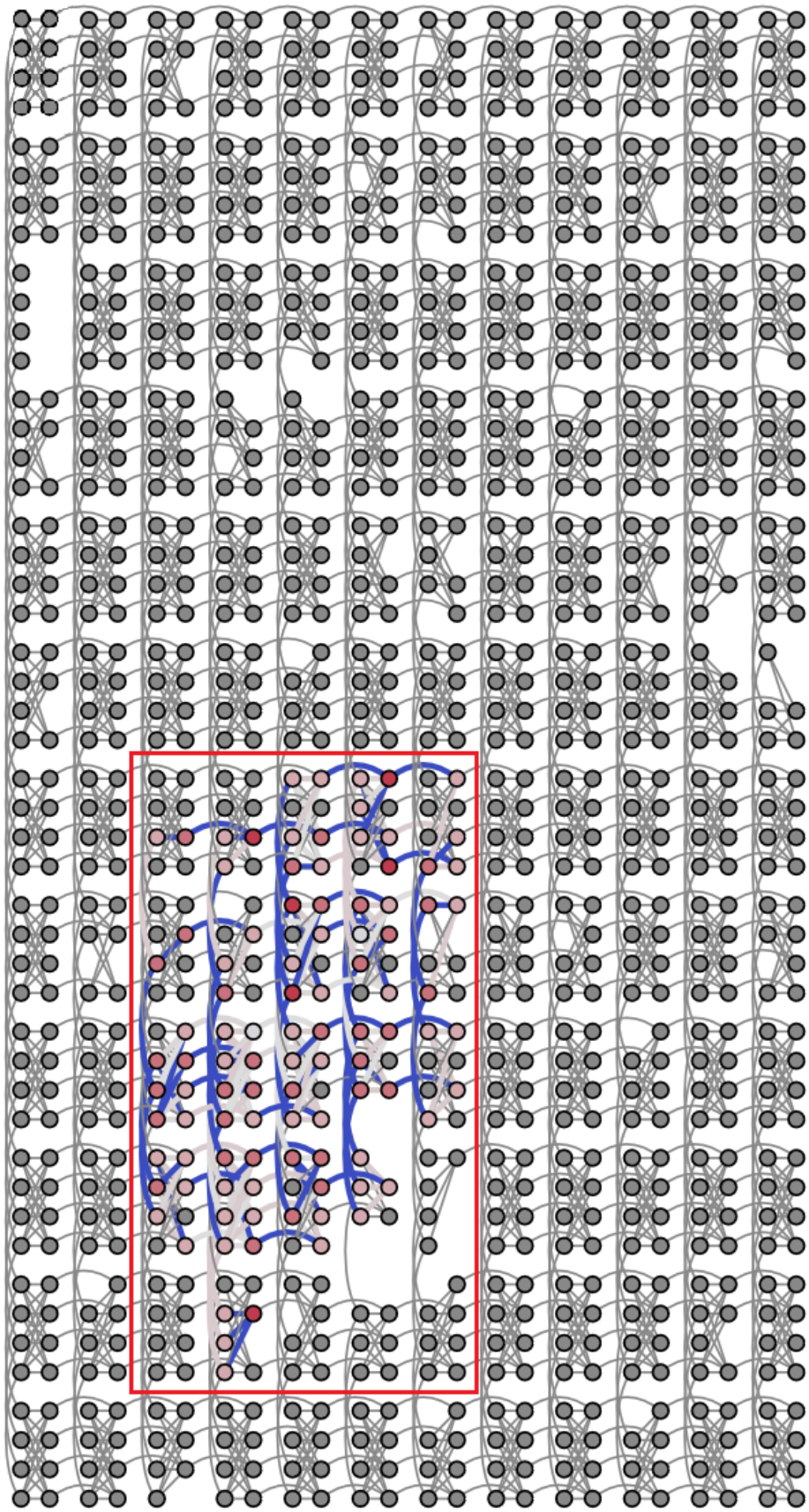

FIGURE 6.11 – Intégration de l'instance QUBO associée à G_2 sur le D-Wave 2X.

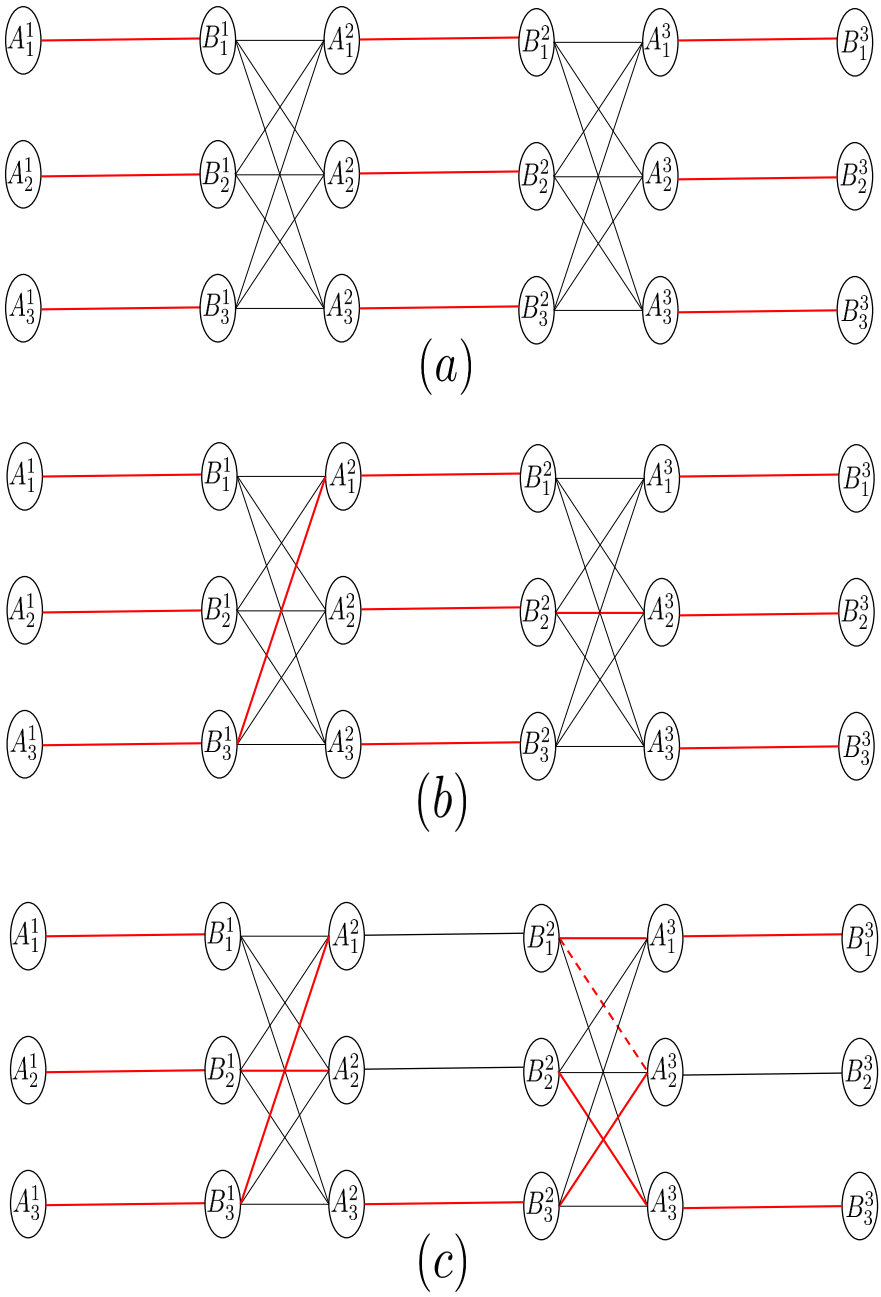

FIGURE 6.12 – Représentation graphique de la meilleure (a), de la médiane (b) et de la pire (c) solution obtenue pour G_2. Les lignes en pointillés représentent les mauvaises de duplication des qubits.

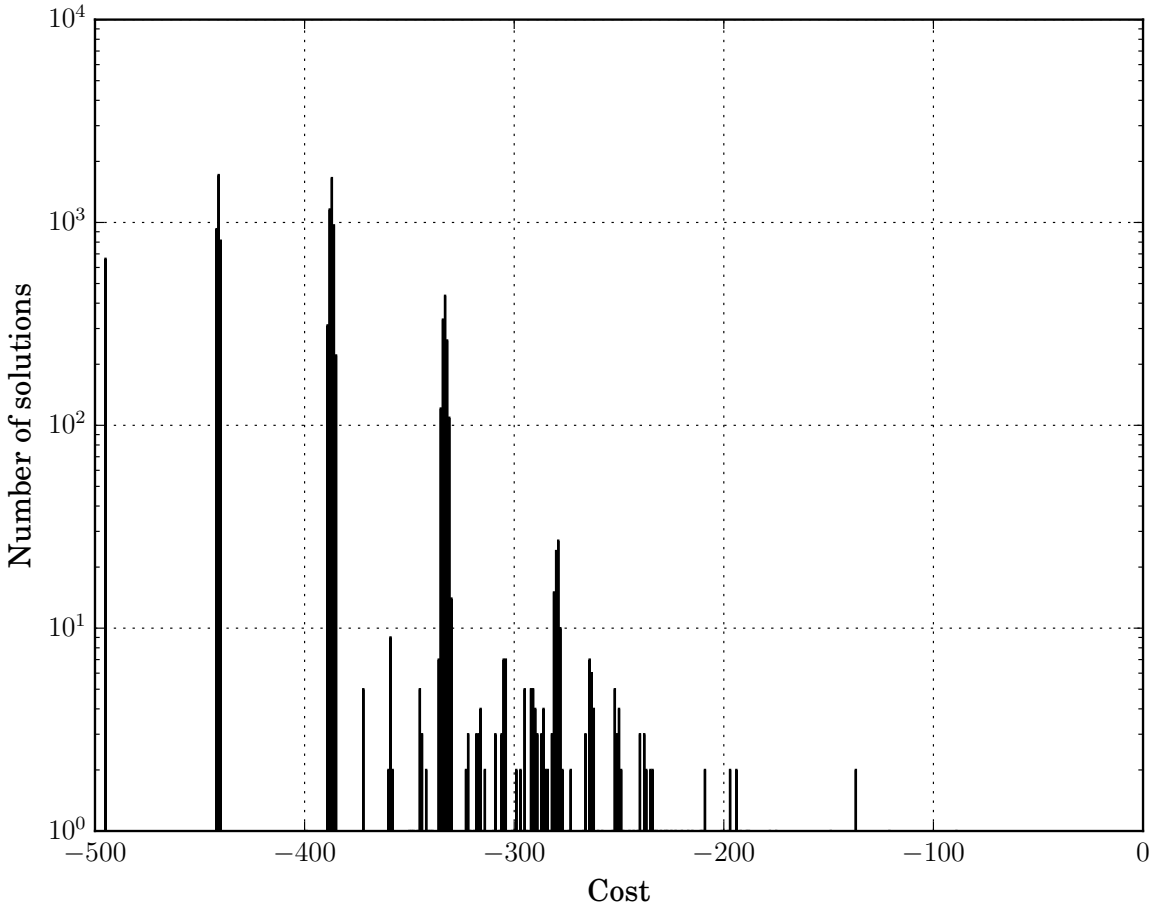

FIGURE 6.13 – Histogramme de la fonction économique sur 10 000 exécutions du recuit sur G_2.

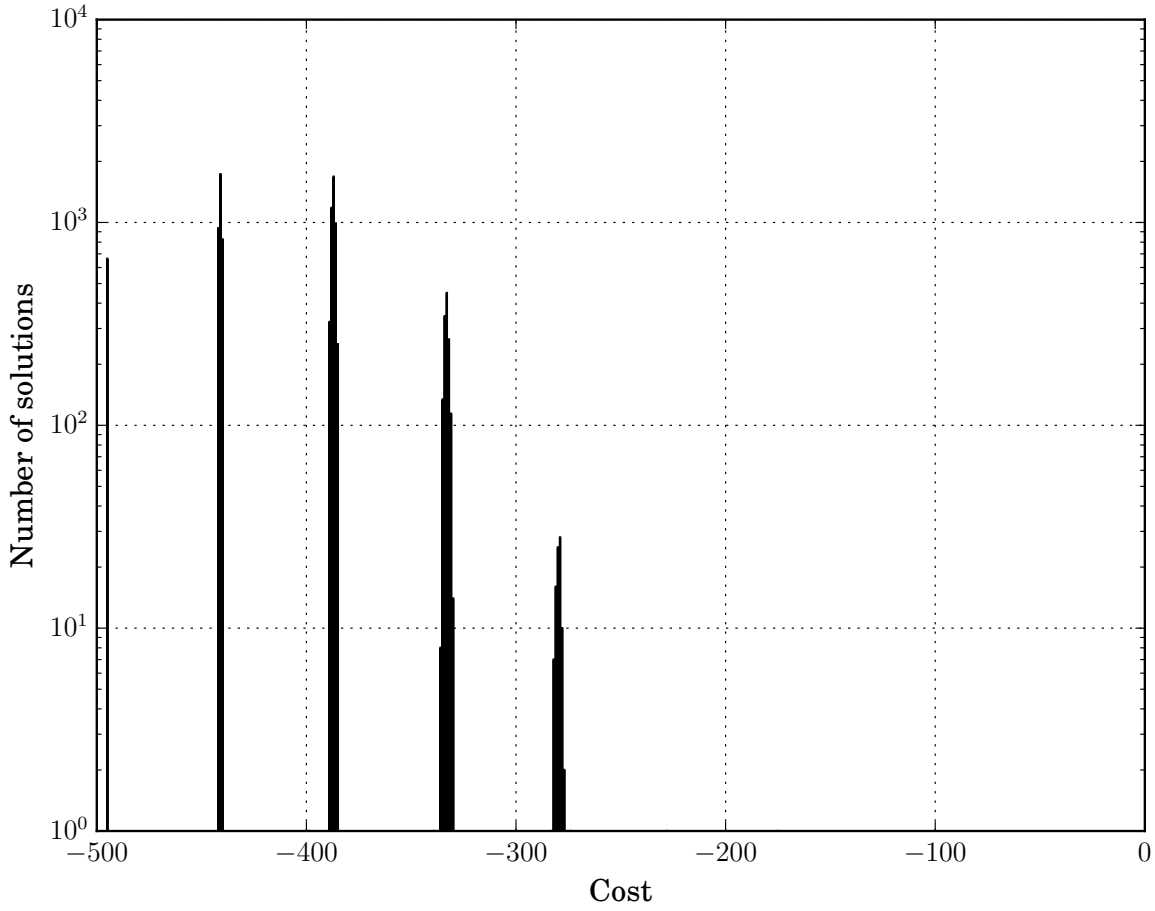

FIGURE 6.14 – Histogramme de la fonction économique sur 10 000 exécutions du recuit sur G_2 (avec les mauvaises duplications corrigées par le vote majoritaire).

des représentations graphiques de la meilleure, médiane et pire solution obtenue (respectivement avec un coût de -1809, -1551 et -549). Pour G_3, la valeur optimale est de -2064, et la meilleure solution obtenue est éloignée d'environ 15% de la solution optimale (le coût médian est de 25%). De plus, ni la meilleure solution ni la médiane ne permettent d'obtenir des couplages valides puisque dans les deux cas, certains sommets sont couverts plusieurs fois. On observe également que la pire solution présente des problèmes de cohérence de duplication. La figure 6.17 montre l'histogramme de la fonction économique renormalisée pour les 10 000 exécutions du recuit que nous avons effectuées. En outre, comme certaines de ces solutions sont incohérentes en ce qui concerne la duplication, la figure 6.18 montre l'histogramme de la fonction économique pour les solutions dans lesquelles les incohérences de duplication ont été corrigées par un vote majoritaire (ce qui fait passer le coût moyen de -1460,8 à -1491,8 et le coût médian de -1549 à -1550, ce qui reste assez marginal).

Instance G_4 : Cette instance conduit à un graphe avec 50 sommets, 125 arêtes et à un QUBO avec 125 variables et 600 coefficients non nuls et non diagonaux. Mettre ce QUBO sur la machine D-Wave a nécessité 951 qubits, comme le montre la figure 6.19 (comme indiqué précédemment, cela représente environ 87% des qubits disponibles pour cette machine D-Wave). Pour 10 000 exécutions, la solution optimale n'a jamais été obtenue. Néanmoins, la figure 6.31 fournit des représentations graphiques de la meilleure, médiane et pire solution obtenue (respectivement avec un coût de -5524, -4526 et -2109). Pour G_4, la valeur optimale est de -6075, donc la meilleure solution obtenue est éloignée d'environ 10% de la solution optimale (un meilleur rapport que pour G_3) et le coût médian de 25%. En outre, ni la meilleure solution ni la médiane ne permettent d'obtenir de couplages valides puisque dans les deux cas, certains sommets sont couverts plusieurs fois. Nous observons également que la pire solution présente des problèmes de cohérence de duplication. La figure 6.21 montre l'histogramme de la fonction économique renormalisée pour les 10 000 exécutions du recuit que nous avons effectuées. En outre, comme certaines de ces solutions sont incohérentes en ce qui concerne la duplication, la figure 6.22 montre l'histogramme de la fonction économique pour les solutions dans lesquelles les incohérences de duplication ont été fixées par un vote majoritaire (ce qui a permis de faire passer le coût moyen de la solution de -4492,4 à -4525,8 et le coût médian de -4525 à -4526, ce qui est également marginal).

