

UNIVERSITÉ D'ORLÉANS
*Mathématiques, Informatique, Physique Théorique et
Ingénierie des Systèmes (MIPTIS)*
Laboratoire d'Informatique Fondamentale d'Orléans (LIFO)

PHD THESIS Defended by :

Arthur BRAIDA

defended on **June 20, 2024**

to obtain the title of : **Docteur de l'Université d'Orléans**

Specialty : **Quantum Computing**

**Analog Quantum Computing for NP-Hard
Combinatorial Graph Problems**

Calcul Analogique Quantique pour des Problèmes Combinatoires
sur Graphes NP-difficile

JURY :

M. Ioan Todinca

M. Simon Martiel

M. Tameem Albash

Professor, Université d'Orléans, *supervisor*

PhD, IBM Quantum, *supervisor*

Research Associate Professor, University of New
Mexico & Principal Scientist, Sandia National Lab,
reviewer

M. Jérémie Roland

M. Thomas Ayrat

M. Giuseppe Di Molfetta

Professor, Université Libre de Bruxelles, *reviewer*

PhD, Eviden QuantumLab

Professor, Université Aix-Marseille, *jury chairman*

Acknowledgments

Tout d'abord, je souhaite remercier Simon. Merci de m'avoir pris en stage, de m'avoir accompagné et soutenu pour que je transforme ce stage en une thèse. Je sais que cela n'aurait pas été possible sans toi. Nous nous sommes aventurés dans une recherche exploratoire, ce qui me convenait très bien. Merci d'avoir eu confiance en moi pour ce projet qui n'était pas ton expertise. Au-delà de cela, j'ai énormément appris à tes côtés, et je reste très impressionné par tes capacités d'abstraction. Je suis content que tu aies trouvé une position qui te convient. Ioan, je ne sais pas à quoi tu t'attendais quand nous t'avons approché avec Simon, mais ce qui a compté en premier pour moi, c'est le rapport humain et honnête que nous avons eu. J'ai ensuite découvert l'immense chance que j'ai eue de tomber sur toi. Tu as su diriger une thèse éloignée de tes sujets avec légèreté tout en trouvant les mots justes pour m'aider durant ces trois années. Merci pour ta franchise et ta sympathie. Beaucoup d'élèves ont de la chance de t'avoir comme professeur. Globalement, merci à vous deux pour m'avoir poussé et encourager à donner le meilleur de moi-même et à atteindre une rigueur technique et scientifique que vous défendez.

Many thanks to Tameem Albash and Jérémie Roland for reviewing my manuscript. Asking recognized researchers in the field was a bet, and I was very honored that you accepted this task. Thank you for your pertinent comments and for the conversation we had. I would also like to thank Thomas Ayrat and Giuseppe di Molfetta for agreeing to be part of my jury.

Je souhaite également remercier l'ANRT pour m'avoir accordé les financements nécessaires pour mener à bien cette thèse CIFRE encadrée par Eviden et l'Université d'Orléans. Cela m'amène à remercier en premier lieu l'équipe QuantumLab d'Eviden. Merci à toute l'équipe pour votre accueil et votre disponibilité sans limites, des discussions au RIE et au bar, vous êtes nombreux à m'avoir marqué. Ensuite, merci à l'équipe du LIFO de m'avoir si bien intégré malgré ma faible présence. Merci à tous ceux avec qui j'ai pu interagir et qui m'ont facilité énormément de tâches, notamment Isabelle.

Enfin, à ma famille et mes amis, vous ne vous en rendez peut-être pas compte, mais vous êtes ce que j'ai de plus précieux. Vous contribuez à mon bien-être au quotidien par tous les échanges et partages riches en contenu et en émotions. J'ai énormément de chance d'être aussi bien entouré. Particulièrement, papa, maman, merci beaucoup de nous avoir rendu la vie aussi facile pour que tout nous soit possible. Frère, continue comme ça, tu fais plaisir. Merci beaucoup Caroline de m'avoir accompagné et énormément aidé pendant les deux mois de rédaction pour que tout se passe le plus aisément possible. Ceci va de paire avec toute la gratitude que j'ai envers Cissine et Ita pour leur accueil chaleureux à la vieille Enseigne. J'ai eu le cadre rêvé pour un des moments les plus compliqués de la thèse. Nonna, merci beaucoup pour toutes tes pensées et attention positives, ainsi que pour ton savoir-

faire culinaires inégalés que j’essaie petit à petit de récupérer. Bien sûr aussi à toute ma famille Braidà et Valerio avec laquelle je continue de passer de moments très agréables. Bibi, merci pour tout ce que nous avons vécu ces presque deux dernières années et tout ce que nous allons vivre à partir de maintenant. Merci pour tous les efforts que tu fais pour me supporter, merci d’avoir confiance en moi.

Benjamin, tu as commencé cette aventure avec moi, même si je n’étais qu’en stage. Le bonheur quotidien que nous avons partagé n’a pas de valeur et m’a permis de donner le meilleur de moi-même et de me lancer dans ce beau projet, merci pour tout. Vous êtes tellement nombreux à avoir contribué à ce résultat aujourd’hui que je ne pourrais citer tout le monde : Theo, Thomas, Salim, Antoine, on s’est bien régala à la colocaspoon et que ça continue, je suis tellement content de continuer à vous voir, Gab et Max et cette belle discussion Nirvana sur la conscience, j’adore tout simplement, Briec et Vincent, un régal de vous avoir rencontrés, Gasp, Xence et Arthur, des amitiés qui ne s’estompent pas, tous ceux de ma prépa, mon école, de Londres et d’autres relations incroyables que je n’ai pu citer mais avec qui j’ai beaucoup partagé, encore merci. Et puis la team d’Orlinz, la coloc à 45000 vitesses, vous avez contribué plus que tout à mon année au bord de la Loire, merci aussi à mes partenaires de squash, grâce à vous j’ai pu progresser auprès de joueurs expérimentés, merci de m’avoir intégré à votre groupe. À tous ceux qui m’ont fait l’immense honneur de se déplacer pour ma soutenance et partager cette journée inoubliable, un vrai bonheur.