6.3.3 Résultats avec inversion de spins

Afin d'améliorer la qualité des résultats obtenus dans nos expériences, nous avons utilisé une inversion de jauges (transformations de spin). Le principe d'une inversion de jauge est d'appliquer une inversion booléenne aux opérateurs σ_i dans notre hamiltonien (en termes de

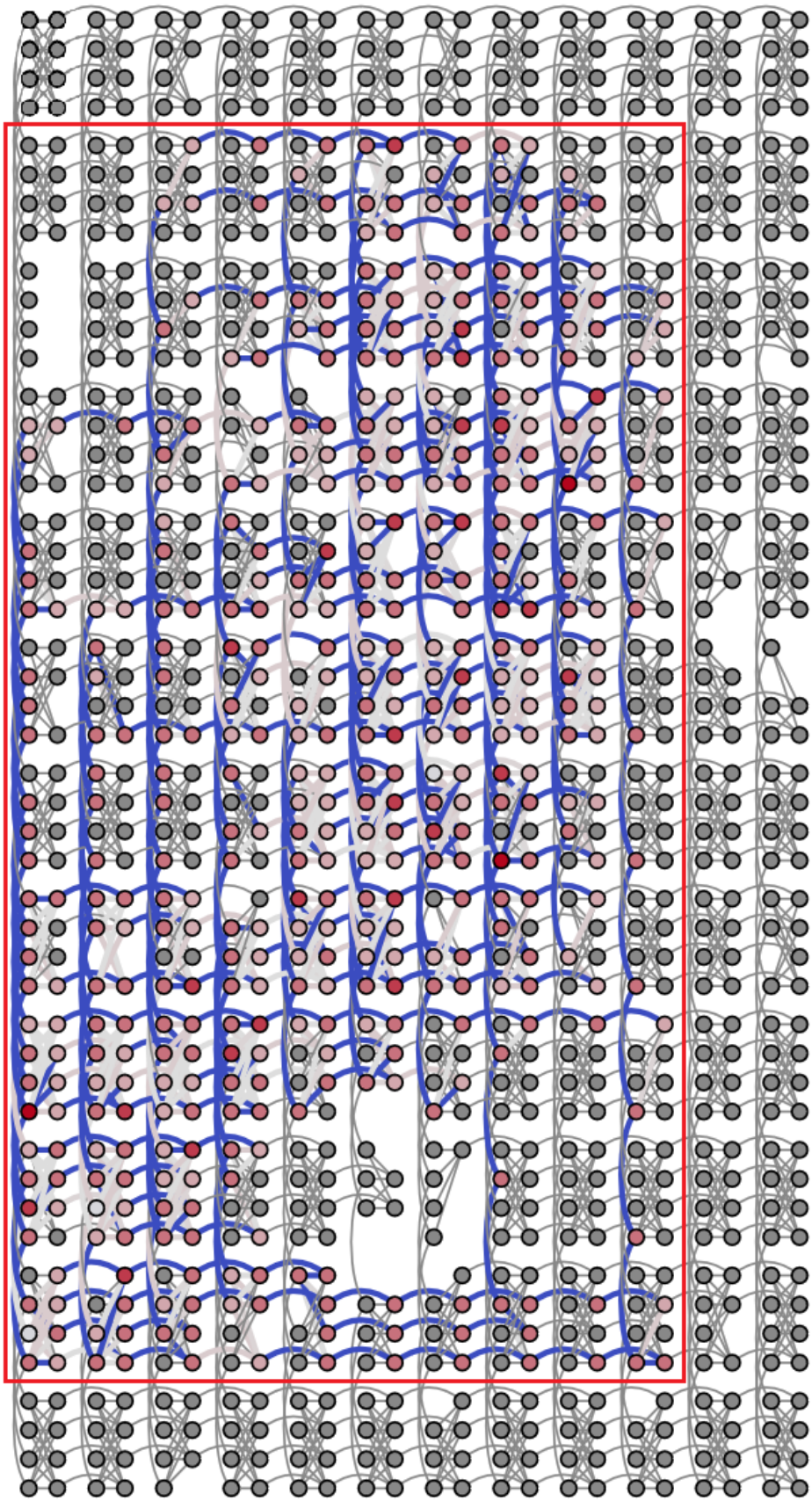

FIGURE 6.15 – Intégration de l'instance QUBO associée à G_3 sur le D-Wave 2X.

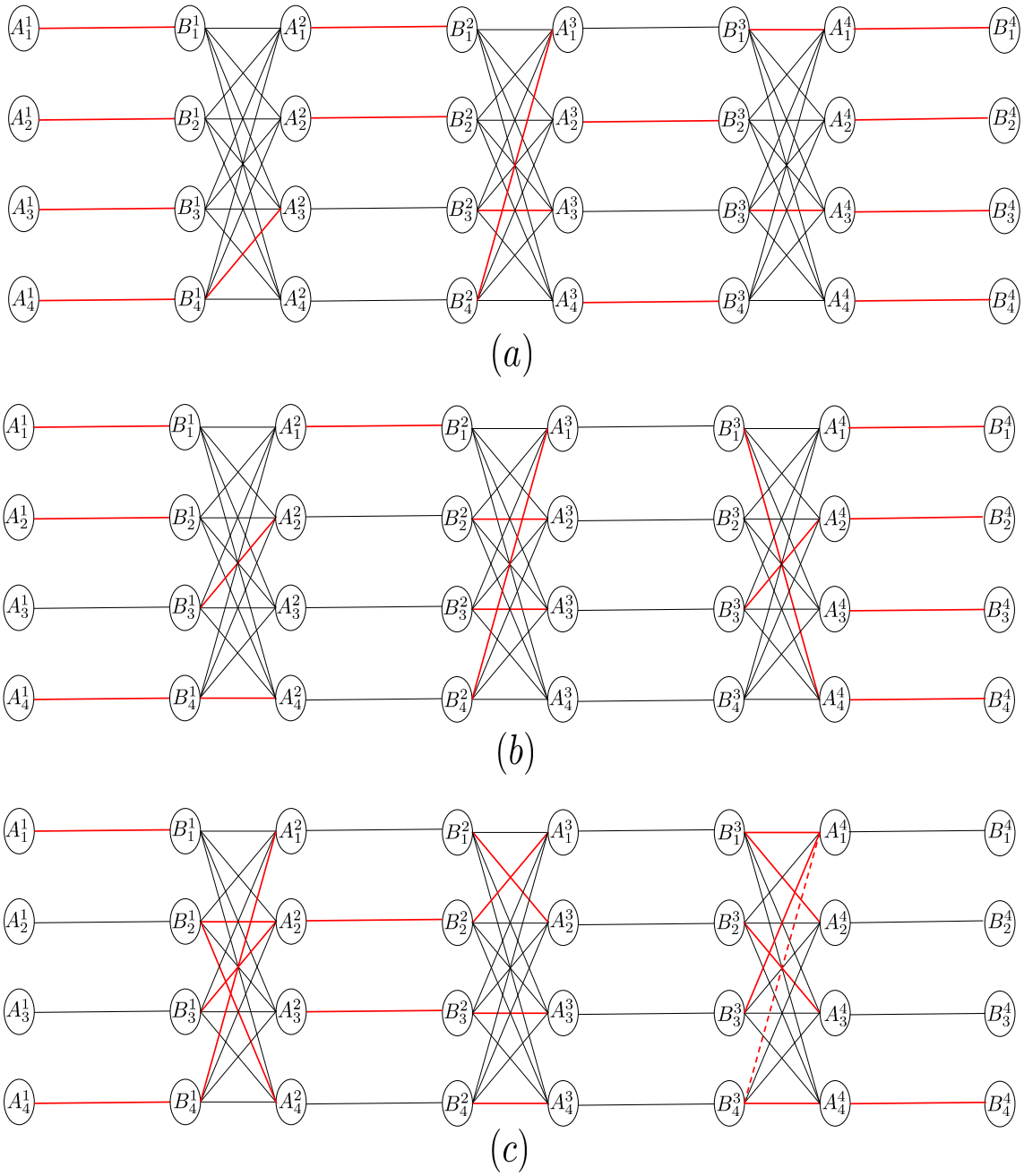

FIGURE 6.16 – Représentation graphique de la meilleure (a), de la médiane (b) et de la pire (c) solution obtenue pour G_3. Les lignes en pointillés représentent les mauvaises de duplication des qubits.

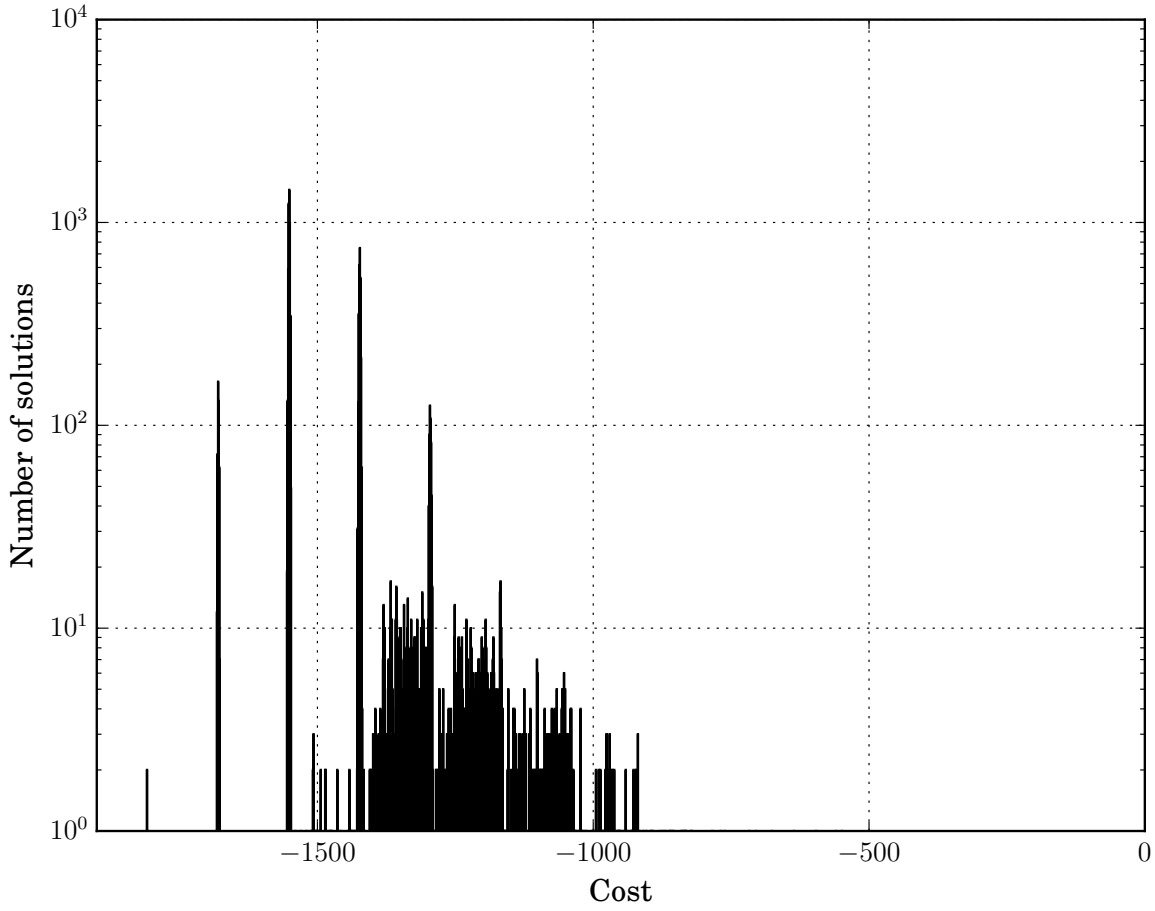

FIGURE 6.17 – Histogramme de la fonction économique sur 10 000 exécutions du recuit sur G_3.

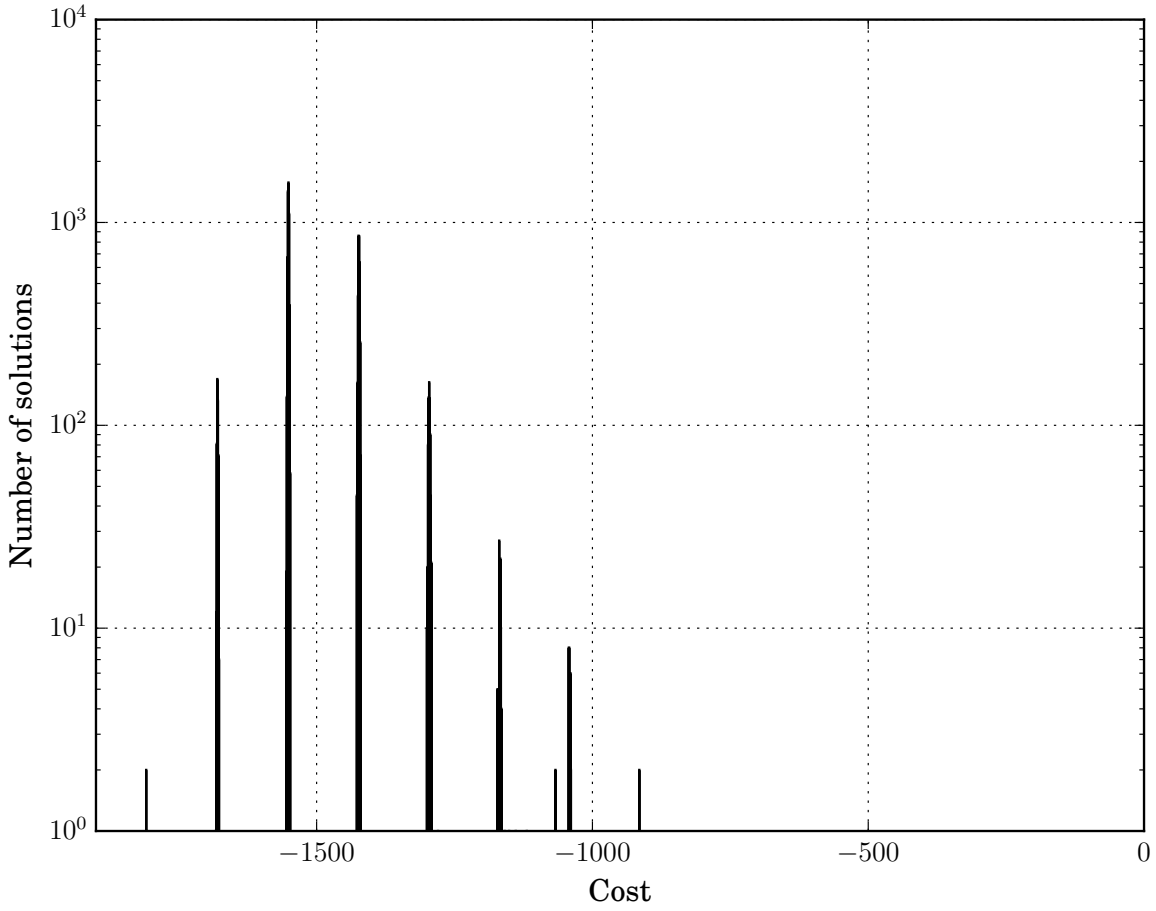

FIGURE 6.18 – Histogramme de la fonction économique sur 10 000 exécutions du recuit sur G_3 (avec des mauvaises duplications corrigées par le vote majoritaire).

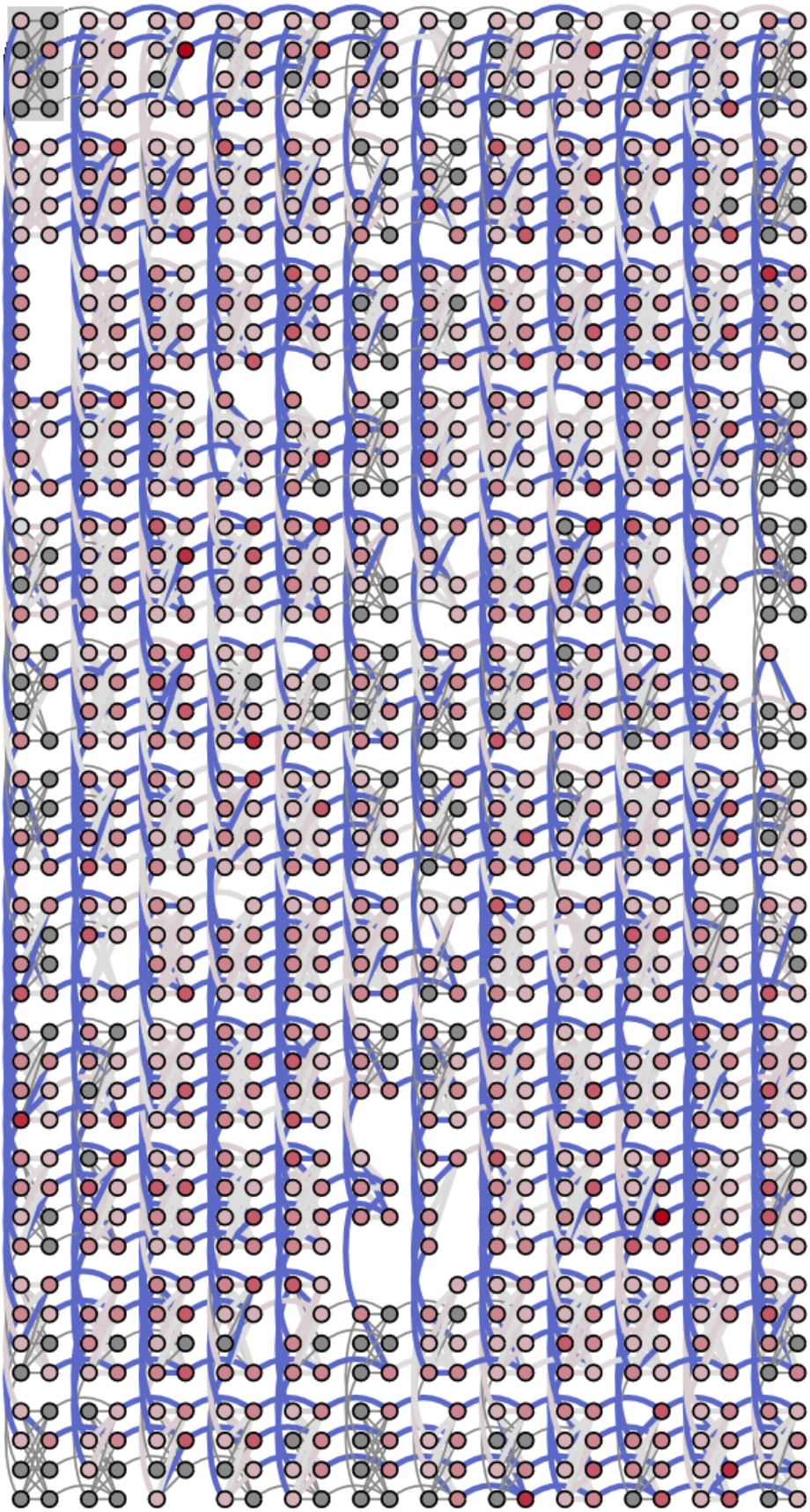

FIGURE 6.19 – Intégration de l'instance QUBO associée à G_4 sur le D-Wave 2X.