Contents

List of Acronyms	vii
Introduction en français	1
1 Introduction	13
1.1 History of computing	13
1.2 From quantum mechanics to quantum computation	15
1.3 Theoretical foundation of analog quantum computing	20
1.4 Contribution of this thesis	25
2 Preliminaries	29
2.1 Theoretical foundation of analog quantum computing	30
2.1.1 Complexity theory	30
2.1.2 Adiabatic quantum computing	32
2.1.3 Quantum annealing	36
2.1.4 Hardware and quantum annealer	37
2.2 Hamiltonians definition for graphs combinatorial problems	38
2.2.1 General optimization on graphs	38
2.2.2 Maximum Cut	40
2.2.3 Maximum Independent Set	42
2.2.4 Maximum weight k -clique problem	43
2.3 Approximation of MaxCut	46
2.3.1 Quantum Approximate Optimization Algorithm (QAOA)	46
2.3.2 Classical local algorithms	49
2.3.3 Overview of few results	50
2.4 Avoided level crossings (AC)	52
2.4.1 Perturbation theory	52
2.4.2 Initial description of an Anti-crossing	54
2.4.3 Perturbative crossings	55
2.4.4 Choi's crossing definition	58
2.5 Lieb-Robinson bound	60
2.5.1 Observation and intuition	60
2.5.2 Mathematical derivation	62
2.5.3 Some known applications	64
I Anti-crossings in Adiabatic Quantum Computing	65
3 Anti-crossings characterization	67
3.1 Results and limitations of Def. 2.6	67

3.1.1	Minimum gap in AC of Def. 2.6	68
3.1.2	AC in Maximum-Weight k -clique problem	71
3.2	New characterization of anti-crossing	75
3.2.1	Variation of $ \phi_0(s)\rangle$ at s^*	75
3.2.2	New anti-crossing definition	77
3.2.3	Thorough comparison of ACs definitions	78
3.3	Limitation and other potential type of AC	82
4	Exponentially closing gaps in AQC	85
4.1	Perturbative Analysis to AQC	85
4.1.1	Initial perturbation	89
4.1.2	Final perturbation	90
4.1.3	Energy crossing	91
4.2	Application to MaxCut over bipartite graphs	93
4.2.1	d -regular bipartite graphs	93
4.2.2	General bipartite graphs	99
4.2.3	Numerical study: AC and other observations	103
4.2.4	Validation of perturbative expansion and limitations	106
4.3	Higher order perturbative expansion for MaxCut	108
II	Approximation ratio in short-time Quantum Annealing	115
5	Direct Lieb-Robinson type approach	117
5.1	Locality in short-time quantum annealing	117
5.1.1	Locality and restriction to regular graph	118
5.1.2	Lieb-Robinson like bound	121
5.2	Two different problem applications	124
5.2.1	QA approximation of MaxCut	124
5.2.2	QA approximation of MIS	125
5.3	Toward a better bound for MaxCut?	127
5.3.1	(Non) Tightness of our LR bound	128
5.3.2	Hints for better approximation ratios	129
6	Tight Lieb-Robinson bound for approximation ratio	133
6.1	Local analysis of QA and Lieb-Robinson Bound	134
6.1.1	Parametrized QA	134
6.1.2	p -local analysis	135
6.1.3	Lieb-Robinson bound reduction	137
6.2	Tight LR bound on regular graphs for MaxCut approximation	139
6.2.1	Tight LR bound on regular graphs in QA	139
6.2.2	Global LR bound	144
6.2.3	Application to approximation ratio of MaxCut	146
6.3	Discussion	149
6.3.1	Tightness of the analysis method	149

6.3.2	Toward better ratios	151
6.3.3	Directions for generalizing the construction	152
7	Effective approximation of MaxCut	155
7.1	Toward a state-dependent LR bound	155
7.1.1	Exact expansion of an edge energy	156
7.1.2	State-dependent LR bound	159
7.2	Double binary tree simulation and optimal approximation ratio . . .	161
7.2.1	Symmetry in T_p for dimensional reduction	161
7.2.2	Optimal approximation ratio for QA?	165
	Conclusion & Outlooks	167
	Appendices	171
A	Some interesting features of the k-clique problem	173
A.1	Influence of the driver graph	173
A.2	Some property of H_0	175
A.3	Lower bound on the minimum gap	176
B	MaxCut H_1 on H_0 eigenvectors	179
B.1	Proof of Proposition 4.1	179
C	Technical results for the LR bound based ratio	183
C.1	Minimization of Equation 6.6	183
C.2	Proof of Equation 6.10	184
C.3	Nested integrals	184
	Bibliography	191

List of Acronyms

AC Avoided level-crossing	16
AQC Adiabatic Quantum Computation	20
LR Lieb-Robinson	24
MaxCut Maximum Cut	18
MIS Maximum Independent Set	33
NISQ Near Intermediate Scale Quantum	23
ODE Ordinary Differential Equation	164
QA Quantum Annealing	19
QAOA Quantum Approximation Optimization Algorithm	23
QUBO Quadratic Unconstrained Binary Optimization	22

Introduction en français

Histoire du calcul

Calculus signifie caillou en latin [Larousse 2023]. Les cailloux ont été la première unité de comptage utilisée par les bergers pour compter les moutons qui entraient et sortaient de la bergerie. Au retour d'une sortie, le nombre de cailloux restants indiquait le nombre de moutons manquants. Ce système a également été utilisé pendant les conflits pour compter le nombre de soldats qui ne revenaient pas de la bataille. Ce système de comptage a rapidement été dépassé par la nécessité d'effectuer des opérations plus complexes, nécessaire pour le commerce par exemple. Vers 2700-2300 avant J.-C., le premier mécanisme de type boulier est inventé. Il est conçu comme une extension des doigts de la main pour permettre de compter jusqu'à plus de 10. À l'époque, on comptait en base 5. Le boulier (Fig. 1.1 (a)) est considéré comme la première version d'une calculatrice. Ces calculs se limitaient encore à des opérations arithmétiques. Plusieurs siècles plus tard, en 1050-771 av. J.-C., dans la Chine ancienne, les voyageurs utilisaient un "Chariot pointant le Sud" pour indiquer le sud au cours d'un voyage. Le mécanisme était basé sur un engrenage différentiel mis en place au début du voyage. Il s'agit du premier calcul analogique enregistré. En effet, à chaque virage, le chariot calcule la différence de rotation des deux roues opposées pour la compenser en tournant le pointeur vers l'orientation initiale. Le premier mécanisme considéré comme un ordinateur est le mécanisme d'Antikythera (Fig. 1 (b)) dans la Grèce antique, datant de 100 avant J.-C. [Efsthathiou & Efsthathiou 2018]. Il a été conçu pour calculer les positions astronomiques à l'aide d'engrenages différentiels. Plusieurs engrenages étaient en jeu : l'un d'eux comptait 365 dents pour les jours de l'année, un autre contrôlait le nombre de lunaisons, et ainsi de suite. Pour l'utiliser, il fallait indiquer la date du jour, et en tournant les engrenages, il donnait la date et les positions de la Lune et du Soleil pour la prochaine éclipse. Ce calcul utilise la rotation mécanique des engrenages comme processus de calcul, les données d'entrée et de sortie étant codées dans la position des engrenages.

Ce type d'ordinateur, qui utilise intelligemment une variation continue d'un dispositif physique, c'est-à-dire un signal analogique, est appelé ordinateur analogique. Un exemple de base de calcul analogique est l'opération de multiplication. Connaissant la loi d'Ohm, $U = RI$ où la tension U est égale à la résistance R multipliée par le courant I , nous pouvons calculer la multiplication entre deux entrées que nous encodons dans R et I . En mesurant U , nous obtenons la multiplication de nos entrées encodées. Ce n'est que vers 1150 qu'est apparu le premier ordinateur analogique "programmable", appelé l'"Horloge Éléphant" inventée par les Arabes [Al-Jazari 1974]. Ici, "programmable" signifie que le mécanisme pouvait être réglé avant d'être mis en marche. Il repose sur la vitesse d'écoulement continues de

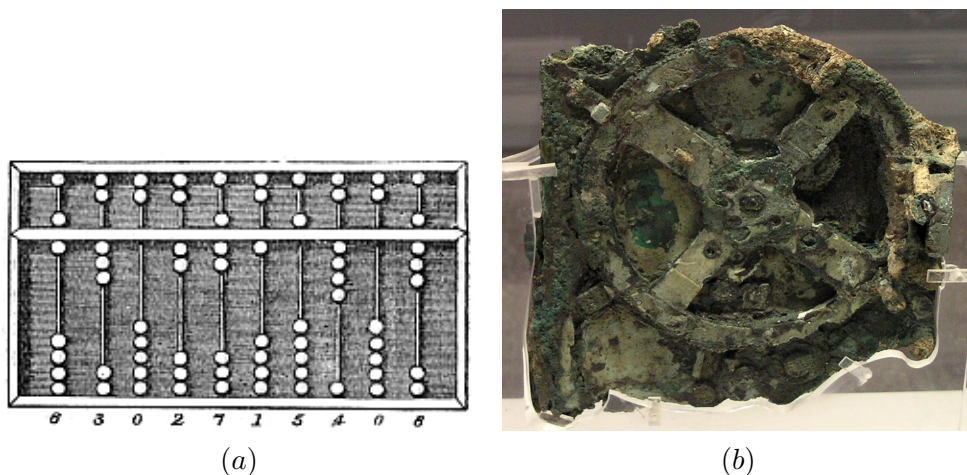


Figure 1: (a) Photo d'un vieux boulier [Baynes 1875] et (b) fragment du mécanisme d'Antikythera de [Wikipedia 2005].

l'eau pour donner le pourcentage de temps diurne écoulé. Bien que les ordinateurs numériques aient commencé à apparaître à la fin du 18^{ème} siècle, à cette époque les ordinateurs analogiques étaient encore plus puissants, comme en témoigne la “machine à prévoir les marées” capable, entre autre, d'effectuer une transformée de Fourier [Wikipedia 2024].

Nous remarquons que tous les systèmes semblables à des ordinateurs mentionnés ci-dessus sont conçus pour effectuer une tâche spécifique, comme compter, indiquer une direction ou donner l'heure. Nous appelons *ordinateur à usage général ou universel* un ordinateur capable de résoudre n'importe quel calcul au sens moderne du terme, c'est-à-dire un ordinateur Turing-complet. Nous comprenons que pour construire un ordinateur universel, nous devons disposer d'une sorte d'ensemble d'opérations de base qui peuvent être utilisées comme éléments constitutifs d'un calcul plus complexe. Ces opérations sont nécessairement basées sur le calcul analogique et doivent également interagir les unes avec les autres. La quantité physique connue la plus pratique pour effectuer un calcul est le signal électrique.

Jusque dans les années 60, l'une des dernières utilisations du calcul analogique a été la simulation de systèmes mécaniques complexes, comme le système de suspension d'une voiture avec deux masses couplées à des ressorts et un amortisseur. [Ulmann 2008]. Vers 1950-1960, les ordinateurs numériques ont surpassé les ordinateurs analogiques [Wikipedia 2024]. Le transistor utilisé comme élément de base est devenu suffisamment puissant pour effectuer tout calcul complexe plus efficacement que les ordinateurs analogiques. Dans un ordinateur numérique, l'information n'est plus représentée par une quantité physique continue, mais par une valeur discrète. Bien entendu, le matériel tel que les transistors utilise toujours des variables physiques continues telles que la tension électrique. Une abstraction supplémentaire permet de faire le lien avec l'information numérisée : par exemple, une tension inférieure à $1/3$ d'une certaine unité représente la valeur binaire '0', et une tension

supérieure à cette valeur représente ‘1’. Cette abstraction et l’universalité des ordinateurs numériques les rendent plus versatiles et plus précis que les ordinateurs analogiques pour effectuer des tâches élaborées.

Au 21^{ème} siècle, il reste très peu de cas de calcul analogique, les plus connus étant la montre mécanique et l’ordinateur de vol. Avec l’essor de l’intelligence artificielle, de nouveaux prototypes d’ordinateurs analogiques sont développés dans les laboratoires d’informatique neuromorphique [James *et al.* 2017]. L’avantage de l’informatique analogique est son efficacité énergétique et sa rapidité. Les algorithmes d’intelligence artificielle ne sont pas conçus pour avoir des valeurs exactes, ce qui fait des ordinateurs analogiques des candidats prometteurs pour accélérer, avec moins d’énergie, l’exécution de ces algorithmes.

Quoi qu’il en soit, pour calculer quelque chose, nous devons nous appuyer sur du matériel physique et sur un encodage de l’information. D’une part, l’encodage dépend du choix entre le calcul numérique ou analogique. D’autre part, le dispositif physique est soumis aux lois de la physique sous-jacente en jeu : pour les dispositifs mécaniques, il suit les lois fondamentales de la dynamique, pour les dispositifs électriques, il suit les lois fondamentales de l’électromagnétisme. Selon les lois en question, les limitations ne sont pas les mêmes. Jusqu’à aujourd’hui, les lois *classiques* de l’électromagnétisme ont dominé son utilisation pour le calcul. Au milieu des années 1920, les lois de la mécanique quantique ont été découvertes. Ce n’est que dans les années 1980 que l’idée d’un ordinateur quantique est née [Benioff 1980, Manin 1980, Feynman 1982]. Comme pour les ordinateurs classiques, il existe des versions quantiques analogiques et numériques. Toutes deux sont basées sur l’équation maîtresse de la physique quantique. Pour accéder à l’informatique quantique, nous devons développer un dispositif physique quantique qui peut être “facilement” contrôlé. Dans la section suivante, nous retracerons l’histoire de la découverte de la mécanique quantique jusqu’à son utilisation en tant que physique sous-jacente de l’informatique.

De la mécanique quantique au calcul quantique

La découverte des lois fondamentales de la mécanique quantique est attribuée à Erwin Schrödinger en 1926 [Schrödinger 1926]. Dans ses travaux, Schrödinger s’est fortement appuyé sur la thèse de Louis de Broglie, qui a été le premier à postuler que le rayonnement pouvait être à la fois ondulatoire et corpusculaire [de Broglie 1924]. Au départ, ni de Broglie ni Schrödinger ne semblaient vraiment croire à cette hypothèse particulière sur la matière. Cette hypothèse contre-intuitive leur a permis d’obtenir des résultats intéressants pour la compréhension de l’atome.