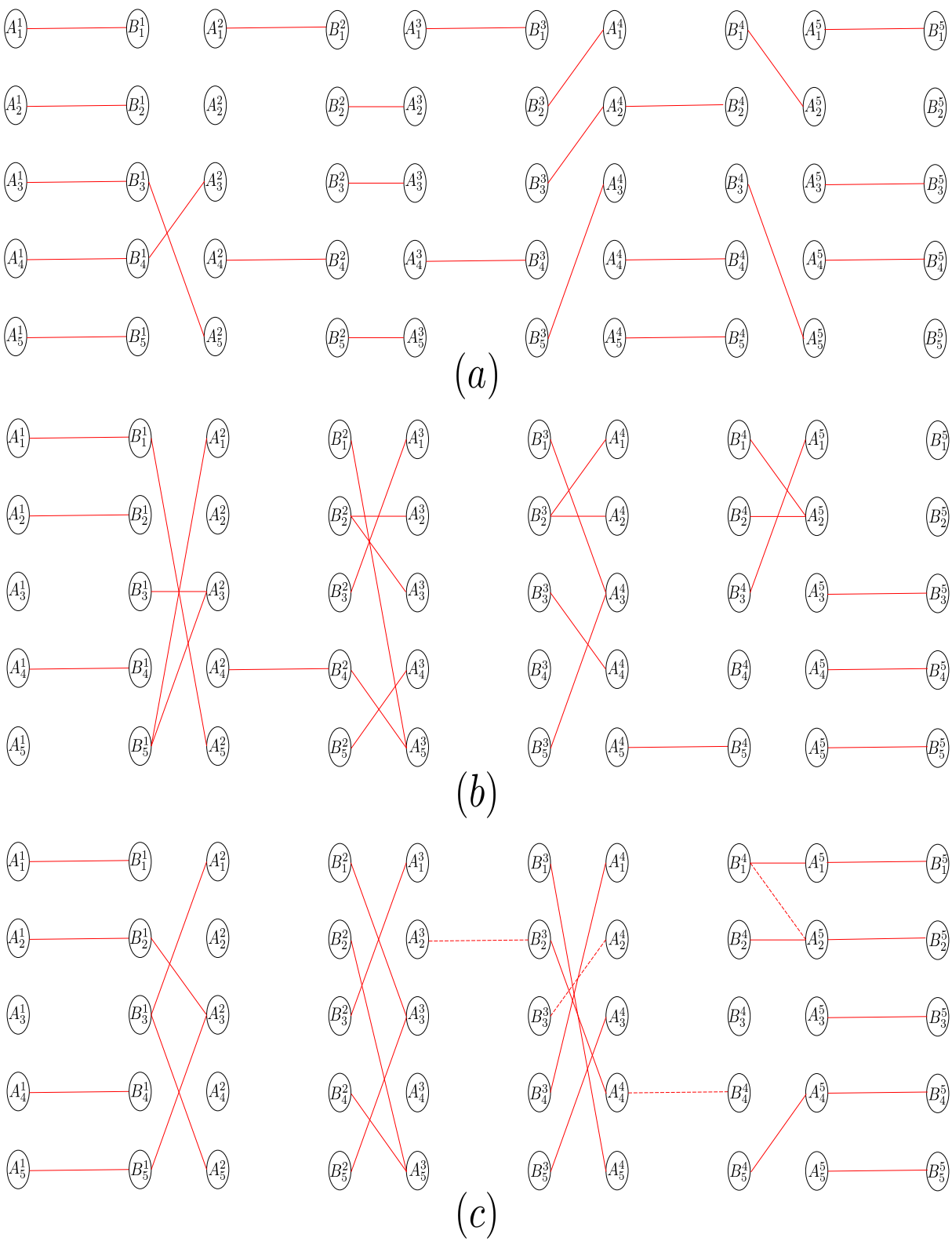

FIGURE 6.20 – Représentation graphique de la meilleure (a), de la médiane (b) et de la pire (c) solution obtenue pour G_4. Les lignes en pointillés représentent les mauvaises duplication des qubits.

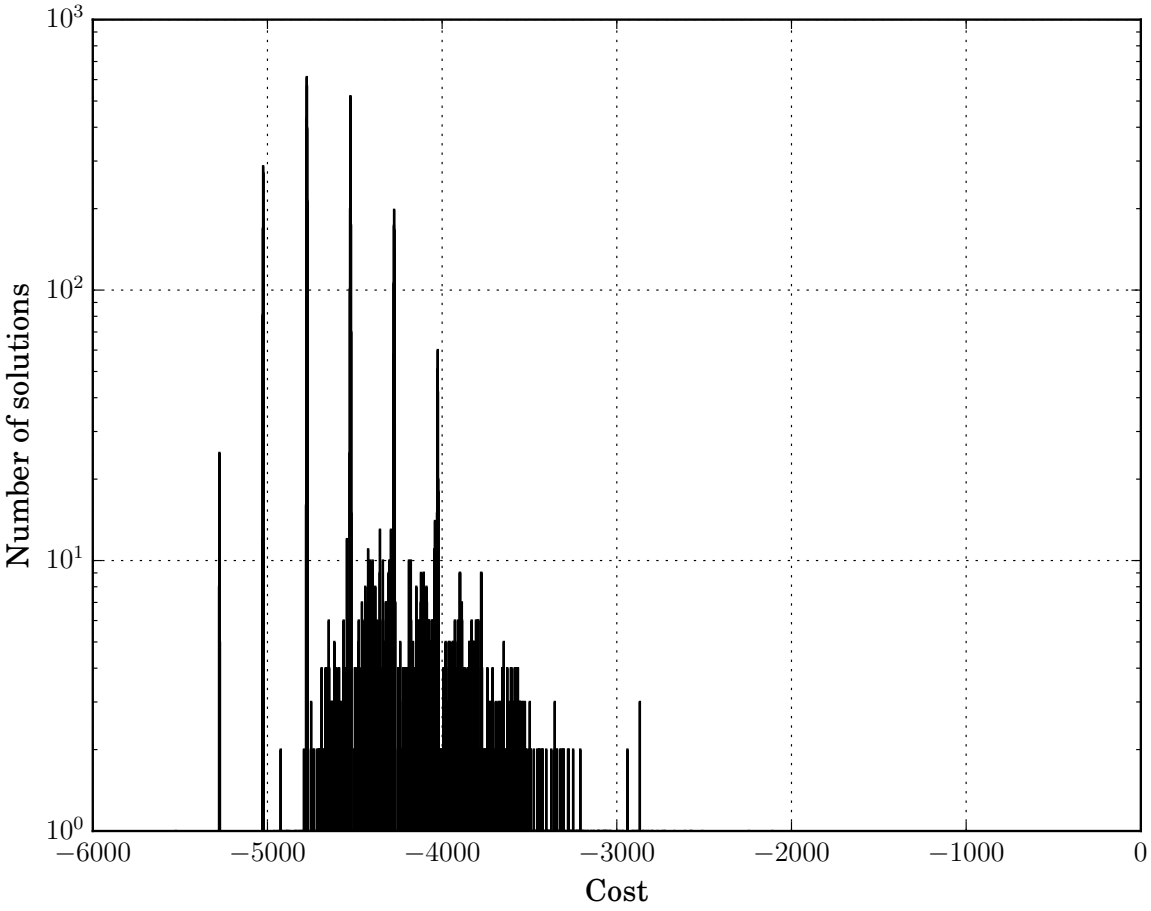

FIGURE 6.21 – Histogramme de la fonction économique sur 10 000 exécutions du recuit sur G_4.

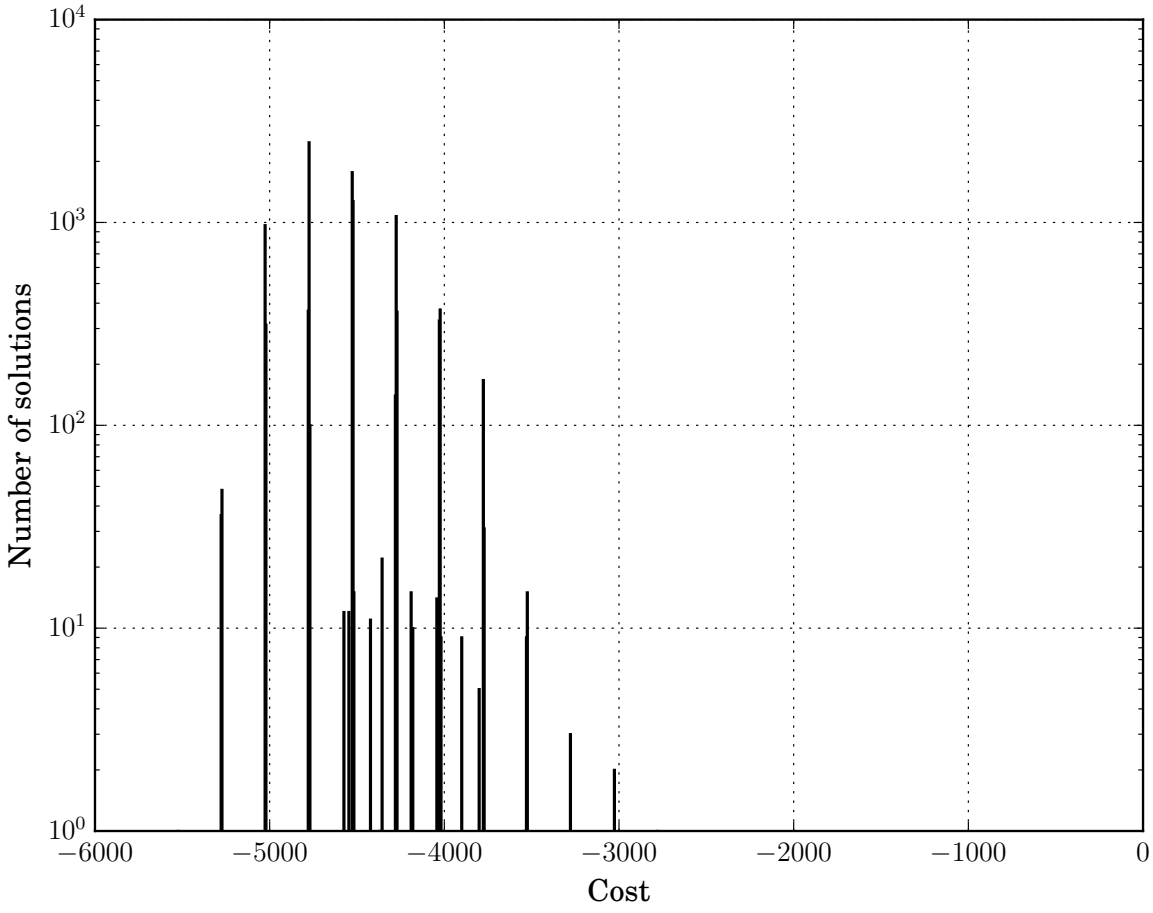

FIGURE 6.22 – Histogramme de la fonction économique sur 10 000 exécutions du recuit sur G_4 (avec des mauvaises duplications corrigées par le vote majoritaire).

QUBO, après duplication des qubits, cela signifie inverser une certaine variable x_i par $1 - y_i$, avec $y_i = 1 - x_i$). Cette transformation a la particularité de ne pas modifier la solution optimale du problème et de limiter l'effet des biais locaux des qubits, ainsi que les erreurs de précision machine [33]. En suivant la procédure couramment utilisée (par exemple [10]), nous avons sélectionné au hasard 10% des qubits physiques utilisés comme inversion de spins pour chaque instance G_n que nous avons intégrée sur le D-Wave. Les résultats sont donnés dans le tableau 6.4 et commentés dans la section suivante.

		sans vote					avec vote				
	opt.	best	worst	mean	median	stdev	best	worst	mean	median	stdev
G_1	-68	-68	-9	-66.8	-68	4.6	-68	-37	-66.8	-68	4.2
G_2	-495	-495	-29	-398.2	-388	48.1	-495	-277	-400.4	-388	44.6
G_3	-2064	-1810	-505	-1454.8	-1548	157.7	-1810	-911	-1496.5	-1550	111.8
G_4	-6275	-5527	-2507	-4609.9	-4675	346.5	-5527	-3030	-4579.2	-4527	314.1

TABLE 6.4 – Résumé des résultats expérimentaux sans (gauche) et avec (droite) vote majoritaire pour corriger les problèmes de duplication du qubit sur G_1, G_2, G_3, G_4.

6.3.4 Comparaison avec et sans inversion de spins

Dans cette section, nous comparons les résultats sur 10 000 exécutions obtenus avec le D-Wave pour nos instances G_1, G_2, G_3, G_4 avec et sans inversions de spins.

Instance G_1 : La solution optimale (avec un coût de -68) a été obtenue 9265 fois avec l'inversion de spin (et 9284 fois avec le vote majoritaire) contre 9673 sans l'inversion de spin, soit aucune nette amélioration entre les deux méthodes. De même la pire solution viole les contraintes de duplication avec un coût de -9 avec l'inversion contre -6 sans l'inversion

Instance G_2 : La solution optimale (avec un coût de -495) n'a été obtenue que 510 fois (soit une probabilité de 6%) avec l'inversion de spin contre 662 fois sans l'inversion de spin (aucun avantage significatif entre les deux méthodes). La solution médiane, avec un coût de -388 pour l'inversion et -389 sans l'inversion ne conduit pas à un couplage valide dans les deux cas. Pour la pire solution, l'utilisation de l'inversion de spin améliore la qualité de la solution, -277 avec et -89 sans (soit une amélioration de 37%).

Instance G_3 : La solution optimale n'a jamais été obtenue avec les deux méthodes. Pour G_3, la meilleure solution obtenue avec inversion de spins est de -1810 contre -1809 sans l'inversion

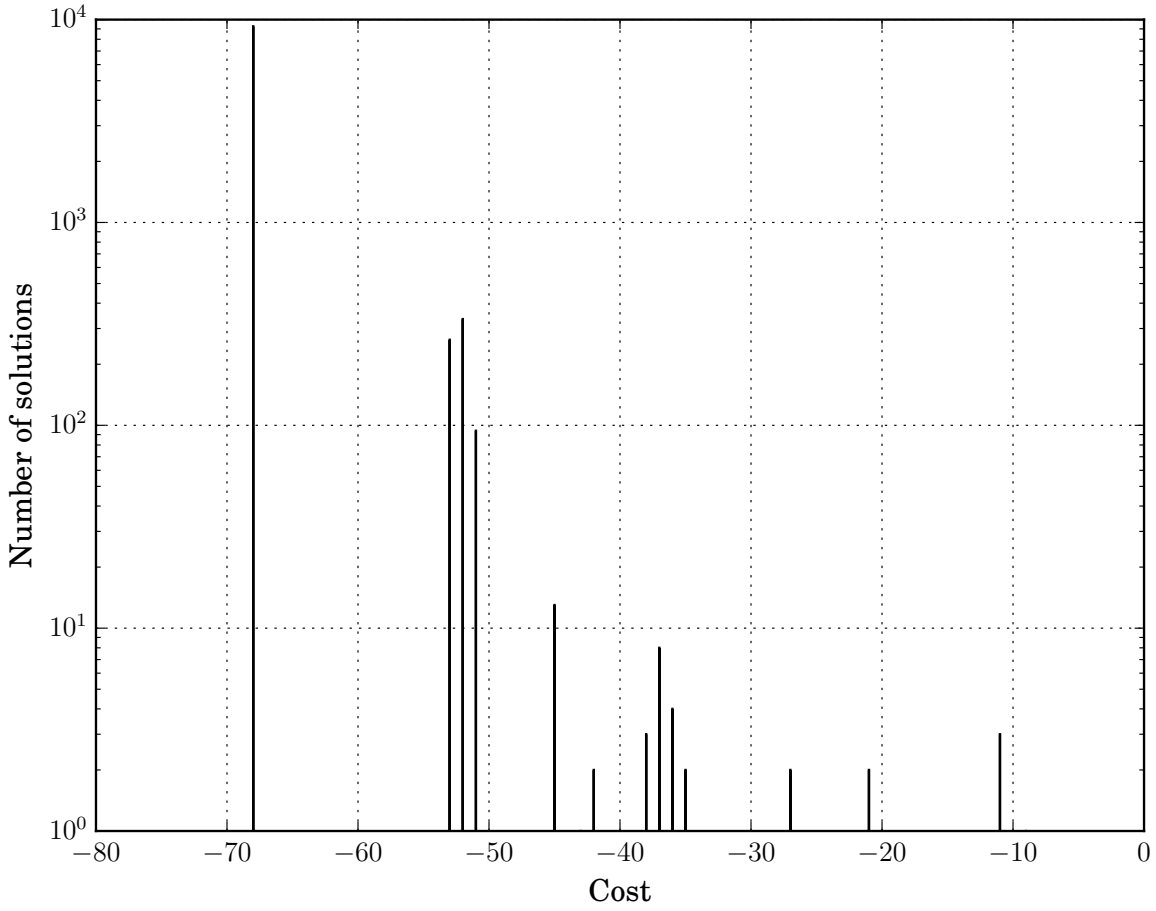

FIGURE 6.23 – Histogramme de la fonction économique sur 10 000 exécutions du recuit sur G_1.