En thermodynamique classique, nous savons qu’au cours d’un processus adiabatique, certaines propriétés physiques telles que l’énergie ou l’entropie sont conservées [Planck 1903]. Nous appelons processus adiabatique une évolution durant laquelle les conditions externes varient suffisamment lentement. En 1928, Born et Fock ont démontré la première version mathématique du théorème adiabatique en physique

quantique afin de mieux comprendre la stabilité des orbitales électroniques dans les atomes [Born & Fock 1928]. Ils ont prouvé que le niveau d'énergie d'un système quantique est conservé en cas de variation lente du champ électromagnétique. En cas de variation rapide, il peut passer à un niveau d'énergie différent, appelé *état excité*. Illustrons (Fig. 2) ce résultat important, car il constitue le fondement théorique du calcul quantique adiabatique (voir Sec. 1.3). Imaginons la situation suivante : Notre objectif est de transporter un bébé de son domicile à un endroit (pour l'instant non défini). Au départ, le bébé est endormi et est transporté dans une poussette. Nous savons que si la personne qui pousse la poussette va lentement, il y a une forte probabilité que le bébé reste endormi. En revanche, si la personne qui pousse la poussette va trop vite, il y a de fortes chances que le bébé se réveille, commence à pleurer et atteigne l'état d'excitation. Le paramètre externe qui change est la vitesse de la poussette, et son environnement extérieur est défini par sa position dans l'espace. Pour un système quantique, l'environnement extérieur est défini par un objet appelé l'Hamiltonien H du système. Pour qu'une évolution soit adiabatique, la variation de H au cours du temps doit être suffisamment lente.

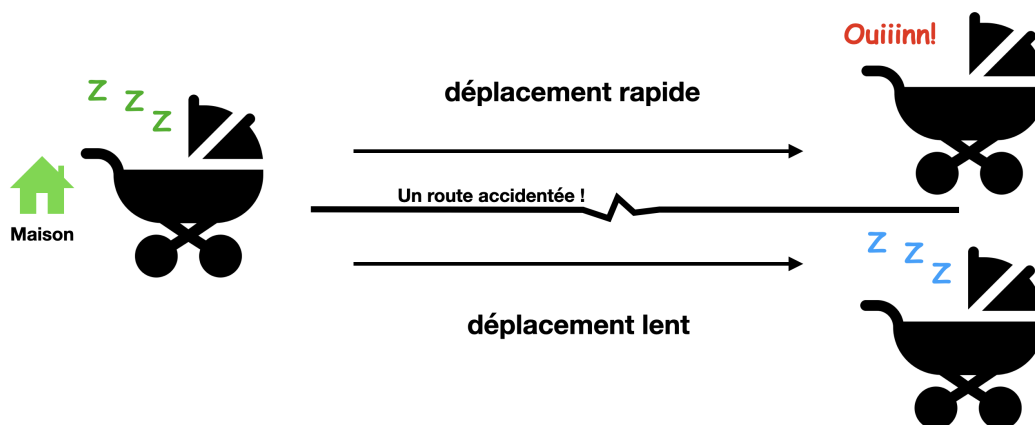


Figure 2: Une image schématique pour expliquer le théorème adiabatique (inspirée par Pauline Besserve). Elle ne vise pas à rendre compte de l'ensemble de la physique sous-jacente. Le bébé endormi représente l'état de plus basse énergie et le bébé qui pleure représente un état excité du système quantique.

Le théorème adiabatique soulève quelques questions intéressantes. Tout d'abord, qu'entend-on par “lent” ? La vitesse de l'évolution est liée à l'inverse du gap Hamiltonien. Le gap est la différence entre les niveaux d'énergie à chaque instant de l'évolution. Pour répondre à cette question sur la validité du théorème adiabatique, von Neumann et Wigner ont développé un résultat appelé “règle de non-croisement”. Cette règle stipule que, sous certaines conditions relatives à l'Hamiltonien, deux niveaux d'énergie ne se croisent jamais réellement lorsque l'Hamiltonien est modifié, mais qu'ils peuvent s'approcher de très près. Le phénomène qui se produit lorsqu'ils se rapprochent suffisamment pour que l'on puisse penser qu'ils se croisent est ap-

pelé *croisement-évité*, que nous définirons formellement au chapitre 2, Sec. 2.4. Le gap atteint donc un minimum, que nous notons $\Delta_{\min}$. Par conséquent, dans le régime de temps infini, le système quantique est assuré de rester au même niveau que celui à partir duquel il a commencé son évolution. Dans notre illustration, cela signifie que le théorème garantit que le bébé restera endormi si la poussette se déplace infiniment lentement. La question est la suivante : que se passe-t-il si l'évolution est plus rapide que la condition adiabatique ? En 1932, Landau et Zener ont calculé indépendamment la probabilité que le système quantique passe d'un niveau d'énergie à l'autre. Ces transitions sont appelées transitions non adiabatiques. En clair, plus le gap est petit, plus la probabilité de saut est grande [Landau 1932, Zener 1932]. La meilleure analogie avec le point de gap minimum dans notre exemple serait la position de la poussette où le bébé pourrait facilement se réveiller, par exemple lorsque la route est accidentée. Les parties bosselées de la route doivent être traversées encore plus lentement que les parties plates. Il arrive que ces passages cahoteux apparaissent plusieurs fois au cours de l'évolution. Dans le travail de [Wilkinson 1987, Wilkinson 1989], l'auteur a entrepris d'abord une description des croisement-évités, puis une étude statistique de leur nombre.

La première version du théorème adiabatique impose de fortes contraintes sur l'Hamiltonien, ce qui limite ses applications potentielles. Une version mathématique plus générale et plus rigoureuse a été dérivée par [Kato 1950] au Japon. En particulier, il s'est débarrassé de deux hypothèses sur le spectre de l'Hamiltonien. Dans sa version, ce dernier peut être continu et présenter des dégénérescences. À la même époque, dans [Anderson 1958], l'auteur démontre un phénomène physique se produisant dans un système quantique évoluant selon l'équation de Schrödinger. Il affirme que, dans certaines conditions, aucune diffusion n'a lieu dans un réseau quantique alors qu'elle était attendue. Plus simplement, il suggère qu'un état quantique peut se localiser dans une région spécifique avant que l'évolution ne se termine. Cette localisation peut se produire dans un "mauvais" niveau d'énergie - par exemple, dans cette thèse, la principale conséquence de ce phénomène est qu'à un moment intermédiaire de l'évolution, l'état d'énergie le plus bas est proche de l'état excité du système à la fin de l'évolution (Fig. 3).

Pour comprendre les différents événements que peut subir un système quantique, il faut également mentionner qu'il peut expérimenter des transitions de phase. Ces transitions peuvent être du premier ou du second ordre, selon la continuité d'évolution de l'état au moment de la transition. Nous renvoyons le lecteur à [Sachdev 1999] pour plus de détails. Un autre résultat théorique important pour comprendre l'évolution quantique est la limite de Lieb-Robinson. De manière informelle, elle indique que l'information voyage à une vitesse finie dans un système quantique [Lieb & Robinson 1972]. Cela signifie, entre autres, qu'il existe une évolution de durée totale T telle qu'il n'y a pas de corrélation entre deux régions suffisamment éloignées au sein du système quantique. La distance "suffisante" est définie par la vitesse de Lieb-Robinson v_{LR} et est égale à $v_{LR}T$. Cette vitesse dépend notamment de la structure du système et de son matériel physique.