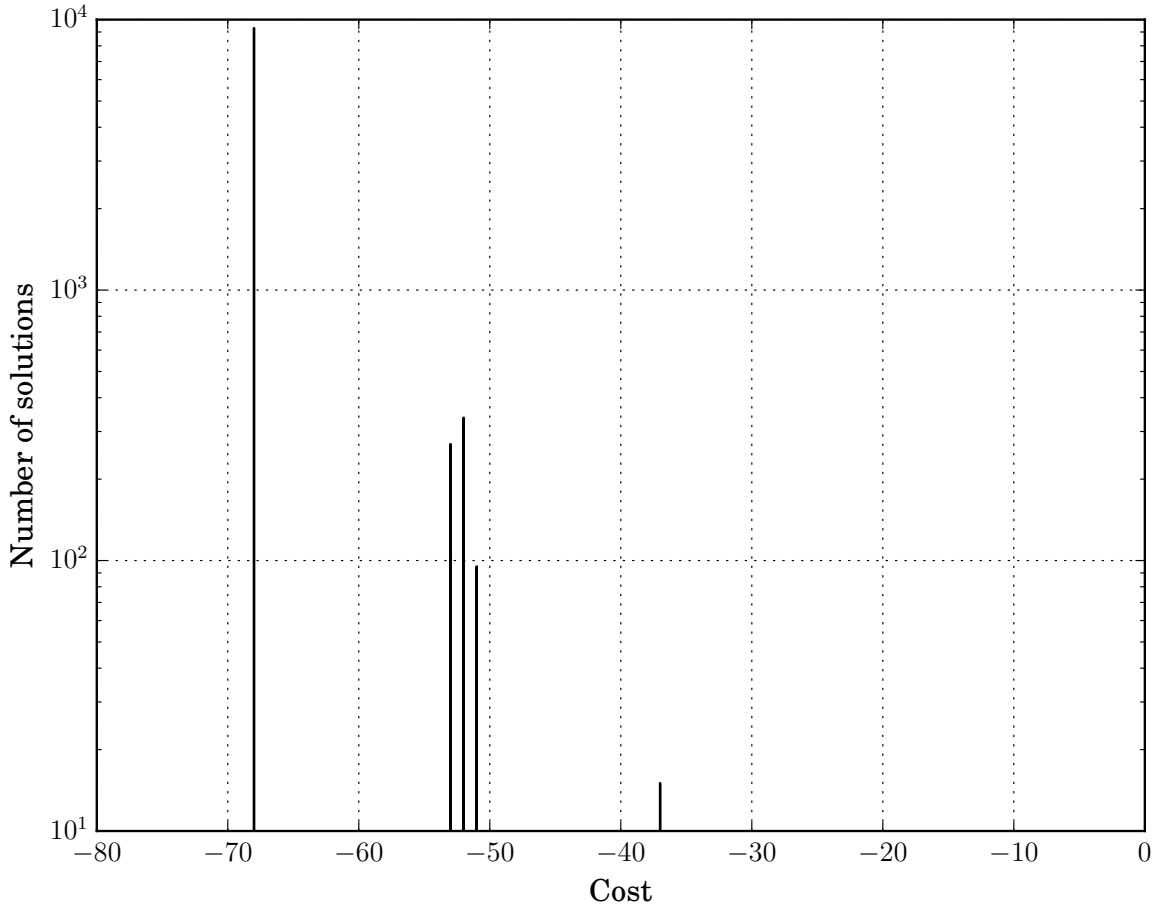

FIGURE 6.24 – Histogramme de la fonction économique sur 10 000 exécutions du recuit sur G_1 (avec des mauvaises duplications corrigées par le vote majoritaire).

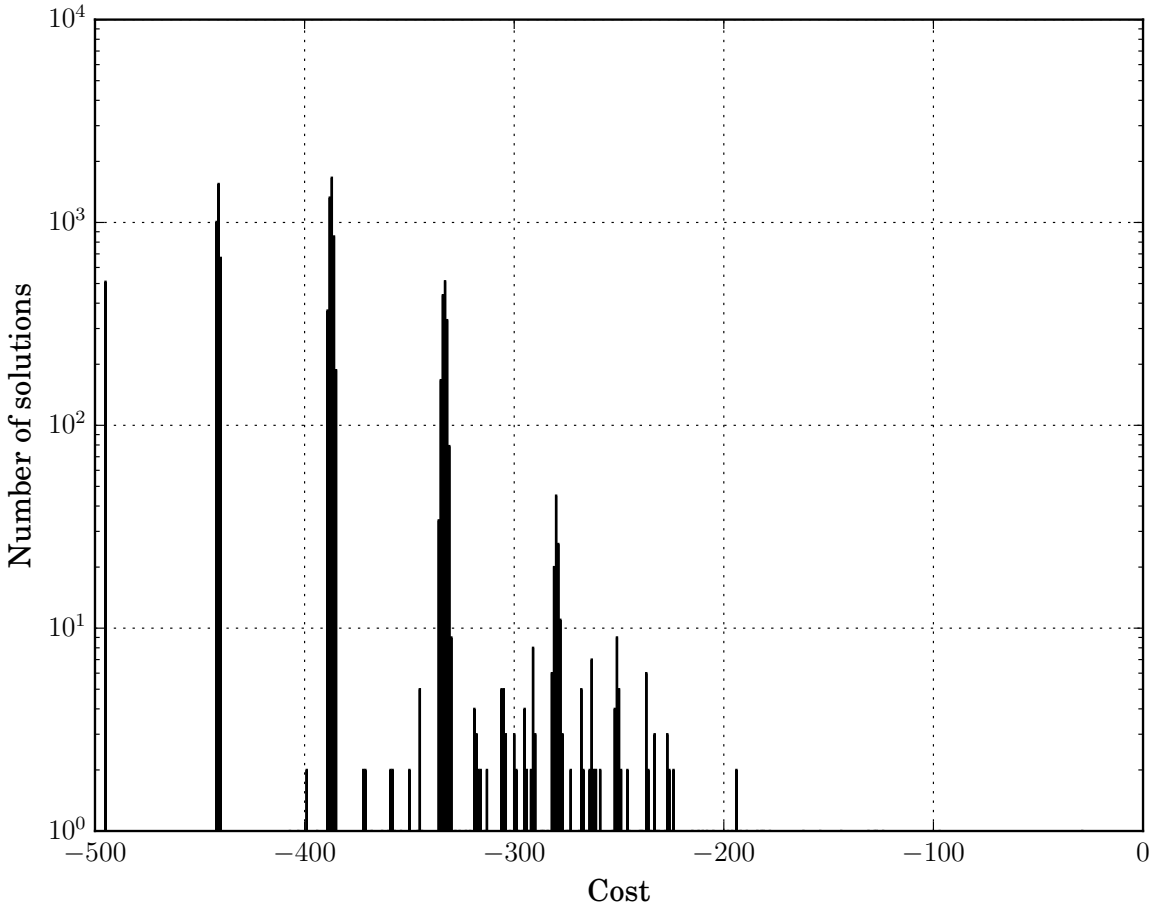

FIGURE 6.25 – Histogramme de la fonction économique sur 10 000 exécutions du recuit sur G_2.

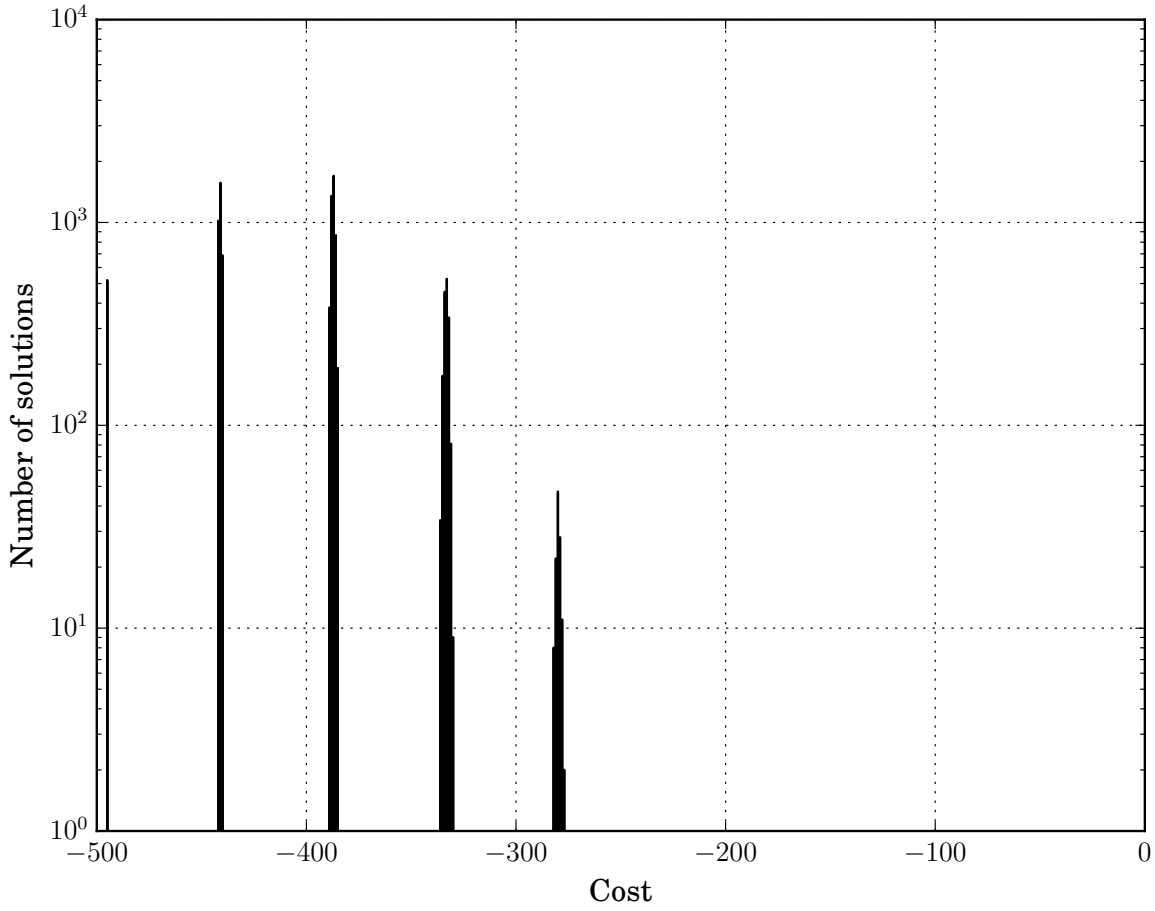

FIGURE 6.26 – Histogramme de la fonction économique sur 10 000 exécutions du recuit sur G_2 (avec des mauvaises duplications corrigées par le vote majoritaire).

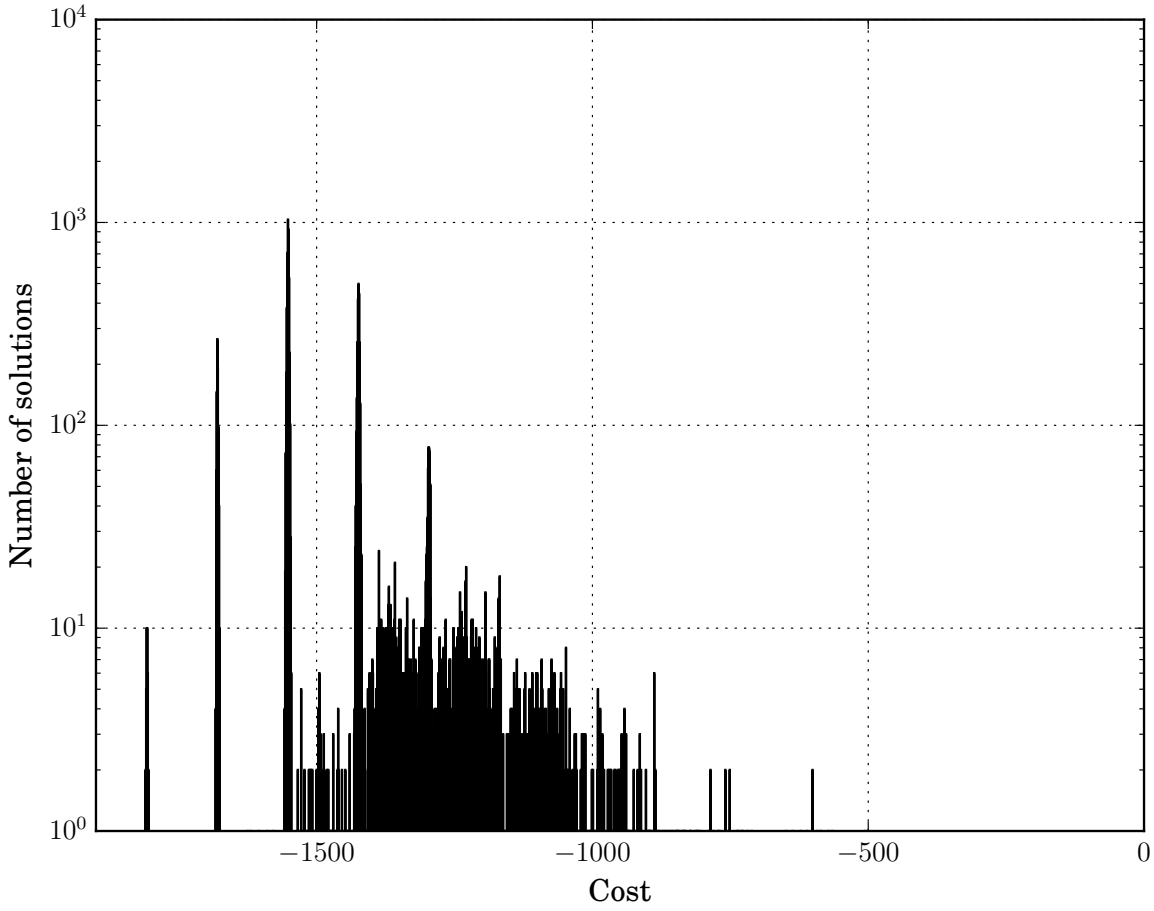

FIGURE 6.27 – Histogramme de la fonction économique sur 10 000 exécutions du recuit sur G_3.

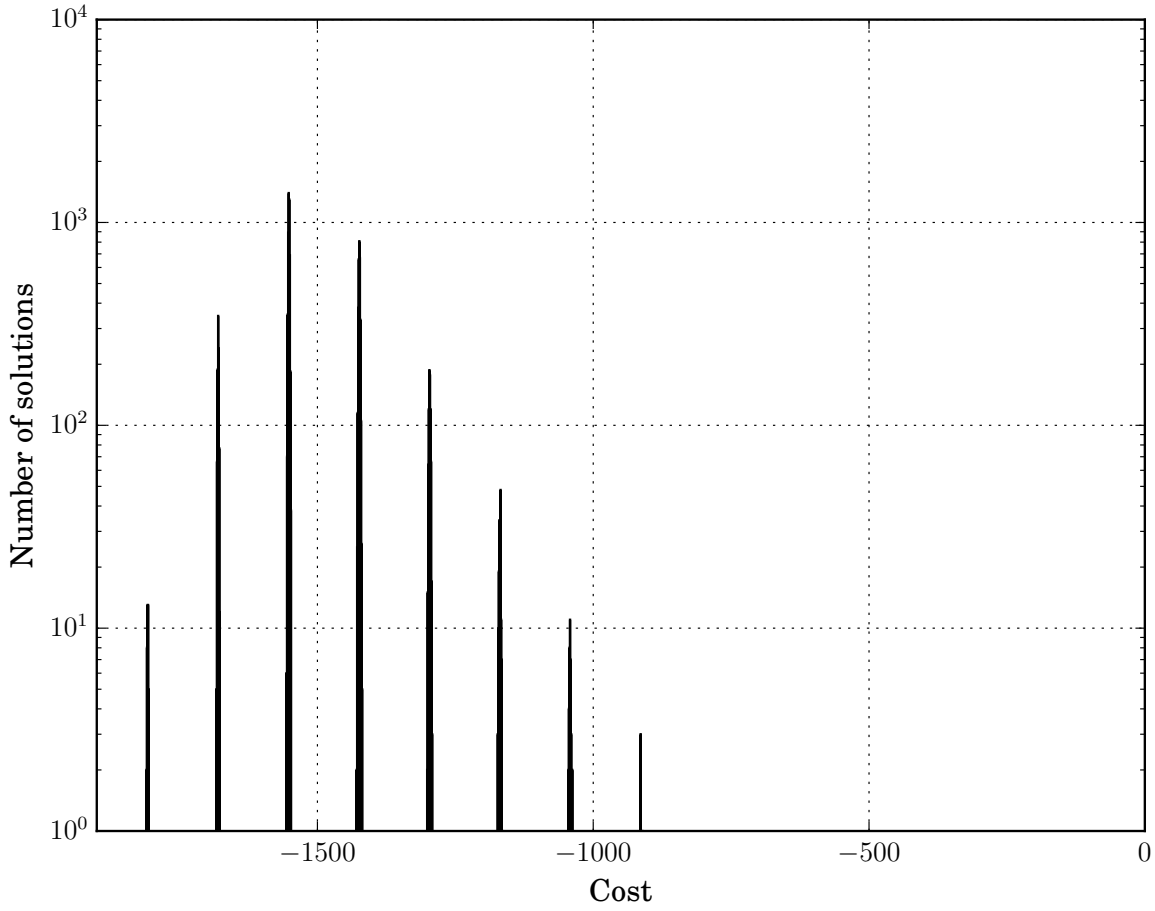

FIGURE 6.28 – Histogramme de la fonction économique sur 10 000 exécutions du recuit sur G_3 (avec des mauvaises duplications corrigées par le vote majoritaire).

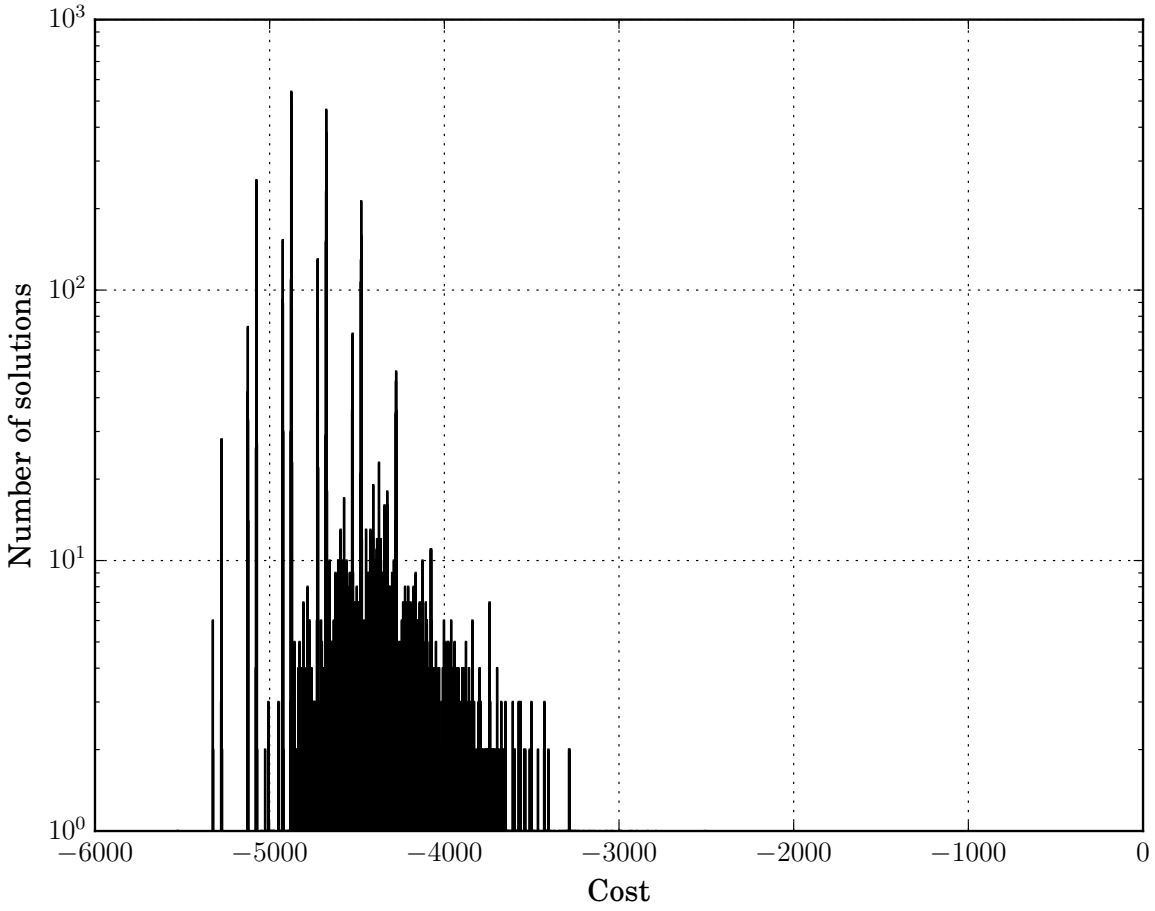

FIGURE 6.29 – Histogramme de la fonction économique sur 10 000 exécutions du recuit sur G_4.