Au début des années 70, peu après le début de la recherche en informatique

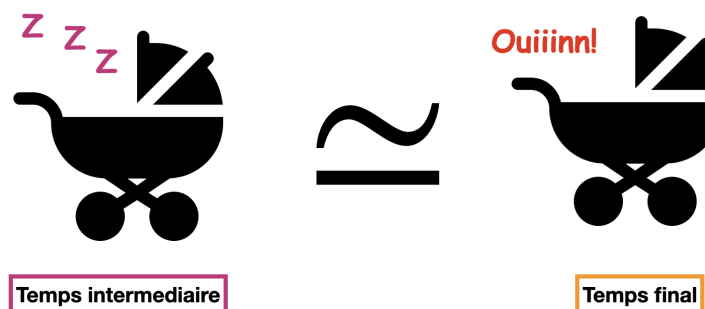


Figure 3: Le phénomène de localisation d’Anderson avec notre illustration. L’état d’énergie le plus bas à un moment intermédiaire est le bébé endormi au milieu du voyage. Ce dernier état est proche au bébé qui pleure à la fin du voyage. Ici le terme “proche” est légèrement tiré par les cheveux et montre une limite de notre analogie.

théorique avec la thèse de Church-Turing, Cook et Levin ont indépendamment prouvé le premier problème combinatoire NP-difficile [Cook 1971, Levin 1973]. Pour l’instant, il suffit de considérer les problèmes NP-difficiles comme des problèmes difficiles qui ne peuvent pas être résolus efficacement. Ce problème est appelé 3SAT pour Satisfiabilité Booléenne sur des clauses de taille trois. Ce problème marque le début de la théorie de la complexité. Ce domaine traite de l’analyse des ressources nécessaires pour calculer certaines fonctions données. En général, il s’agit de ressources temporelles. À la même époque, Karp a prouvé par réduction un zoo de 21 problèmes combinatoires qui sont NP-difficiles [Karp 1972]. Par exemple, le problème de la coupe maximale (MaxCut) est l’un d’entre eux. Illustrons ce problème (Fig. 4).

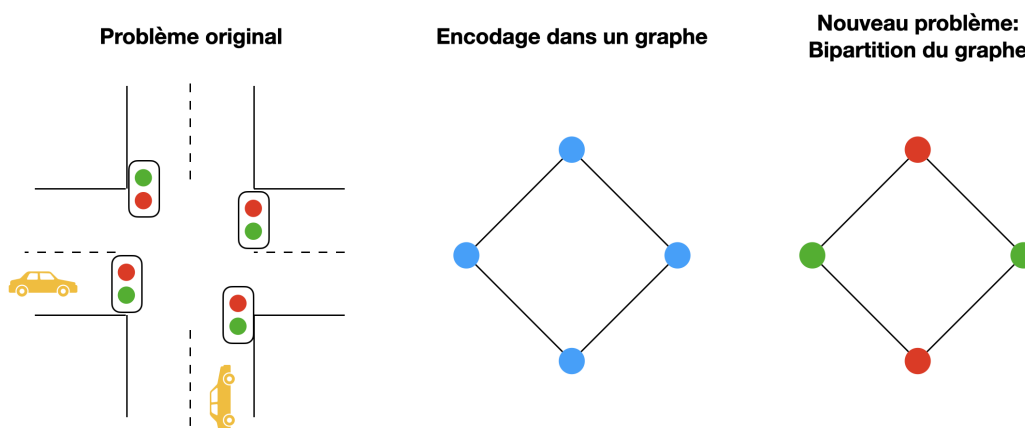


Figure 4: Illustration du problème de la coupe maximale. À gauche, le problème original. Au milieu, son encodage dans une instance de graphe. À droite, une solution possible au problème de la coupe maximale sur le graphe donné. Cette dernière solution est également une solution au problème original.

Imaginons une intersection avec des feux de circulation. L'objectif est d'initialiser les couleurs des feux de circulation afin d'empêcher le plus grand nombre possible d'événements indésirables de se produire. Supposons que nous n'ayons que deux couleurs, le rouge et le vert. Une façon de résoudre ce problème est de l'encoder dans un graphe ; chaque feu de circulation est représenté par un sommet dans le graphe, et nous connectons deux sommets si les feux de circulation correspondants devraient, idéalement, être de couleurs différentes. Le nouvel objectif du problème original est de trouver une bipartition rouge/verte dans le graphe généré. Il s'agit de la coupe maximale, car si nous affichons les sommets verts à droite et les sommets rouges à gauche, toutes les arêtes traversant la bipartition peuvent être coupées. Nous comprenons qu'avec un carrefour plus complexe comme celui de la Fig. 5, trouver la meilleure bicoloration est beaucoup moins simple. Le nombre de combinaisons de couleurs explose exponentiellement avec le nombre n de feux de circulation sous la forme (nombre de couleurs) n .

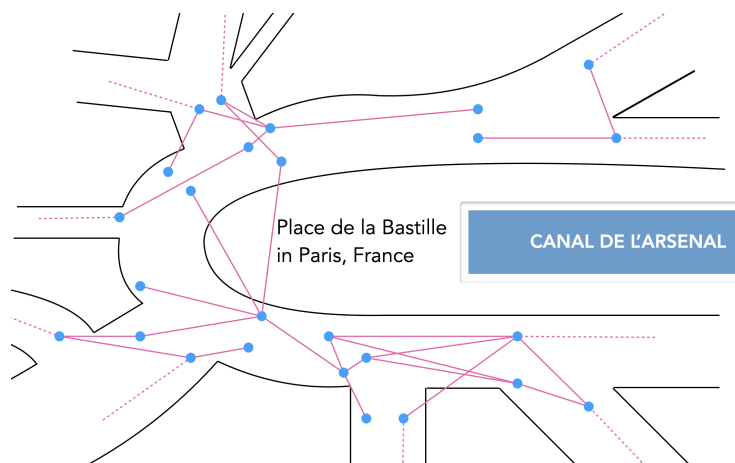


Figure 5: Exemple d'un problème de feux de circulation sur une instance complexe. Lorsque le nombre de feux de circulation augmente, le nombre de combinaisons explose. La meilleure solution pourrait être *frustré*, ce qui signifie que même deux feux de circulation liés ont la même couleur.

Dans cette thèse, nous étudions principalement ce type de problèmes, appelés problèmes combinatoires sur graphes NP-difficile. Nous renvoyons le lecteur à la revue suivante [Barahona *et al.* 1988] sur l'application des problèmes combinatoires. Par exemple, la solution de MaxCut permet de concevoir des circuits électriques et de résoudre le problème d'Ising en physique. La difficulté NP de MaxCut signifie que nous ne connaissons pas d'algorithme dont le temps d'exécution maximum varie de façon polynomiale avec la taille de l'entrée pour résoudre le problème sur un graphe d'entrée arbitraire. Cependant, nous pouvons toujours être intéressés par des solutions approximatives qui peuvent être atteintes efficacement, c'est-à-dire en temps polynomial. De tels algorithmes sont appelés *algorithmes d'approximation*, et nous évaluons leur performance en calculant leur ratio d'approximation. En

bref, le taux d'approximation nous indique à quel point nous sommes proches de la solution optimale. Le meilleur algorithme d'approximation pour MaxCut sur un graphe général est celui de [Goemans & Williamson 1995], qui atteint un ratio d'approximation de 0,87856, c'est-à-dire que cet algorithme garantit que sa solution de sortie coupe plus de 87,856% de toutes les arêtes possibles qui peuvent être coupées dans la solution optimale. Si nous limitons le graphe à des graphes 3-réguliers, c'est-à-dire que chaque sommet est lié à exactement 3 autres, les auteurs de [Halperin *et al.* 2002] ont prouvé que leur algorithme atteint une approximation de 0,93 en temps polynomial.

Pendant cette période, la première idée d'un ordinateur quantique a été lancée en 1982 par Feynman [Feynman 1982]. Il pensait que la meilleure façon de simuler la nature, qui est quantique, était d'utiliser du matériel quantique. L'idée a été formalisée par Deutsch, qui a présenté l'ordinateur quantique universel [Deutsch & Penrose 1985]. Il est fondamentalement similaire à un ordinateur "classique", mais les lois physiques sous-jacentes qui régissent l'appareil sont des lois quantiques. À la fin des années 80, une équipe italienne a eu l'idée d'utiliser les fluctuations quantiques pour résoudre des problèmes combinatoires [de Falco *et al.* 1988, Apolloni *et al.* 1989]. Ils ont introduit l'algorithme du recuit quantique, une métaheuristique générale du même type que le recuit simulé pour minimiser une fonction de coût donnée. Il utilise l'effet tunnel quantique pour surmonter les barrières potentielles, alors que le recuit simulé nécessite une température élevée pour surmonter ces barrières (Fig. 6). Dans [Kadowaki & Nishimori 1998], les auteurs ont fourni la première comparaison rigoureuse du recuit simulé et du recuit quantique.

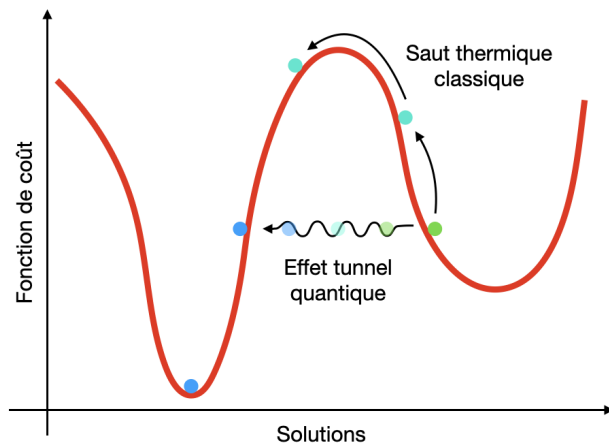


Figure 6: Explication schématique de la physique qui aide l'état à franchir la barrière de potentiel.

Ce n'est que dans les années 1990 que les premiers algorithmes quantiques ont formellement démontré une "accélération quantique" par rapport aux ordinateurs classiques. Cela signifie que les ordinateurs quantiques peuvent résoudre des

problèmes plus efficacement que les ordinateurs classiques [Deutsch & Jozsa 1992, Bernstein & Vazirani 1997]. Il s’agit d’algorithmes numériques, car la propriété discrète rend le cadre plus propice aux démonstrations. Parmi eux, les deux plus célèbres sont l’algorithme de [Shor 1994] pour la décomposition des nombres en facteurs premiers et l’algorithme de [Grover 1996] pour la recherche d’une aiguille dans une botte de foin. Ces travaux ont lancé une course sans précédent vers la réalisation de l’avantage quantique en pratique.

Structure du manuscrit et contributions

Dans la suite de ce manuscrit, nous reprenons cette introduction en anglais en y ajoutant tout les travaux des années 2000 à aujourd’hui qui constituent l’état de l’art duquel la recherche présentée dans cette thèse s’est lancée.

La recherche présentée dans ce manuscrit se base principalement sur des articles publiés ou acceptés dans différentes revues scientifiques [Braidà & Martiel 2021, Braidà *et al.* 2022, Braidà *et al.* 2024a, Braidà *et al.* 2024b]. Nous avons essayé de les laisser tels qu’ils ont été publiés, nous avons modifié certaines notations pour qu’elles restent harmonieuses dans le manuscrit et nous avons ajouté certains travaux non publiés qui sont de nature plus exploratoire, mais qui pourraient fournir des idées et des possibilités futures. Dans le chapitre 2, nous introduisons formellement les notations du calcul quantique analogique et la définition des Hamiltoniens pour les problèmes combinatoires sur graphes. Nous détaillons différents algorithmes pour le problème MaxCut. Ensuite, nous définissons clairement la notion de croisement évités et la borne de Lieb-Robinson.

Sauf indication contraire, tout le contenu des parties I et II relève du travail original du doctorant avec ses superviseurs.

La partie I se concentre sur la compréhension du phénomène de croisement évité dans le calcul quantique adiabatique (CQA). Dans le chapitre 3, nous commençons par la caractérisation récente de [Choi 2020] et prouvons comment sa paramétrisation est liée au gap minimum. Ensuite, en appliquant sa définition à un modèle jouet pour le problème de la k -clique de poids maximum, nous mettons en évidence comment la définition peut manquer la caractérisation d’un croisement évité (Sec. 3.1). Ces observations nous conduisent à proposer une nouvelle définition, plus générale que la précédente (voir Fig. 7 et Sec. 3.2). Elle utilise un développement de Taylor pour justifier notre nouvelle définition et fournit en même temps la démonstration rigoureuse de la description initiale d’un croisement évité (Sec. 2.4.2). Nous validons notre définition par des analyses numériques. Finalement, nous montrons quelques cas de limitation que notre définition ne capturera pas non plus (Sec. 3.3). Dans le chapitre 4, nous prouvons un théorème qui donne une condition d’apparition des croisement-évités sous la validité de l’expansion de perturbation au premier ordre (Sec. 4.1). Nous appliquons ensuite ce théorème au problème MaxCut pour prouver que CQA résout efficacement MaxCut sur des graphes bipartis réguliers et qu’une irrégularité suffisante conduit à un gap se réduisant exponentiellement (Sec. 4.2).

Bien que l'implication générale de cette observation serait de conclure à l'échec de CQA dans ce cas, en utilisant notre nouvelle définition de croisement-évités, une étude numérique suggère que ce n'est probablement pas le cas. Pour de petites instances, la probabilité de mesurer l'état optimal semble atteindre efficacement une valeur constante. Cette observation frappante nous amène à poser la question de la possibilité de prouver un théorème adiabatique relaxé de sa dépendance dans le gap. La dernière section du chapitre (Sec. 4.3) suggère d'autres études intéressantes du problème MaxCut en utilisant la théorie perturbative. Il est tout à fait remarquable que nous puissions analyser différents coefficients du processus quantique à l'aide de la terminologie de théorie des graphes. Cela permettrait-il de prouver des résultats intéressants ?