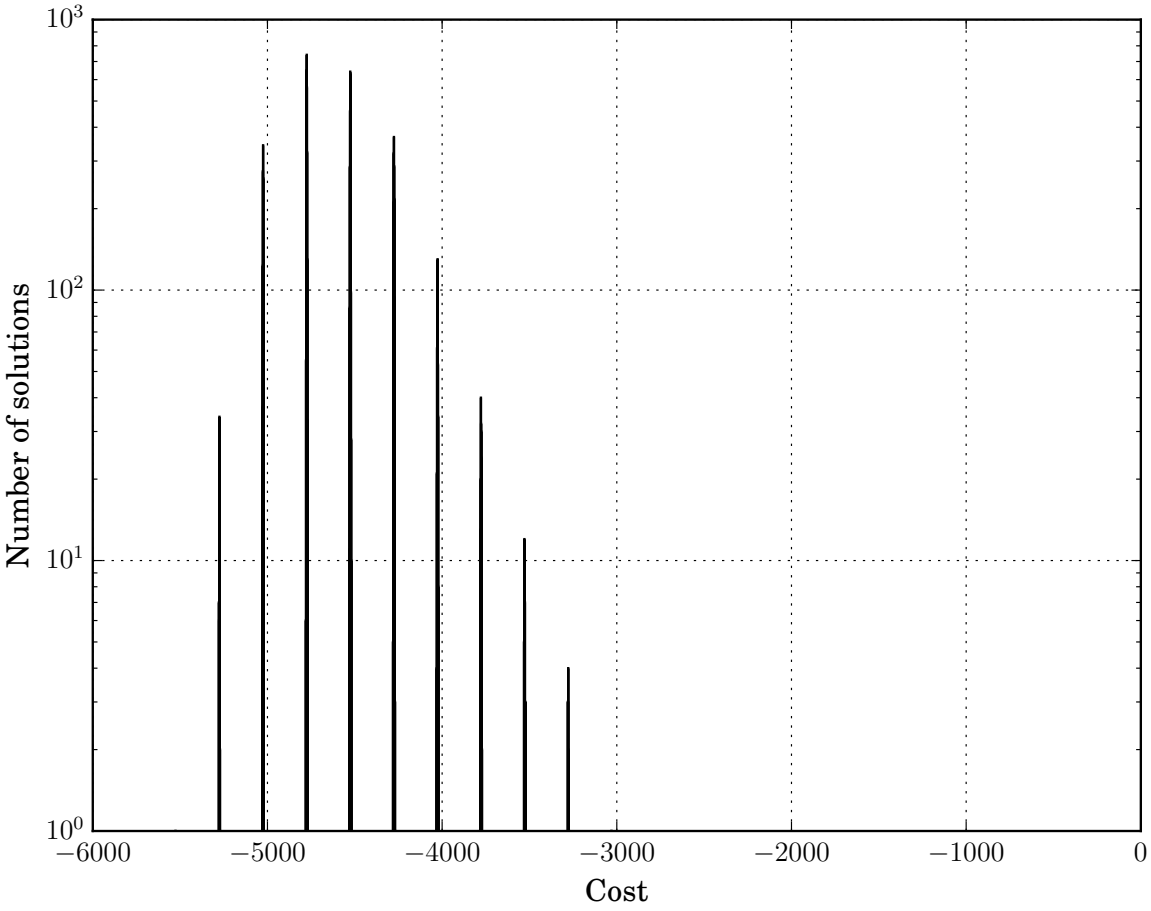

FIGURE 6.30 – Histogramme de la fonction économique sur 10 000 exécutions du recuit sur G_4 (avec des mauvaises duplications corrigées par le vote majoritaire).

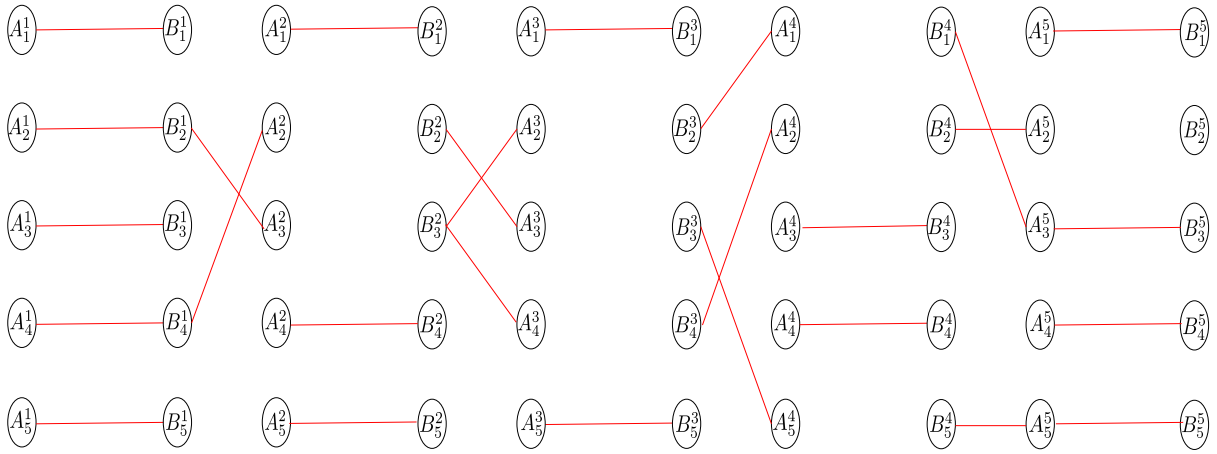

FIGURE 6.31 – Représentation graphique de la meilleure solution obtenue pour G_4 avec l'inversion de spins

(écart d'environ 15% avec la valeur optimale). La valeur médiane est identique dans les deux cas (-1548 avec inversion contre -1551 sans inversion) et la pire solution -505 contre -549 (soit un écart de l'ordre de 2% entre les pires solutions par rapport à la solution optimale).

Instance G_4 : La solution optimale n'a jamais été obtenue avec les deux méthodes. Le coût de la meilleure solution obtenue avec l'inversion de spins est de -5527 (la médiane et la pire solution obtenues sont respectivement de -4675 et -2507) contre -5524 sans l'inversion de spin (la médiane et la pire solution obtenues sont respectivement de -4526 et -2109). Pour G_4 avec et l'inversion de spins, la valeur optimale est de -6075, donc la meilleure solution obtenue est éloignée d'environ 10% et le coût médian de 25%.

Nous pouvons observer plusieurs tendances en comparant les résultats avec les deux méthodes. La première est qu'aucune d'elle ne permet d'améliorer significativement la solution optimale pour G_3 et G_4. L'écart entre l'utilisation ou non de l'inversion ne permet pas d'obtenir la solution optimale et l'écart entre elles est négligeable. De plus, la solution médiane n'est pas une solution valide pour le problème de couplage et l'écart n'est pas non plus significatif entre les deux méthodes. Deuxièmement, la pire solution est améliorée en utilisant l'inversion de spin, ce qui laisse entendre que son utilisation pourrait être intéressante avec une sélection supérieure à 10% des coefficients sélectionnés dans cette méthode.

En utilisant le vote majoritaire pour supprimer les mauvaises duplications, les résultats obtenus avec et sans inversions de spins sont sensiblement similaires pour chaque taille d'instance et pour la meilleure, la médiane et la pire solution.

6.4 Discussion

Dans ce chapitre, notre objectif était de fournir une première étude sur le comportement d'un recuit quantique existant lorsqu'il est confronté à un problème combinatoires connu pour vaincre le recuit classique. Notre étude démontre au moins que, ces cas particuliers de problèmes de couplage biparti ne sont en faite pas du tout simples à résoudre par le recuit quantique. De plus, ces problèmes méritent d'être inclus comme une référence standard des problèmes pour ces systèmes émergents. En outre, comme ce dernier problème est polynomial (et que les cas spécifiques examinés dans ce chapitre ont des solutions optimales simples), il permet de quantifier précisément la qualité des solutions obtenues par recuit quantique en terme de distance par rapport à l'optimum.

Nous avons pu observer un certain nombres de points : Premièrement, la nécessité de duplication des qubits limite fortement la taille du problème qui peut être intégré sur le dispositif, ce qui a conduit à un rapport entre 5 et 10 qubits pour une variable du problème. Pourtant, un D-Wave d'environ 1 000 qubits devrait pouvoir s'attaquer à des problèmes combinatoires avec quelques centaines de variables, une taille qui, bien que non négligeable, nécessite d'intégrer des contraintes (par exemple dans notre cas, des contraintes de correspondance exigeant que chaque sommet soit couvert au maximum une fois) dans la fonction économique. Deuxièmement, même avec des constantes de pénalité soigneusement choisies, les solutions obtenues conduisent souvent à des solutions non valables pour le problème de couplage. Cela est vrai aussi bien pour les problèmes de cohérence de duplication des qubits (c'est-à-dire que les qubits représentant la même variable du problème ont des valeurs différentes) mais aussi pour les contraintes spécifiques au problème. Cela signifie que l'utilisation opérationnelle d'un recuit quantique nécessite une ou plusieurs étapes de post-traitement (par exemple, la résolution des incohérences de duplication des qubits par un vote à la majorité), y compris des contraintes spécifiques à un problème (par exemple, la transformation de couplage non valables en couplage valables).

Bien que le D-Wave n'ait pas réussi à trouver des solutions optimales pour des tailles d'instance non triviales, cela n'exclut pas l'existence d'un avantage du recuit quantique *tel qu'il est mis*

en œuvre dans les systèmes D-Wave par rapport au recuit classique (dont l'existence, comme souligné précédemment, a déjà été établie sur des problèmes spécialement conçus [10]). Toutefois, nos résultats tendent à exclure (ou à confirmer) l'absence d'un avantage exponentiel dans le cas général du recuit quantique par rapport au recuit classique.

Comme nous adoptons le point de vue du pire cas (nous considérons des instances qui sont difficiles à résoudre pour le recuit), ce point de vue n'implique pas que ces machines sont incapables d'être utiles en pratique. En effet, leur capacité à résoudre les instances G_4 en quelques dizaines de μs les rendent rapides par rapport aux implémentations logicielles du recuit classique. Dans la même lignée que [145], la présente étude fournit des preuves expérimentales supplémentaires qu'il existe des problèmes polynomiaux liés à NP qui sont difficiles pour le recuit quantique et classique et que le recuit quantique n'est pas beaucoup plus performant.

Chapitre 7

Analyse des contraintes induites par les topologies d'interconnexions des qubits

Résumé

Les travaux expérimentaux présentés précédemment ont confronté l'approche du recuit quantique, telle qu'elle est mise en œuvre dans les dispositifs D-Wave, à des cas pathologiques de problèmes de couplages bipartis (polynomiaux) connus pour être difficile pour les algorithmes classiques de recuit simulé et ont montré que l'approche quantique était également assez peu performante dans de tels cas. Dans cette contribution, nous examinons dans quelle mesure les topologies d'interconnexion des qubits expliquent ces derniers résultats expérimentaux. En particulier, nous fournissons des preuves que la faible connectivité de ces topologies conduisent à des problèmes de QUBO de tailles artificiellement plus grands peut en partie expliquer les observations décevantes du chapitre 6. Par conséquent, ce chapitre laisse entendre que des topologies d'interconnexion plus denses sont nécessaires pour libérer le potentiel de l'approche du recuit quantique.

7.1 Introduction

L'étude [168] que nous avons présenté au Chapitre 6 a confronté expérimentalement une machine quantique "Washington" (2X) aux instances pathologiques du problème de couplage de cardinalité maximale proposées par Sasaki et Hajek [150] afin de montrer que le recuit simulé était effectivement incapable de résoudre certains problèmes polynomiaux en un temps polynomial. Il s'avère que les résultats ont été plutôt décevants et notre hypothèse sur ces observations résulterait probablement de la topologie d'interconnexion des qubits qui serait trop creuse et qu'un nombre relativement important de qubits serait nécessaire pour pouvoir intégrer des instances relativement plus grandes (par exemple, 951 qubits pour 125 variables). De plus, pour

toutes les instances sauf les plus petites en taille, la machine n'a pas réussi à trouver des solutions d'assez bonne qualité malgré l'utilisation de la méthode d'inversion de spin du chapitre 6.

Ce chapitre fait suite à ces travaux et tente d'examiner dans quelle mesure la topologie d'interconnexion des qubits influence ces résultats. Pour ce faire, nous étudions comment le *recuit simulé* est capable de résoudre nos cas difficiles de problèmes de cardinalité maximale lorsqu'ils sont intégrés dans les topologies Chimera et la future topologie d'interconnexion des qubits Pegasus [58].

Topologie Pegasus La prochaine génération de D-Wave pourra atteindre plus de 5 000 qubits interconnectés dans une topologie de graphes appelée Pegasus [36, 58]. Cette topologie englobera l'ancienne topologie Chimera comme un sous-graphe. Elle sera plus dense avec un degré de couplages entre les qubits de 15 contre seulement 6 dans le Chimera. Bien que les machines basées sur la technologie Pegasus ne soient pas encore commercialisées, la chaîne d'outils logiciels D-Wave supporte déjà cette nouvelle interconnexion qui permet de réaliser des expériences préliminaires au moins en termes de simulation des problèmes sur le graphe (comme nous le détaillons dans la section 7.2.1).

7.2 Résultats expérimentaux

7.2.1 Implémentation sur D-Wave

Les matrices QUBO définies dans le chapitre précédent ne sont pas directement intégrables aux topologies d'interconnexion Chimera et Pegasus et, par conséquent, nous devons recourir à la duplication des qubits. Le pipeline logiciel de D-Wave permet d'automatiser ce processus de duplication pour les deux topologies, mais ce besoin de duplication (ou, de manière équivalente, le fait que la topologie d'interconnexion sous-jacente soit si creuse) limite la taille des instances qui peuvent être intégrées sur la machine. Le tableau 7.1 fournit le nombre de qubits requis pour chacune des quatre instances pour G_1, G_2, G_3 et G_4 sur les topologies d'interconnexion Chimera et Pegasus. Comme nous n'avions accès qu'à une machine avec 1098 qubits opérationnels, nous n'avons pu monter en taille d'instances que jusqu'à G_4. De toute évidence, des instances plus importantes (jusqu'à G_5) pouvaient être intégrées sur une machine 2000Q avec 2048 qubits et, comme le montre le tableau 7.1, la topologie encore plus dense Pegasus (qui est prévue dans les machines de la prochaine génération $\approx$ 5 000 qubits) permettra d'aller jusqu'à G_7.

	#var.	#qubits (Chim.)	#qubits (Peg.)
G_1	8	16	8
G_2	27	100	46
G_3	64	431	164
G_4	125	951	513

TABLE 7.1 – Nombre de qubits requis pour traiter les instances QUBO associées à G_1, G_2, G_3 et G_4.

Ainsi, la duplication des qubits conduit donc à un QUBO avec plus de variables et une fonction économique qui comprend un ensemble supplémentaire de contraintes de pénalité. De cette façon, nous pouvons favoriser les solutions dans lesquelles les qubits représentent la même variable et sont bien dans le même état final.

7.2.2 Résolution avec le recuit simulé

Comme le recuit simulé existe depuis longtemps, il n'est pas nécessaire d'introduire la méthode générale mais plutôt de spécifier les choix de paramètres libres utilisés ici. Dans notre cas, nous avons utilisé une loi de décroissance de la température standard de la forme $T_{k+1} = 0,95T_k$ commençant à $T_0 = |c_0|$ (c_0 est le coût élevé de la solution aléatoire initiale) et s'arrêtant lorsque $T < 10^{-3}$. Le paramètre primordial de notre implémentation est le nombre d'itérations de l'algorithme de Metropolis fonctionnant pour chaque k à température constante que nous fixons à n, $n^{1.5}$ et n^2 avec n, le nombre de variables dans le QUBO. Pour n itérations par plateau de température, l'algorithme de Metropolis est rapide mais il doit faire moins d'itérations pour atteindre sa distribution stationnaire. Ainsi, l'algorithme 3 devrait fournir des résultats de moins bonne qualité mais cela nous permet d'obtenir un premier étalon de solutions. À l'autre extrémité du spectre, n^2 itérations par plateau signifie qu'il est possible d'obtenir des résultats de bien meilleure qualité, mais avec un temps de calcul beaucoup plus important. Le tableau 7.2 présente les résultats obtenus lors de la résolution du QUBO initial pour G_1, G_2, G_3 et G_4 avec un recuit simulé et plusieurs nombres d'itérations pour l'algorithme de recuit (sur seulement 30 exécutions).

Á partir des résultats du tableau ci-dessus, nous pouvons constater rapidement que le recuit simulé est bien plus performant comparé au D-Wave. Les pires solutions sur 30 exécutions du recuit sont presque toujours les meilleures solutions obtenues par D-Wave sur 10 000 exécutions (voir tableau 6.4). De plus, le fait que le recuit simulé (même avec seulement n d'itérations par

Algorithm 3 Recuit Simulé

procedure INITIALISATION(S, T, C_a, P) ▷ Solution initiale S_i, Énergie initiale E_i, Différence d'énergie E_{ij}, Température T, Condition d'arrêt C_a, pallier par plateau P
  while NOT $C_a \rightarrow$ **end do**
    while NOT $P \rightarrow$ **end do**
      New solution :S_j
      $E_{ij} = E_j - E_i$ ▷ Différence d'énergie entre les deux solutions
      if $E_{ij} < 0$ **then**
        $S_i = S_j$
      else
        $S_i = S_j$ with as probability $exp(-E_{ij}/T)$ ▷ Probabilité d'accepter la solution
    $T \searrow$
  return S_i ▷ Retourne la meilleure solution

	palier	opt.	best	worst	mean	median	stdev
G_1	n	-68	-68	-68	-68	-68	0
	$n^{1.5}$	-68	-68	-68	-68	-68	0
	n^2	-68	-68	-68	-68	-68	0
G_2	n	-495	-495	-495	-495	-495	0
	$n^{1.5}$	-495	-495	-495	-495	-495	0
	n^2	-495	-495	-495	-495	-495	0
G_3	n	-2064	-2064	-1810	-2004.7	-2064	79.9
	$n^{1.5}$	-2064	-2064	-2064	-2064	-2064	0
	n^2	-2064	-2064	-2064	-2064	-2064	0
G_4	n	-6275	-6275	-5528	-5785.3	-5777	178.9
	$n^{1.5}$	-6275	-6275	-6026	-6241.8	-6275	86.1
	n^2	-6275	-6275	-6275	-6275	-6275	0

TABLE 7.2 – Résultats expérimentaux obtenus lors de la résolution des instances QUBO pour G_1, G_2, G_3 et G_4 au moyen d'un recuit simulé pour plusieurs itérations par plateau de température.

plateau) trouve la solution optimale avec une probabilité élevée, suggère que les tailles des instances jusqu'à G_4 sont trop petites pour atteindre le régime (asymptotique) du nombre exponentiel d'itérations du théorème de Sasaki & Hajek.
Nous pouvons affirmer clairement que ces instances sont suffisamment petites pour rester relativement faciles à résoudre de manière classique, bien que nous puissions observer que la moyenne varie pour un faible nombre d'itérations par plateau (par exemple, pour G_4 aucune variation ne se produit pour n^2 itérations par plateau). Pourtant, comme nous l'avons montré dans le chapitre précédent, le recuit quantique n'a pas réussi à résoudre de manière optimale les instances G_3 et G_4. Il est à souligner que le temps de calcul n'est pas une contrainte supplémentaire pour la résolution de ces cas, le recuit simulé s'exécute en moins de 5 secondes (même G_4 avec n^2 itérations par plateau) sur un PC portable avec un code modérément optimisé. Néanmoins, les instances résolues ici ne sont pas étendues pour être intégrées sur le D-Wave. Ces premiers résultats nous confortent dans l'idée que le recuit simulé est encore capable de résoudre ces tailles de problèmes. Nous allons présenter dans la section suivante les résultats quand nos instances sont dupliquées pour être intégrées sur la machine.

7.2.3 L'étude du biais topologique

Il faut souligner que cette comparaison entre notre recuit simulé et les résultats obtenus sur D-Wave 2X est parfaitement légitime car nous comparons les capacités d'optimisation de deux appareils avec leurs contraintes opérationnelles. Cependant, il faut également noter que, par exemple sur G_4, le recuit simulé a résolu un problème QUBO à 125 variables alors que le recuit quantique a dû résoudre un problème QUBO (artificiellement) plus important à 951 variables pour compenser les limites de l'interconnexion des qubits. Ainsi, bien que le QUBO le plus grand soit équivalent au QUBO non dupliqué, il est nécessaire de vérifier si les QUBO dupliqués sont plus difficiles à résoudre par un recuit simulé.

Pour ce faire, nous avons considéré les instances obtenues après avoir intégré le QUBO dupliqué pour G_4 [1] sur les topologies Chimera et Pegasus et nous avons tenté de les résoudre, cette fois, avec le recuit simulé (voir tableau 7.3).

Le tableau 7.3 fournit les résultats obtenus lors de la résolution des instances QUBO dupliqués pour G_4 sur les topologies Chimera et Pegasus au moyen du recuit simulé (en considérant également plusieurs nombres d'itérations par plateau, comme dans la section précédente). Cette fois, les

1. Nous nous sommes limités à G_4 car c'est la plus grande instance que nous avons pu résoudre.

	palier	opt.	best	worst	mean	median
G_4 (Chim.)	n	-6275	-2213	3662	1453.9	1401.0
	$n^{1.5}$	-6275	-4526	-2654	-3585.6	-3699.8
	n^2	-6275	-5028	-4027	-4473.1	-4527.0
D-Wave		-6275	-5025	-3551	-4447.7	-4525
G_4 (Peg.)	n	-6275	-3930	-785	-2609.3	-2708.5
	$n^{1.5}$	-6275	-5028	-3580	-4305.5	-4281.0
	n^2	-6275	-5278	-4530	-5035.9	-5028.0

TABLE 7.3 – Résultats expérimentaux obtenus lors de la résolution des instances QUBO dupliqués pour G_4 sur les topologies Chimera et Pegasus au moyen d'un recuit simulé (30 passages) pour plusieurs itérations par plateau de température. Notez que la ligne "D-Wave" résulte de la sélection aléatoire de 30 sorties parmi les 10 000 passages qui mènent au tableau 6.4.

résultats obtenus sur D-Wave sont compétitifs avec ceux obtenus par le recuit simulé. Ce qui signifie que les instances dupliquées sont beaucoup plus difficiles à résoudre que les instances initiales, malgré leur équivalence et l'utilisation d'un plus grand nombre d'itérations par plateau (temps de calcul plus important pour les résoudre).
De plus, nous pouvons constater, sans surprise, que la topologie plus dense Pegasus conduit à des QUBO dupliqués plus petits que ceux sur Chimera et donne de meilleurs résultats (avec un recuit simulé car aucune machine n'est encore commercialisée avec cette topologie). Pourtant, bien que cette topologie soit meilleure, les résultats obtenus restent très éloignés de ceux obtenus par le recuit simulé sur les instances QUBO non dupliqués. En terme de temps de "calcul", le D-Wave est plusieurs ordres de grandeur plus rapide que le recuit classique. En effet, lorsqu'il faut moins d'une seconde pour effectuer 10 000 exécutions de recuit quantique, la résolution du QUBO dupliqué (variables $\approx$ 1000) sur G_4 avec le recuit simulé pour n^2 itérations par plateau, prend maintenant plusieurs minutes sur un ordinateur portable [2].

En conclusion de cette section, il apparaît donc que la machine D-Wave est compétitive avec un algorithme de recuit simulé pour n^2 itérations par plateau en termes de qualité d'optimisation et pour ce cas précis, surclasse l'ordinateur classique de plusieurs ordres de grandeur en terme de rapidité. A noter tout de même que nos résultats sont compétitifs avec ceux du recuit simulé quand nous résolvons un problème de la même façon que la machine D-Wave nous force à le rendre plus complexe (plus dense en coefficients). En effet, sans la duplication qui complexifie artificiellement le problème initial, le recuit simulé obtient la solution optimale. Dans ce cas,

2. Et le recuit simulé étant un algorithme intrinsèquement séquentiel, il se parallélise mal.

il n'est pas nécessaire de transformer l'instance en un problème plus dense pour le résoudre efficacement en terme de qualité de solutions. Le recuit simulé est compétitif seulement lorsque nous résolvons le même problème que le recuit quantique.
Il apparaît que le fait de devoir intégrer des instances QUBO dans les topologies Chimera et Pegasus tend à produire des QUBO plus gros et plus dense, qui sont beaucoup plus difficiles à résoudre avec le recuit simulé. Cela laisse donc entrevoir qu'elles sont également contre-productive pour le recuit quantique et que ces topologies d'interconnexion de qubits devraient être plus densément connectées.

7.3 Discussion

Ce chapitre fait partie d'une série de travaux qui visent à étudier le comportement des recuits quantiques existants lorsqu'ils sont confrontés à d'anciens problèmes combinatoires connus pour piéger le recuit classique, comme le cas particulier du problème de couplage de cardinalité maximale. En outre, comme ce dernier problème est polynomial, il permet de quantifier précisément la qualité des solutions obtenues par le recuit quantique en terme de distance à l'optimum.

Ce chapitre démontre expérimentalement que la topologie d'interconnexions des qubits dans les ordinateurs quantiques adiabatiques existants est une étape nécessaire sur la compréhension du potentiel de cette technologie. Tout d'abord, la nécessité de duplication des qubits limite considérablement la taille des problèmes qui peuvent être intégrés sur les dispositifs de recuit quantique. Ensuite, comme l'illustre plus précisément ce chapitre, ce besoin de duplication tend également à alourdir le problème d'optimisation à résoudre, ce qui conduit à des résultats de qualité nettement inférieure à ceux connus. Ce fait tend malheureusement à faire disparaître l'avantage temporel écrasant du recuit quantique par rapport au recuit simulé.

De plus, nos résultats montrent que, lorsqu'ils sont résolus sans tenir compte d'une topologie (ou, de manière équivalente, en supposant un réseau de qubits entièrement connecté), les tailles d'instance que nous sommes capables de intégrer sur le D-Wave restent résolvables par le recuit simulé. En d'autres termes, le régime dans lequel ces instances deviennent (asymptotiquement) difficiles à résoudre n'est pas encore accessible à ces machines, en raison de la nécessité d'assigner des variables à plusieurs qubits. Ensuite, lorsque le recuit simulé est utilisé pour résoudre les instances QUBO (artificiellement) plus grandes résultant des problèmes originaux, il ne donne pas de meilleurs résultats qu'un recuit quantique. Cela indique donc que les contraintes imposées par les interconnexions de qubits actuellement utilisées tendent à obscurcir suffisamment

le problème d'optimisation pour empêcher le recuit classique et le recuit quantique de fonctionner correctement.

Par conséquent, même si l'informatique quantique analogique peut bien sûr présenter de nombreux intérêts d'un point de vue théorique et expérimental, nous soutenons qu'à moins que des interconnexions de qubits beaucoup plus denses ne soient développées, il sera difficile pour cette approche de concurrencer les algorithmes classiques sur les problèmes du monde réel, tant en termes de taille que de complexité du modèle, même si le nombre de qubits ne cesse d'augmenter.

Chapitre 8

Conclusion et travaux futurs

8.1 Conclusion

Les travaux menés dans le cadre de cette thèse visaient à évaluer les performances et les caractéristiques d'un ordinateur quantique de type D-Wave basé sur le principe du recuit quantique. Á ce jour, il existe de nombreux travaux et résultats pour démontrer que ces machines sont bien quantiques (ou non) et sur leur utilité, mais très peu sur leurs performances.

Nous avons pu voir au chapitre 5 qu'une nouvelle approche pour résoudre des problèmes QUBO denses en ne faisant appel qu'une seule fois au recuit quantique restait un problème encore assez difficile pour les machines actuellement commercialisées. L'algorithme mis en œuvre avait pour but de venir sélectionner des coefficients et de les intégrer sur le graphe Chimera. L'objectif de cette approche était d'étudier l'existence de relaxations creuses conformes à la topologie d'interconnexion des qubits de la machine afin de savoir s'il est possible d'utiliser une seule invocation de l'oracle.

Dans cette contribution, les résultats obtenus nous ont permis d'observer deux tendances. La première, les problèmes traités admettent bien des relaxations creuses avec une bonne qualité de solutions par rapport aux solutions obtenues par d'autres solveurs. L'intérêt est d'obtenir en premier lieu des matrices conforment aux nombres d'arrêtes disponibles dans le Chimera correspondant mais sans être isomorphes à la topologie. La deuxième, les matrices obtenues après relaxations qui sont à la fois creuses et isomorphes à la topologie Chimera nous ont permis d'observer que les résultats ne sont pas significatifs pour être de bonne qualité par rapport aux autres algorithmes.

Nous pouvons ainsi conclure que la résolution de problèmes arbitraires, même de taille modérée, avec un seul appel au recuit quantique n'est expérimentalement pas envisageable. Une voie

possible, si notre algorithme de relaxation avait donné des solutions proches de celles données par D-Wave, aurait été la conception d'heuristiques tirant profit de l'oracle pour résoudre efficacement des problèmes de plus grandes tailles. Mais les résultats obtenus au Chapitre 5 montrent que déterminer un bon sous-ensemble de coefficients et obtenir une bonne qualité de solution est possible en omettant la topologie d'interconnexion des qubits. En incluant le graphe formé par la sélection des coefficients comme isomorphe à la topologie, nous observons des solutions de mauvaises qualités (écart des coûts avec notre premier algorithme de plus de 50%).

A partir de cette première analyse sur la capacité d'un D-Wave à résoudre efficacement des problèmes arbitraires, nous nous sommes intéréssés au Chapitre 6 sur le comportement de cette machine lorsqu'elle est confrontée cette fois-ci à des instances très particulières. Notre objectif était de fournir une étude sur le comportement d'un recuit quantique lorsqu'il est confronté à un problème combinatoire connu pour piéger le recuit classique. Ce problème, le problème de couplage biparti de cardinalité maximale, a été choisi spécifiquement pour être difficile à résoudre au moyen d'un recuit simulé. L'intérêt est de tester le processeur "Washington"(2X) avec 1098 qubits opérationnels sur différentes tailles de ces instances pathologiques afin de déterminer s'il est capable de les résoudre. Nous avons pu observer que pour toutes les tailles d'instances intégrables sur le D-Wave, sauf les plus triviales, le recuit quantique ne parvient pas à obtenir la solution optimale. De plus la meilleure solution obtenue pour les plus grandes instances ne sont pas des couplages et ne correspondent pas à une solution valide du problème. Nous pouvons en conclure que le recuit quantique est lui aussi incapable de donner une bonne solution au problème et qu'il tombe dans les mêmes pièges que le recuit classique. Plus généralement, nous pouvons aussi en déduire qu'il existe au moins un problème polynomial qu'une telle machine ne peut pas résoudre efficacement. En outre, il nous permet de quantifier la qualité des solutions obtenues par rapport l'optimalité du problème.

Cependant, le fait que le D-Wave n'arrive pas à obtenir la solution optimale pour des d'instances de grandes tailles, n'exclut en rien la possible existence d'un avantage quantique par rapport aux machines classiques [10] dans d'autres cas. Dans notre cas précis, les résultats impliquent l'absence d'un avantage exponentiel dans le cas général du recuit quantique par rapport au recuit classique. En effet, nous avons adopté un point de vue "pire cas" pour déterminer avec précision sa capacité à résoudre efficacement un problème. Mais les résultats n'impliquent pas

que les machines à recuit quantique soient inutiles, et que leur capacité à résoudre en quelques dizaines de μs offre une rapidité de calcul non négligeable par rapport aux implémentations du recuit classique. Néanmoins, la nécessité de dupliquer les qubits oblige à augmenter considérablement la taille des problèmes et tend aussi à alourdir le problème d'optimisation à résoudre. Cette contrainte conduit à des résultats nettement moins bons que ceux attendus et fait disparaître l'avantage temporel du recuit quantique par rapport au recuit simulé.

Les résultats du D-Wave sur le problème de couplage nous a fourni une base expérimentale qui peut être comparée aux résultats d'un recuit simulé. Le Chapitre 7 nous a permis de comparer les deux types de recuits et de vérifier expérimentalement les conclusions du Chapitre 6. Théoriquement nous savons que ces instances sont un piège pour le recuit simulé, mais nous avons vu que pour être intégrées sur un D-Wave, même la plus petite instance G_1 doit être adaptée à l'architecture. Pour comparer les résultats des deux types de recuits, il nous a fallu résoudre le QUBO étendu avec un recuit simulé. Ainsi, nous avons examiné dans quelle mesure la topologie Chimera explique les résultats du chapitre 6. La conclusion qui est alors apparue est que cette topologie oblige les problèmes QUBO à être de taille artificiellement plus importantes et sont plus difficiles à résoudre. En comparant les résultats sur toutes les tailles d'instances nous pouvons confirmer que les deux recuits obtiennent des résultats similaires (mais pour G_3 et G_4 pas la solution optimale). Par conséquent, nous pouvons affirmer que des topologies d'interconnexion plus denses sont nécessaires pour réellement réaliser le potentiel de l'approche de résolution par recuit quantique.

Ces conclusions sur la contrainte topologique impliquant qu'aucun des deux recuits ne peut résoudre efficacement le problème est facilement vérifiable. En effet nos résultats montrent que, lorsque nous résolvons toutes les tailles d'instances sans tenir compte de la topologie, nous sommes capables d'obtenir la solution optimale avec un recuit simulé. Ainsi, le régime dans lequel les instances deviennent difficiles à résoudre n'est pas accessible aux D-Wave en raison de la nécessité d'assigner les variables à plusieurs qubits. Nous pouvons en conclure que les contraintes imposées par les interconnexions des qubits tendent à alourdir suffisamment le problème d'optimisation pour empêcher le recuit classique et le recuit quantique de fonctionner de manière optimale.

En premier lieu, l'obligation de dupliquer les qubits limite fortement la taille des problèmes qui peuvent être intégrés sur l'ordinateur. En effet, un D-Wave d'environ 1 000 qubits ne peut s'attaquer qu'à des problèmes combinatoires qu'avec quelques centaines de variables (une taille

qui est certes non négligeable). En outre, la nécessité d'intégrer des contraintes (par exemple dans notre cas, des contraintes de couplage exigeant que chaque sommet soit couvert au maximum une fois) dans la fonction économique, même avec des constantes de pénalité soigneusement choisies, conduit souvent à des solutions non valables. Cela est vrai aussi bien pour les problèmes de cohérence de duplication des qubits (c'est-à-dire lorsque que les qubits représentant la même variable du problème ont des valeurs différentes) mais aussi pour les contraintes spécifiques au problème. Cela signifie que l'utilisation opérationnelle d'un recuit quantique nécessite une ou plusieurs étapes de post-traitement (par exemple, la résolution des incohérences de duplication par un vote à la majorité), y compris des contraintes spécifiques à un problème.