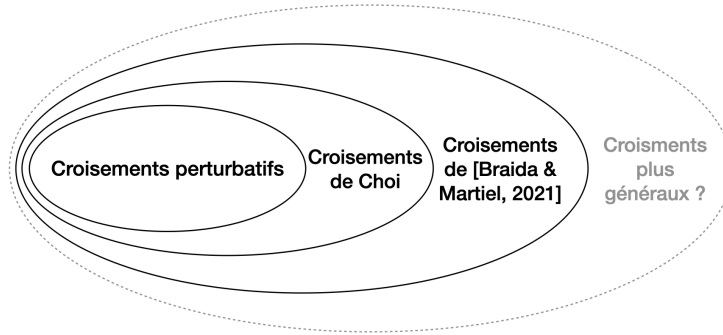


Figure 7: Inclusion des définitions des croisement-évités.

La partie II se concentre sur les potentialités du recuit quantique en tant qu'algorithme d'approximation. Plus précisément, nous nous attaquons à l'étude du recuit quantique courts en temps et constant pour MaxCut sur les graphes cubiques (i.e. 3-régulier). Notre première approche (Chapitre 5) est une adaptation directe de la démonstration du rapport d'approximation de QAOA [Farhi *et al.* 2014]. Nous décrivons d'abord les différentes étapes de la preuve dans le régime du calcul analogique (Sec. 5.1). Ensuite, le travail réside dans le calcul d'une borne de Lieb-Robinson (LR) suffisamment petite à distance 1 pour dériver une borne inférieure non triviale (au-dessus du choix aléatoire) sur le ratio d'approximation lorsque le recuit quantique est étudié comme un algorithme 1-local (0.5933 dans Fig. 1.11 et Sec. 5.2). Une étude de l'optimalité de la borne de type LR utilisée nous conduit à apporter des arguments numériques convaincants selon lesquels le ratio devrait être supérieur à celui de QAOA (0.6925). Nous suggérons un algorithme non implémentable, dans le sens où il est trop coûteux à exécuter, dont le ratio est probablement proche de 0.6963 (Sec. 5.3). Dans le dernier projet détaillé dans le chapitre 6, nous développons une borne de LR super serrée atteignant des valeurs numériques étonnantes nous permettant de réduire le temps nécessaire pour l'algorithme précédent. Avec l'introduction d'une version paramétrée du recuit quantique, nous avons pu prouver qu'il atteint avec une analyse 1-locale un ratio d'approximation supérieur à 0.7020, surpassant tout algorithme 1-local connu pour ce problème (voir Fig. 8 et

Sec. 6.2). Nous discutons de l’optimalité de cette nouvelle borne et prouvons que la méthode employée atteint presque son optimalité (Sec. 6.3). Dans un dernier chapitre (Chapitre 7), nous développons quelques idées sur la localité dans le recuit quantique, et ce qui est nécessaire pour prouver une borne de LR dépendante de l’état. Finalement, nous donnons quelques arguments sur le ratio d’approximation effectif en argumentant sur le pire graphe et en tirant astucieusement parti des symétries des graphes, grâce auxquelles nous sommes capables de simuler rapidement des graphes de grande taille.

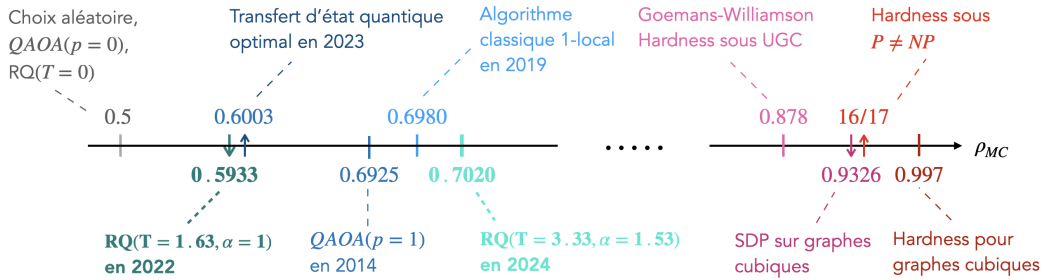


Figure 8: Ratio d’approximation atteint par différents algorithmes que nous détaillons par la suite. En gras sont indiquées nos contributions.

Le dernier chapitre Conclusion et perspectives résume les principaux résultats et suggère quelques pistes de recherche.

Introduction

Contents

1.1	History of computing	13
1.2	From quantum mechanics to quantum computation	15
1.3	Theoretical foundation of analog quantum computing	20
1.4	Contribution of this thesis	25

1.1 History of computing

Calculus means pebble in Latin [Larousse 2023]. Pebbles were the first counting unit used by shepherds to count sheep entering and leaving the sheepfold. On the way back, the number of pebbles remaining indicated the number of sheep missing. This system was also used during conflicts to count the number of soldiers who failed to return from battle. This counting system was soon surpassed by the need to carry out more complex operations, such as trade. Around 2700-2300 BC, the first abacus-type mechanism was invented. It was designed as an extension of the fingers of the hand to enable counting to more than 10. At the time, counting was done in base 5. The abacus (Fig. 1.1 (a)) is considered to be the first version of a calculator. These calculations were still limited to arithmetic operations. Several centuries later, in 1050-771 BC, in ancient China, travelers used a south-pointing chariot to indicate south during a journey. The mechanism was based on a differential gear set up at the start of the journey. This was the first reported analog calculation. The first mechanism considered to be a computer was the Antikythera mechanism (Fig. 1.1 (b)) in ancient Greece, dating from 100 BC [Efsthathiou & Efsthathiou 2018]. It was designed to compute astronomical positions using differential gears. Several gears were involved: one had 365 teeth for the days of the year, another controlled the number of lunations, and so on. To use it, you had to indicate the day's date, and by turning the gears, it gave the date and the positions of the moon and sun for the next eclipse. This computation uses the mechanical rotation of the gears as the computational process, with the input and output data encoded in the position of the gears.

This type of computer, which intelligently uses a continuous variation of a physical device, i.e. an analog signal, is called an analog computer. A basic example of analog computation is the multiplication operation. Knowing Ohm's law, $U = RI$

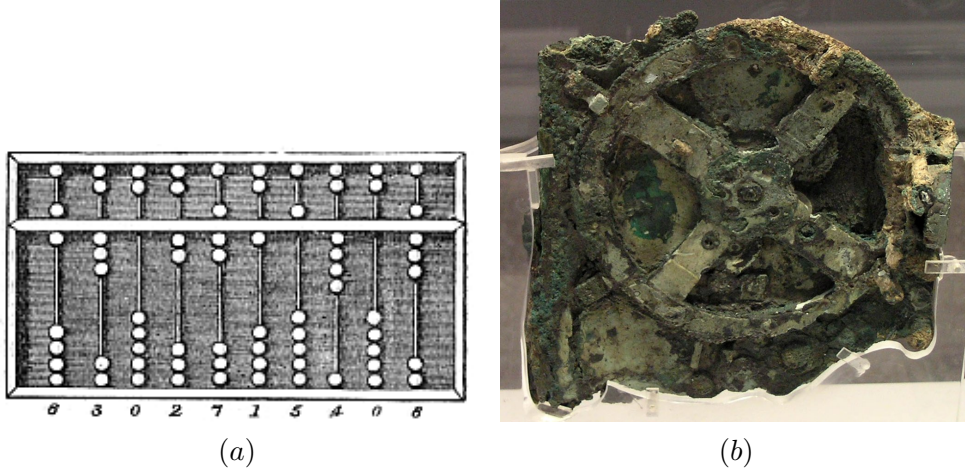


Figure 1.1: (a) Picture of the abacus in [Baynes 1875] and (b) fragment of the Antikythera mechanism from [Wikipedia 2005].

where voltage U is equal to resistance R multiplied by current I , we can compute the multiplication between two inputs that we encode as R and I . By measuring U , we obtain the multiplication of our encoded inputs. It was not until around 1150 that the first “programmable” analog computer appeared, called the Elephant Clock invented by the Arabs [Al-Jazari 1974]. Here, “programmable” means that the mechanism could be adjusted before it was started. It operates on the speed of continuous hydraulic quantities to give the percentage of daylight time elapsed. Although digital computers began to appear at the end of the 18th century, at that time analog computers were still more powerful, as witnessed by the tide-predicting machine capable of performing the Fourier transform [Wikipedia 2024].