Par conséquent, même si l'informatique quantique analogique peut bien sûr présenter de nombreux intérêts, nous soutenons qu'à moins que des interconnexions de qubits beaucoup plus denses ne soient développées, il sera difficile pour cette approche de concurrencer les algorithmes classiques sur les problèmes du monde réel, tant en termes de taille que de complexité du modèle, même si le nombre de qubits ne cesse d'augmenter.

8.2 Travaux futurs

Premièrement comme nous l'avons montré, la topologie devrait fortement être remise en question afin de moins dupliquer les qubits pour une seule variable et ainsi éviter de créer des problèmes artificiellement plus complexes. Bien entendu, une telle topologie devra également être confrontée aux contraintes physiques de réalisation. Dans cet objectif, il serait avantageux de tester nos problèmes sur la prochaine génération des machines D-Wave avec la topologie Pegasus [58] qui devrait être 2.5 fois plus denses en connectivité que le Chimera. En plus d'avoir une meilleure connectivité, la future génération de machine D-Wave aura un temps de cohérence et une qualité des états quantiques bien supérieurs ce qui permettra en théorie d'améliorer la qualité du recuit.
Même si la nouvelle topologie apportera des résultats significativement de meilleure qualité d'après l'étude montrée dans le Chapitre 7, la topologie Chimera disponible sur les D-Wave actuels reste une contrainte forte pour venir intégrer des problèmes même peu denses en coefficients.

Sur les instances du problème de couplage biparti, un point de vue théorique serait nécessaire pour adapter la preuve de Sasaki et Hajek [150] dans le cadre du recuit quantique. En effet il serait judicieux de s'appuyer mathématiquement sur ces problèmes afin de confronter les résultats

théoriques avec nos résultats sur la machine. Au vu de nos résultats expérimentaux, nous avons pu constater que la machine n'apporte pas d'avantage significatif en terme de performance. La question qui en résulte est, est-ce une limitation intrinsèque du recuit adiabatique en tant que tel ou est-ce la machine D-Wave qui implémente imparfaitement le modèle théorique ?
Faire la preuve relative au modèle théorique apportera un élément de réponse à cette question. Nous savons que le recuit simulé ne résout pas efficacement ce type de problème de couplage (nombre exponentiel d'itérations en moyenne), mais pour le modèle adiabatique, est-ce que l'implémentation du modèle théorique empêche la machine d'obtenir la solution optimale ou est-ce le modèle théorique qui est lui-même limité ?

Malheureusement, si c'est le modèle théorique, quelle que soit la machine D-Wave, elle aura elle-même une limitation. En revanche, si c'est le contraire, nous ne devrions pas obtenir un nombre exponentiel d'itérations sur ces instances. Dans ce cas-là, la machine limite fortement la bonne résolution du problème et peut être que la preuve nous montrera comment améliorer la machine.

Pour finir sur les perspectives, le problème de couplage sur la famille des graphes G_n nous permettrait d'étudier des algorithmes classiques émergents qui s'inspirent du quantique. Le Recuit Quantique Simulé [55] (SQA) pourrait être un nouvel étalon entre le recuit classique et le recuit quantique. Pouvoir comparer ces trois types de recuits sur des problèmes aussi difficiles pour cette famille d'algorithmes nous permettrait de mieux situer la qualité des solutions obtenues.
En effet, l'intérêt de pouvoir comparer les résultats du D-Wave avec ceux d'un SQA nous permettra de déterminer s'il existe des problèmes de cohérences liés à la machine ou si la limitation sur nos résultats vient du recuit quantique adiabatique.

Répondre à cette question nous donnera un deuxième élément de réponse. Est-ce que le recuit adiabatique n'est pas capable de résoudre le problème ou le recuit adiabatique donne de bonne solutions mais la machine D-Wave n'y parvient pas ? Dans ce cas-là, c'est plus un problème d'adéquation entre la théorie et la machine qui fait que nous observons des résultats de mauvaise qualité. Ou alors, le modèle théorique empêche forcément la bonne résolution des problèmes et donc la machine ne pourra pas résoudre correctement les problèmes puisqu'elle implémente le modèle théorique imparfait.

Bibliographie

[1] Scott AARONSON. « BQP and the polynomial hierarchy ». In : *Proceedings of the forty-second ACM symposium on Theory of computing*. 2010, p. 141-150.

[2] Scott AARONSON. « The limits of quantum ». In : *Scientific American* 298.3 (2008), p. 62-69.

[3] Scott AARONSON et Alex ARKHIPOV. « The computational complexity of linear optics ». In : *Proceedings of the forty-third annual ACM symposium on Theory of computing*. 2011, p. 333-342.

[4] Scott AARONSON et Lijie CHEN. « Complexity-theoretic foundations of quantum supremacy experiments ». In : *arXiv preprint arXiv :1612.05903* (2016).

[5] Emile HL AARTS, Peter JM VAN LAARHOVEN et al. « Statistical cooling : A general approach to combinatorial optimization problems. » In : *Philips J. Res.* 40.4 (1985), p. 193-226.

[6] Ali J ABHARI et al. *Scaffold : Quantum programming language*. Rapp. tech. PRINCETON UNIV NJ DEPT OF COMPUTER SCIENCE, 2012.

[7] Dorit AHARONOV et al. « Adiabatic quantum computation is equivalent to standard quantum computation ». In : *SIAM review* 50.4 (2008), p. 755-787.

[8] Farhi E. et AL. « Quantum computation by adiabatic evolution ». In : *arXiv preprint quant-ph/0001106* (2000).

[9] Harris R. et AL. « Experimental demonstration of a robust and scalable flux qubit ». In : *Phys. Rev. B* 81 (13 2010), p. 134510.

[10] T. ALBASH et D. LIDAR. « Demonstration of a scaling advantage for a Quantum Annealer over Simulated Annealing ». In : *Physical Review X* 8 (3 2018).

[11] Tameem ALBASH, Victor MARTIN-MAYOR et Itay HEN. « Temperature scaling law for quantum annealing optimizers ». In : *Physical review letters* 119.11 (2017), p. 110502.

[12] Tameem ALBASH et al. « Consistency tests of classical and quantum models for a quantum annealer ». In : *Physical Review A* 91.4 (2015), p. 042314.

[13] Greg ALOUPIS et al. « Classic Nintendo games are (computationally) hard ». In : *Theoretical Computer Science* 586 (2015), p. 135-160.

[14] Thorsten ALTENKIRCH et Jonathan GRATTAGE. « A functional quantum programming language ». In : *20th Annual IEEE Symposium on Logic in Computer Science (LICS'05)*. IEEE. 2005, p. 249-258.

[15] Boris ALTSHULER, Hari KROVI et Jérémie ROLAND. « Anderson localization makes adiabatic quantum optimization fail ». In : *Proceedings of the National Academy of Sciences* 107.28 (2010), 12446–12450. ISSN : 1091-6490. DOI : 10.1073/pnas.1002116107.

[16] Mohammad HS AMIN et Miles FH STEININGER. *Adiabatic quantum computation with superconducting qubits*. US Patent 7,135,701. 2006.

[17] I ANTONIADIS, E CREMMER et KS STELLE. « Les supercordes ». In : *Gazette des Mathématiciens* 87 (2001), p. 17-41.

[18] Sanjeev ARORA et Boaz BARAK. *Computational complexity : a modern approach*. Cambridge University Press, 2009.

[19] Frank ARUTE et al. « Quantum supremacy using a programmable superconducting processor ». In : *Nature* 574.7779 (2019), p. 505-510.

[20] Alain ASPECT, Jean DALIBARD et Gérard ROGER. « Experimental test of Bell's inequalities using time-varying analyzers ». In : *Physical review letters* 49.25 (1982), p. 1804.

[21] COMMISSARIAT À L'ENERGIE ATOMIQUE. « Le supercalculateur TERA-10 ». In : (2006).

[22] T. BAKER, J. GILL et R. SOLOVAY. « Relativizations of the P not egal to NP question ». In : *SIAM Journal on Computing* 4 (1975), p. 431-442.

[23] Francisco BARAHONA. « On the computational complexity of Ising spin glass models ». In : *Journal of Physics A : Mathematical and General* 15.10 (1982), p. 3241.

[24] Adriano BARENCO et al. « Elementary gates for quantum computation ». In : *Physical review A* 52.5 (1995), p. 3457.

[25] Ethan BERNSTEIN et Umesh VAZIRANI. « Quantum complexity theory ». In : *SIAM Journal on computing* 26.5 (1997), p. 1411-1473.

[26] Stefano BETTELLI, Tommaso CALARCO et Luciano SERAFINI. « Toward an architecture for quantum programming ». In : *The European Physical Journal D-Atomic, Molecular, Optical and Plasma Physics* 25.2 (2003), p. 181-200.

[27] Jacob BIAMONTE et al. « Quantum machine learning ». In : *Nature* 549.7671 (2017), p. 195-202.

[28] Zhengbing BIAN et al. « Discrete optimization using quantum annealing on sparse Ising models ». In : *Frontiers in Physics* 2 (2014), p. 56.

[29] Zhengbing BIAN et al. « Mapping constrained optimization problems to quantum annealing with application to fault diagnosis ». In : *Frontiers in ICT* 3 (2016), p. 14.

[30] F. BLOCH. « Nuclear Induction ». In : *Phys. Rev.* 70 (7-8 1946), p. 460-474. DOI : 10.1103/PhysRev.70.460.

[31] Sergio BOIXO et al. « Characterizing quantum supremacy in near-term devices ». In : *Nature Physics* 14.6 (2018), p. 595-600.

[32] Sergio BOIXO et al. « Evidence for quantum annealing with more than one hundred qubits ». In : *Nature physics* 10.3 (2014), p. 218-224.

[33] Sergio BOIXO et al. « Experimental signature of programmable quantum annealing ». In : *Nature communications* 4 (2013), p. 2067.

[34] Sergio BOIXO et al. « Quantum annealing with more than one hundred qubits ». In : *arXiv preprint arXiv :1304.4595* (2013).

[35] M. BOOTH, S. P. REINHARDT et A. ROY. « Partitioning optimization problems for hybrid classical ». In : *quantum execution. Technical Report* (2017), p. 01-09.

[36] Kelly BOOTHBY et al. « Next-generation topology of d-wave quantum processors ». In : *arXiv preprint arXiv :2003.00133* (2020).

[37] Max BORN. « Quantenmechanik der stoßvorgänge ». In : *Zeitschrift für Physik* 38.11-12 (1926), p. 803-827.

[38] Sergey BRAVYI, David GOSSET et Robert KÖNIG. « Quantum advantage with shallow circuits ». In : *Science* 362.6412 (2018), p. 308-311.

[39] Geoff BRUMFIEL. *Quantum computing at 16 qubits*. 2007.

[40] Jean-Luc BRYLINSKI et Ranee BRYLINSKI. « Universal quantum gates ». In : *Mathematics of quantum computation* 79 (2002).

[41] Paul I BUNYK et al. « Architectural considerations in the design of a superconducting quantum annealing processor ». In : *IEEE Transactions on Applied Superconductivity* 24.4 (2014), p. 1-10.

[42] G BURDET et M PERRIN. « Spéculation sur la cosmologie quantique primordiale ». In : *letters in mathematical physics* 30.4 (1994), p. 317-325.

[43] R. E. BURKARD et U. FINCKE. « Probabilistic asymptotic properties of some combinatorial optimization problems ». In : *Discrete Mathematics* 12 (1985), p. 21-29.

[44] Jun CAI, William G MACREADY et Aidan ROY. « A practical heuristic for finding graph minors ». In : *arXiv preprint arXiv :1406.2741* (2014).

[45] Yudong CAO, Gian Giacomo GUERRESCHI et Alán ASPURU-GUZIK. *Quantum Neuron : an elementary building block for machine learning on quantum computers*. 2017. arXiv : 1711.11240 [quant-ph].

[46] V. CERNY. « Thermodynamical approach to the traveling salesman problem : an efficient simulation algorithm ». In : *Journal of Optimization Theory and Applications* 5 (1985), p. 41-51.

[47] Serena CHEN, Kimberly DUCKWORTH et Shelly CHAIKEN. « Motivated heuristic and systematic processing ». In : *Psychological Inquiry* 10.1 (1999), p. 44-49.

[48] Gabriel CHÊNEVERT, Mathématiques APPLIQUÉES et ISEN LILLE. « Dimension algorithmique et chiffrement post-quantique ». In : (2018).

[49] Andrew M CHILDS. « Lecture notes on quantum algorithms ». In : *Lecture notes at University of Maryland* (2017).

[50] Andrew M CHILDS, Edward FARHI et John PRESKILL. « Robustness of adiabatic quantum computation ». In : *Physical Review A* 65.1 (2001), p. 012322.

[51] V. CHOI. « Minor-embedding in adiabatic quantum computation : I. The parameter setting problem ». In : *Quantum Information Processing* 7.5 (2008), p. 193-209.

[52] Stephen A COOK. *The complexity of theorem-proving procedures, STOC'71 : Proceedings of the third annual ACM symposium on Theory of computing*. 1971.

[53] Rachel COURTLAND. « Google plans to demonstrate the supremacy of quantum computing ». In : *IEEE Spectrum* (2017).

[54] Pierluigi CRESCENZI et al. « A compendium of NP optimization problems ». In : *URL : http ://www. nada. kth. se/˜ viggo/problemlist/compendium. html* (1997).

[55] E. CROSSON et A. W. HARROW. « Simulated Quantum Annealing can be exponentially faster than Classical Simulated Annealing ». In : *IEEE FOCS*. 2016, p. 714-723.

[56] Elizabeth CROSSON et al. « Different strategies for optimization using the quantum adiabatic algorithm ». In : *arXiv preprint arXiv :1401.7320* (2014).

[57] W. van DAM, M. MOSCA et U. VAZIRANI. « How powerful is adiabatic quantum computation ? » In : *Proceedings 42nd IEEE Symposium on Foundations of Computer Science* (2001). DOI : 10.1109/sfcs.2001.959902.

[58] Nike DATTANI, Szilard SZALAY et Nick CHANCELLOR. « Pegasus : The second connectivity graph for large-scale quantum annealing hardware ». In : *arXiv preprint arXiv :1901.07636* (2019).

[59] Nikesh S. DATTANI et Nathaniel BRYANS. *Quantum factorization of 56153 with only 4 qubits*. 2014. arXiv : 1411.6758 [quant-ph].

[60] Vasil S DENCHEV et al. « What is the computational value of finite-range tunneling ? » In : *Physical Review X* 6.3 (2016), p. 031015.

[61] David DEUTSCH. « Quantum theory, the Church-Turing principle and the universal quantum computer ». In : *Proceedings of the Royal Society of London A : Mathematical, Physical and Engineering Sciences*. T. 400. 1818. The Royal Society. 1985, p. 97-117.

[62] David DEUTSCH et Richard JOZSA. « Rapid solution of problems by quantum computation ». In : *Proceedings of the Royal Society of London. Series A : Mathematical and Physical Sciences* 439.1907 (1992), p. 553-558.

[63] Keith J DEVLIN et al. *The millennium problems : the seven greatest unsolved mathematical puzzles of our time*. T. 100. Basic books New York, 2002.

[64] Michael J DINNEEN, Anuradha MAHASINGHE et Kai LIU. « Finding the chromatic sums of graphs using a D-Wave quantum computer ». In : *The Journal of Supercomputing* 75.8 (2019), p. 4811-4828.

[65] Paul Adrien Maurice DIRAC. « A new notation for quantum mechanics ». In : *Mathematical Proceedings of the Cambridge Philosophical Society*. T. 35. 3. Cambridge University Press. 1939, p. 416-418.

[66] David P DIVINCENZO. « Quantum gates and circuits ». In : *Proceedings of the Royal Society of London. Series A : Mathematical, Physical and Engineering Sciences* 454.1969 (1998), p. 261-276.

[67] David P DIVINCENZO et al. « The physical implementation of quantum computation ». In : *arXiv preprint quant-ph/0002077* (2000).

[68] Jonathan P DOWLING. *Schrödinger's killer app : race to build the world's first quantum computer*. CRC Press, 2013.

[69] Johann DRÉO et al. *Métaheuristiques pour l'optimisation difficile*. 2003.

[70] Richard W EGLESE. « Simulated annealing : a tool for operational research ». In : *European journal of operational research* 46.3 (1990), p. 271-281.

[71] Paul EHRENFEST. « Le principe de correspondance ». In : *Paper read in the Third Solvay Conference. In [Solvay 1923, pp. 248–254]. Reprinted in [Klein 1959a, pp. 436–442]*. 1923.

[72] Albert EINSTEIN, Boris PODOLSKY et Nathan ROSEN. « Can quantum-mechanical description of physical reality be considered complete ? » In : *Physical review* 47.10 (1935), p. 777.

[73] Charles EPSTEIN. « Adiabatic quantum computing : An overview ». In : *Quantum Complexity Theory* 6.845 (2012), p. 26.

[74] E. FARHI, J. GOLDSTONE et S. GUTMANN. *Quantum adiabatic evolution algorithms versus simulated annealing*. Rapp. tech. 0201031. arXiv :quant-ph, 2002.

[75] E. FARHI et al. « A Quantum Adiabatic Evolution Algorithm Applied to Random Instances of an NP-Complete Problem ». In : *Science* 292.5516 (2001), 472–475. ISSN : 1095-9203. DOI : 10.1126/science.1057726.

[76] Edward FARHI, Jeffrey GOLDSTONE et Sam GUTMANN. *Quantum Adiabatic Evolution Algorithms versus Simulated Annealing*. 2002. arXiv : quant-ph/0201031 [quant-ph].

[77] Edward FARHI et al. *How to Make the Quantum Adiabatic Algorithm Fail*. 2005. arXiv : quant-ph/0512159 [quant-ph].

[78] Edward FARHI et al. « Performance of the quantum adiabatic algorithm on random instances of two optimization problems on regular hypergraphs ». In : *Physical Review A* 86.5 (2012). ISSN : 1094-1622. DOI : 10.1103/physreva.86.052334.

[79] Edward FARHI et al. *Quantum Adiabatic Algorithms, Small Gaps, and Different Paths*. 2009. arXiv : 0909.4766 [quant-ph].

[80] Richard P FEYNMAN. « Simulating physics with computers ». In : *International journal of theoretical physics* 21.6 (1982), p. 467-488.

[81] Aleta Berk FINNILA et al. « Quantum annealing : a new method for minimizing multidimensional functions ». In : *Chemical physics letters* 219.5-6 (1994), p. 343-348.

[82] Frank GAITAN et Lane CLARK. « Graph isomorphism and adiabatic quantum computing ». In : *Physical Review A* 89.2 (2014). ISSN : 1094-1622. DOI : 10.1103/physreva.89.022342.

[83] Frank GAITAN et Lane CLARK. « Ramsey Numbers and Adiabatic Quantum Computing ». In : *Physical Review Letters* 108.1 (2012). ISSN : 1079-7114. DOI : 10.1103/physrevlett.108.010501.

[84] Michael R GAREY et David S JOHNSON. *Computers and intractability*. T. 174. freeman San Francisco, 1979.

[85] Silvano GARNERONE, Paolo ZANARDI et Daniel A LIDAR. « Adiabatic quantum algorithm for search engine ranking ». In : *Physical review letters* 108.23 (2012), p. 230506.

[86] S. GEMAN et D. GEMAN. « Stochastic Relaxation, Gibbs Distribution, and the Bayesian Restoration of Images ». In : *IEEE Transactions on Pattern Analysis and Machine Intelligence* (1984), p. 721-741.

[87] Daniel GOTTESMAN. *An Introduction to Quantum Error Correction*. 2000. arXiv : quant-ph/0004072 [quant-ph].

[88] Alexander S GREEN et al. « An introduction to quantum programming in quipper ». In : *International Conference on Reversible Computation*. Springer. 2013, p. 110-124.

[89] Lev GROSSMAN. « The quantum quest for a revolutionary computer ». In : *Time, Issue of* 17 (2014).

[90] Lov K GROVER. « A fast quantum mechanical algorithm for database search ». In : *Proceedings of the twenty-eighth annual ACM symposium on Theory of computing*. 1996, p. 212-219.

[91] Stanley P GUDDER. « A superposition principle in physics ». In : *Journal of Mathematical Physics* 11.3 (1970), p. 1037-1040.

[92] Christophe GUYEUX et al. « Is protein folding problem really a NP-complete one ? First investigations ». In : *Journal of bioinformatics and computational biology* 12.01 (2014), p. 1350017.

[93] B. HAJEK. « Cooling Schedule for Optimal Annealing ». In : *Mathematics of Operations Research* 13 (1988), p. 311-329.

[94] Bruce HAJEK et Galen SASAKI. « Simulated annealing—to cool or not ». In : *Systems & control letters* 12.5 (1989), p. 443-447.

[95] Sean HALLGREN. « Polynomial-time quantum algorithms for Pell's equation and the principal ideal problem ». In : *Journal of the ACM (JACM)* 54.1 (2007), p. 4.

[96] Aram W HARROW et Ashley MONTANARO. « Quantum computational supremacy ». In : *Nature* 549.7671 (2017), p. 203-209.

[97] Corporate HEADQUARTERS. « Programming with D-Wave : Map Coloring Problem ». In : (2013).

[98] Werner HEISENBERG. « Physics and beyond : Encounters and conversations ». In : (1971).

[99] M. van Houweninge J. B. G. FRENK et A. H. G. Rinnooy KAN. « Asymptotic properties of the quadratic assignment problem ». In : *Mathematics of Operations Research* 10 (1985), p. 100-116.

[100] Ik Su JANG. « Quantum Computer ». In : (2010).

[101] Shuxian JIANG et al. « Quantum annealing for prime factorization ». In : *Scientific reports* 8.1 (2018), p. 1-9.

[102] Mark W JOHNSON et al. « Quantum annealing with manufactured spins ». In : *Nature* 473.7346 (2011), p. 194.

[103] Tadashi KADOWAKI et Hidetoshi NISHIMORI. « Quantum annealing in the transverse Ising model ». In : *Physical Review E* 58.5 (1998), p. 5355.

[104] Phillip KAYE, Raymond LAFLAMME, Michele MOSCA et al. *An introduction to quantum computing*. Oxford university press, 2007.

[105] Iordanis KERENIDIS, Anupam PRAKASH et Dániel SZILÁGYI. *Quantum Algorithms for Portfolio Optimization*. 2019. arXiv : 1908.08040 [math.OC].

[106] James KING et al. « Quantum annealing amid local ruggedness and global frustration ». In : *Journal of the Physical Society of Japan* 88.6 (2019), p. 061007.

[107] Scott KIRKPATRICK, C Daniel GELATT et Mario P VECCHI. « Optimization by simulated annealing ». In : *science* 220.4598 (1983), p. 671-680.

[108] Scott KIRKPATRICK, Bart SELMAN et al. « Critical behavior in the satisfiability of random boolean expressions ». In : *Science-AAAS-Weekly Paper Edition-including Guide to Scientific Information* 264.5163 (1994), p. 1297-1300.

[109] Gary A. KOCHENBERGER et al. « The unconstrained binary quadratic programming problem : a survey ». In : *J. Comb. Optim.* 28.1 (2014), p. 58-81.

[110] B. KORTE et J. VYGEN. *Combinatorial optimization, theory and algorithms*. Springer, 2012.

[111] Kazue KUDO. « Constrained quantum annealing of graph coloring ». In : *Physical Review A* 98.2 (2018), p. 022301.

[112] Trevor LANTING et al. « Entanglement in a quantum annealing processor ». In : *Physical Review X* 4.2 (2014), p. 021041.

[113] Benjamin P LANYON et al. « Towards quantum chemistry on a quantum computer ». In : *Nature chemistry* 2.2 (2010), p. 106-111.

[114] Ryan LAROSE. « Overview and comparison of gate level quantum software platforms ». In : *Quantum* 3 (2019), p. 130.

[115] Eugene L LAWLER et al. *The traveling salesman problem : a guided tour of combinatorial optimization*. T. 3. Wiley New York, 1985.

[116] M LEWIS et F GLOVER. « Quadratic unconstrained binary optimization problem preprocessing : Theory and empirical analysis ». In : *Networks* 70.2 (2017), p. 79-97.

[117] A. LUCAS. « Ising formulations of many NP problems ». In : *Frontiers in Physics* 2 (2014), p. 5.

[118] W MACREADY. *Programming with QUBOs*.

[119] Yu I MANIN. *Vychislimoe i nevychislimoe (Computable and Noncomputable), Moscow : Sov*. 1980.

[120] Roman MARTOŇÁK, Giuseppe E SANTORO et Erio TOSATTI. « Quantum annealing of the traveling-salesman problem ». In : *Physical Review E* 70.5 (2004), p. 057701.

[121] C. MCGEOCH et C. WANG. « Experimental Evaluation of an Adiabiatic Quantum System for Combinatorial Optimization ». In : *Proceedings of the ACM International Conference on Computing Frontiers*. CF '13. Ischia, Italy : ACM, 2013, 23 :1-23 :11. ISBN : 978-1-4503-2053-5.

[122] Eugen MERZBACHER. « The early history of quantum tunneling ». In : *Physics Today* 55.8 (2002), p. 44-50.

[123] Nicholas METROPOLIS et al. « Equation of state calculations by fast computing machines ». In : *The journal of chemical physics* 21.6 (1953), p. 1087-1092.

[124] Satoshi MORITA et Hidetoshi NISHIMORI. « Mathematical foundation of quantum annealing ». In : *Journal of Mathematical Physics* 49.12 (2008), p. 125210.

[125] Jack MYERS. « The Current State and Potential of Quantum Computing ». In : ().

[126] Florian NEUKART et al. « Traffic flow optimization using a quantum annealer ». In : *Frontiers in ICT* 4 (2017), p. 29.

[127] Hartmut NEVEN et al. « QBoost : Large Scale Classifier Training withAdiabatic Quantum Optimization ». In : *Asian Conference on Machine Learning*. 2012, p. 333-348.

[128] Hartmut NEVEN et al. « Training a large scale classifier with the quantum adiabatic algorithm ». In : *arXiv preprint arXiv :0912.0779* (2009).

[129] Dennis M NEWNS et Chang C TSUEI. *Quantum computing with d-wave superconductors*. US Patent 6,495,854. 2002.

[130] Michael A NIELSEN et Isaac CHUANG. *Quantum computation and quantum information*. 2002.

[131] Michael A NIELSEN et Isaac L CHUANG. « Quantum computation and quantum information ». In : *Phys. Today* 54.2 (2001), p. 60.

[132] A. NOLTE et R. SCHRADER. « Simulated annealing and its problems to color graphs ». In : *Algorithms—ESA 96*. T. 1136. Lecture Notes in Computer Science. Springer, 1996, p. 138-151.

[133] Bernhard ÖMER. « Structured quantum programming ». In : (2003).

[134] Roman ORUS, Samuel MUGEL et Enrique LIZASO. « Quantum computing for finance : overview and prospects ». In : *Reviews in Physics* 4 (2019), p. 100028.

[135] Anargyros PAPAGEORGIOU et Joseph F TRAUB. « Measures of quantum computing speedup ». In : *Physical Review A* 88.2 (2013), p. 022316.

[136] Jennifer PAYKIN, Robert RAND et Steve ZDANCEWIC. « QWIRE : a core language for quantum circuits ». In : *ACM SIGPLAN Notices* 52.1 (2017), p. 846-858.

[137] Edwin PEDNAULT et al. *Breaking the 49-Qubit Barrier in the Simulation of Quantum Circuits*. 2017. arXiv : 1710.05867 [quant-ph].

[138] Edwin PEDNAULT et al. *Leveraging Secondary Storage to Simulate Deep 54-qubit Sycamore Circuits*. 2019. arXiv : 1910.09534 [quant-ph].

[139] WangChun PENG et al. « Factoring larger integers with fewer qubits via quantum annealing with optimized parameters ». In : *SCIENCE CHINA Physics, Mechanics & Astronomy* 62.6 (2019), p. 60311.

[140] John PRESKILL. « Quantum computing and the entanglement frontier ». In : *arXiv preprint arXiv :1203.5813* (2012).

[141] John PRESKILL. « Quantum Computing in the NISQ era and beyond ». In : *Quantum* 2 (2018), p. 79.

[142] Kristen L. PUDENZ et Daniel A. LIDAR. « Quantum adiabatic machine learning ». In : *Quantum Information Processing* 12.5 (2012), 2027–2070. ISSN : 1573-1332. DOI : 10.1007/s11128-012-0506-4.

[143] JF PUGET. *D-Wave vs. CPLEX comparison, Part 1, Part 2, Part 3*.

[144] Ran RAZ et Avishay TAL. « Oracle separation of BQP and PH ». In : *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing*. 2019, p. 13-23.

[145] B. E. REICHARDT. « The quantum adiabatic optimization algorithm and local minima ». In : *ACM STOC*. 2004, p. 502-510.

[146] Jérémie ROLAND et Nicolas J. CERF. « Quantum search by local adiabatic evolution ». In : *Physical Review A* 65.4 (2002). ISSN : 1094-1622. DOI : 10.1103/physreva.65.042308.

[147] Troels F RØNNOW et al. « Defining and detecting quantum speedup ». In : *Science* 345.6195 (2014), p. 420-424.

[148] Martin RÖTTELER. « Quantum algorithms : A survey of some recent results ». In : *Informatik-Forschung und Entwicklung* 21.1-2 (2006), p. 3-20.

[149] G. E. SANTORO et al. « Theory of Quantum Annealing of Spin Glass ». In : *Science* 295 (2016), p. 2427-2430.

[150] G. H. SASAKI et B. HAJEK. « The time complexity of maximum matching by simulated annealing ». In : *Journal of the ACM* 35 (1988), p. 387-403.

[151] J. SCHAUER. « Asymptotic behavior of the quadratic knapsack problems ». In : *European Journal of Operational Research* 255 (2016), p. 357-363.

[152] Erwin SCHRÖDINGER. « Discussion of probability relations between separated systems ». In : *Mathematical Proceedings of the Cambridge Philosophical Society*. T. 31. 4. Cambridge University Press. 1935, p. 555-563.

[153] Benjamin SCHUMACHER. « Quantum coding ». In : *Physical Review A* 51.4 (1995), p. 2738.

[154] Alex SELBY. *D-Wave : comment on comparison with classical computers*. 2013.

[155] Peter SELINGER. « Towards a quantum programming language ». In : *Mathematical Structures in Computer Science* 14.4 (2004), p. 527-586.

[156] Seung Woo SHIN et al. « How" Quantum" is the D-Wave Machine ? » In : *arXiv preprint arXiv :1401.7087* (2014).

[157] Peter W SHOR. « Algorithms for quantum computation : Discrete logarithms and factoring ». In : *Foundations of Computer Science, 1994 Proceedings., 35th Annual Symposium on*. Ieee. 1994, p. 124-134.

[158] Peter W SHOR. « Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer ». In : *SIAM review* 41.2 (1999), p. 303-332.

[159] Patrick SIARRY et al. « Enhanced simulated annealing for globally minimizing functions of many-continuous variables ». In : *ACM Transactions on Mathematical Software (TOMS)* 23.2 (1997), p. 209-228.

[160] Vadim N SMELYANSKIY et al. « A near-term quantum computing approach for hard computational problems in space exploration ». In : *arXiv preprint arXiv :1204.2821* (2012).

[161] R. D. SOMMA et S. BOIXO. « Spectral Gap Amplification ». In : *SIAM Journal on Computing* 42.2 (2013), 593–610. ISSN : 1095-7111. DOI : 10.1137/120871997.

[162] Andriyan Bayu SUKSMONO et Yuichiro MINATO. *Finding Hadamard matrices by a quantum annealing machine*. 2019. arXiv : 1902.07890 [quant-ph].

[163] Krysta M. SVORE, Matthew B. HASTINGS et Michael FREEDMAN. *Faster Phase Estimation*. 2013. arXiv : 1304.0741 [quant-ph].

[164] Mario SZEGEDY. « Quantum speed-up of Markov chain based algorithms ». In : *45th Annual IEEE symposium on foundations of computer science*. IEEE. 2004, p. 32-41.

[165] Olawale TITILOYE et Alan CRISPIN. « Quantum annealing of the graph coloring problem ». In : *Discrete Optimization* 8.2 (2011), p. 376-384.

[166] V Vargomax VARGOMAX. « Generalized Super Mario Bros. is NP-complete ». In : *Proceedings of the 6th Binarennial Workshop about Symposium on Robot Dance Party of Conference in Celebration of Harry Q. Bovik's 0x40th Birthday (SIGBOVIK 2007)*. 2007, p. 87-88.

[167] Umesh VAZIRANI. « A survey of quantum complexity theory ». In : *Proceedings of Symposia in Applied Mathematics*. T. 58. 2002, p. 193-220.

[168] D. VERT, R. SIRDEY et S. LOUISE. *Revisiting old combinatorial beasts in the quantum age : quantum annealing versus maximal matching*. Rapp. tech. 1910.05129. arXiv (quant-ph), 2019.

[169] Daniel VERT, Renaud SIRDEY et Stephane LOUISE. « On the limitations of the chimera graph topology in using analog quantum computers ». In : *Proceedings of the 16th ACM International Conference on Computing Frontiers*. ACM. 2019, p. 226-229.

[170] James D. WHITFIELD, Jacob BIAMONTE et Alán ASPURU-GUZIK. « Simulation of electronic structure Hamiltonians using quantum computers ». In : *Molecular Physics* 109.5 (2011), 735–750. ISSN : 1362-3028. DOI : 10.1080/00268976.2011.552441.

[171] A Peter YOUNG, Sergey KNYSH et Vadim N SMELYANSKIY. « Size dependence of the minimum excitation gap in the quantum adiabatic algorithm ». In : *Physical review letters* 101.17 (2008), p. 170503.

[172] Stefanie ZBINDEN et al. « Embedding Algorithms for Quantum Annealers with Chimera and Pegasus Connection Topologies ». In : *International Conference on High Performance Computing*. Springer. 2020, p. 187-206.

[173] Paolo ZULIANI. « Compiling quantum programs ». In : *Acta Informatica* 41.7-8 (2005), p. 435-474.

université PARIS-SACLAY

ÉCOLE DOCTORALE
Sciences et technologies de l'information et de la communication (STIC)

Titre : Etude des performances des machines à recuit quantique pour la résolution de problèmes combinatoires

Mots clés : Optimisations, ordinateur quantique adiabatique, Graphe d'interconnexion des qubits

Résumé : La principale contribution de cette thèse est d'étudier expérimentalement le comportement des ordinateurs quantiques analogiques tels que ceux commercialisés par D-Wave lorsqu'ils sont confrontés à des cas de problèmes de couplage biparti de cardinalité maximale spécifiquement conçus pour être difficiles à résoudre au moyen d'un recuit simulé. Nous comparons un "Washington" (2X) de D-Wave avec 1098 qubits utilisables sur différentes tailles d'instances et nous observons que pour tous ces cas, sauf les plus triviaux, la machine ne parvient pas à obtenir une solution optimale. Ainsi, nos résultats suggèrent que le recuit quantique, du moins tel qu'il est mis en œuvre dans un dispositif D-Wave, tombe dans les mêmes pièges que le recuit simulé et fournit donc des preuves supplémentaires suggérant qu'il existe des problèmes polynomiaux qu'une telle machine ne peut pas résoudre efficacement pour atteindre l'optimalité. En outre, nous étudions dans quelle mesure les topologies d'interconnexion des qubits expliquent ces derniers résultats expérimentaux.

Title : Study of the performance of quantum annealing machines for solving combinatorial problems

Keywords : Optimization, Adiabatic Quantum Computing, Qubit Interconnection Graphs

Abstract : The main contribution of this thesis is to investigate the behavior of analog quantum computers as commercialized by D-Wave when confronted to instances of the maximum cardinality matching problem which is specifically designed to be hard to solve by means of simulated annealing. We benchmark a D-Wave "Washington" (2X) with 1098 operational qubits on various sizes of such instances and observe that for all but the most trivially small of these it fails to obtain an optimal solution. Thus, our results suggest that quantum annealing, at least as implemented in a D-Wave device, falls in the same pitfalls as simulated annealing and hence provides additional evidences suggesting that there exist polynomial-time problems that such a machine cannot solve efficiently to optimality. Additionally, we investigate the extent to which the qubits interconnection topologies explains these latter experimental results.

Université Paris-Saclay
Espace Technologique / Immeuble Discovery
Route de l'Orme aux Merisiers RD 128 / 91190 Saint-Aubin, France