We note that all the computer-like systems mentioned above are designed to perform a specific task, such as counting, indicating direction or telling the time. We call a *general-purpose* or a *universal computer* a computer capable of solving any computation in the modern sense of the term, i.e. a Turing-complete computer. We understand that to build a universal computer, we need to have some kind of set of basic operations that can be used as building blocks for a more complex calculation. These operations are necessarily based on analog computation and must also interact with each other. The most practical known physical quantity for performing a computation is the electrical signal.

Until the 60s, one of the last uses of analog calculation was to simulate complex mechanical systems, such as the suspension system of a car with two masses coupled to springs and a damper [Ulmann 2008]. Around 1950-1960, digital computers outperformed analog computers [Wikipedia 2024]. The transistor used as the basic building block became powerful enough to perform any complex calculation more efficiently than analog computers. In a digital computer, information is no longer represented by a continuous physical quantity, but by a discrete value. Of course, hardware such as transistors still use continuous physical variables such as electrical

voltage. A further abstraction makes the link with digitized information: for example, a voltage less than $1/3$ by a certain unit represents the binary value of ‘0’, and a voltage greater than this value represents ‘1’. This abstraction and the universality of digital computers make them more versatile and precise than analog computers for performing elaborate tasks.

In the 21st century, there are very few cases of analog calculation left, the best-known being the mechanical watch and the flight computer. With the rise of artificial intelligence, new prototypes of analog computers are being developed in laboratories for neuromorphic computing [James *et al.* 2017]. The advantage of analog computing is its energy efficiency and speed. Artificial intelligence algorithms are not designed to have exact values, which makes analog computers promising candidates for speeding up, with less energy, the execution of these algorithms.

In any case, to compute something, we need to rely on physical hardware and information encoding. On the one hand, the encoding depends on the choice between digital or analog computation. On the other hand, the physical device is governed by the laws of the underlying physics at play : for mechanical devices, it follows the fundamental laws of dynamics, for electrical devices, it follows the fundamental laws of electromagnetism. Depending on the laws involved, the limitations are not the same. Until today, the *classical* laws of electromagnetism have dominated its use for computation. In the mid-1920s, the laws of quantum mechanics were discovered. It wasn’t until the 1980s that the idea of a quantum computer was born [Benioff 1980, Manin 1980, Feynman 1982]. As with classical computers, there are analog and digital quantum versions. Both are based on the master equation of quantum physics. To access quantum computing, we need to develop a quantum physical device that can be “easily” manipulated. In the following section, we will trace the history from the discovery of quantum mechanics to its use as a computational framework.

1.2 From quantum mechanics to quantum computation

The discovery of the fundamental laws of quantum mechanics is attributed to Erwin Schrödinger in 1926 [Schrödinger 1926]. In his work, Schrödinger relied heavily on the thesis of Louis de Broglie, who was the first to postulate that radiation could be both wave and corpuscular [de Broglie 1924]. Neither de Broglie nor Schrödinger at first seemed to really believe in this particular hypothesis about matter. This counter-intuitive assumption allowed them to derive interesting results for the understanding of the atom.

In classical thermodynamics, we know that during an adiabatic process, certain physical properties such as energy or entropy are conserved [Planck 1903]. Classical evolution is irreversible, so the process is said to be adiabatic if the evolution is extremely fast. Quantum evolution, on the other hand, is reversible. We call a quantum adiabatic process an evolution in which external conditions vary sufficiently slowly. In 1928, Born and Fock proved the first mathematical version of the adiabatic

theorem in quantum physics to better understand the stability of electronic orbitals in atoms [Born & Fock 1928]. They proved that the energy level of a quantum system is conserved under a slow variation of the electromagnetic field. Under fast variation, a state initialized in the lowest energy level, called the *ground-state*, can move to a different energy level, called the *excited state*. Let us illustrate (Fig. 1.2) this important result as it is the theoretical foundation of adiabatic quantum computing (see Sec. 1.3). Imagine the following: Our goal is to transport a baby from its home to a (for now undefined) location. Initially the baby is asleep, and is transported in a stroller. We know that if the person who is pushing the stroller goes slowly, there is a high probability that the baby will stay asleep. On the other hand, if the person pushing the stroller goes too fast, there is a high probability that the baby will wake up, start crying and reach the excited state. The external parameter that changes is the speed of the stroller, and its external environment is defined by its position in space. For a quantum system, the external environment is defined by an object called the Hamiltonian H of the system. For an evolution to be adiabatic, the rate of the change of H is set to be slow enough.

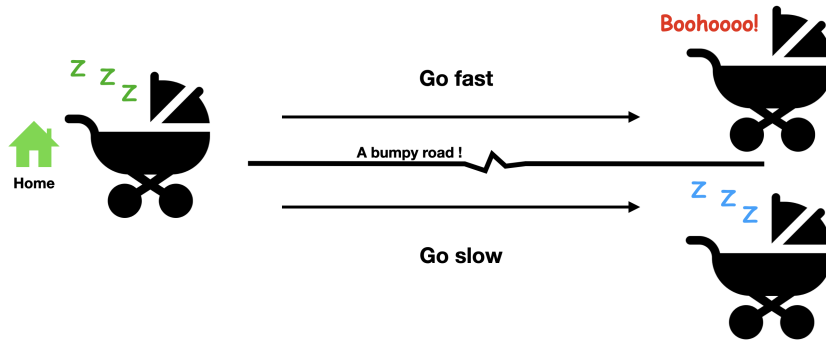


Figure 1.2: A schematic picture to explain the adiabatic theorem (inspired by Pauline Besserve). It does not aim to capture the whole underlying physics. The asleep baby represents the state of lowest energy and the crying baby represents an excited state of the quantum system.

The adiabatic theorem raises some interesting questions. First, what does “slow” mean here? The rate is related to the inverse of the Hamiltonian gap. The gap is the difference in energy levels throughout the evolution. To answer this question about the validity of the adiabatic theorem, von Neumann and Wigner derived a result called the “no-crossing rule”. This rule states that, under certain conditions on the Hamiltonian, two energy levels never actually cross when the Hamiltonian is modified, but can come very close. The phenomenon when they come so close enough that we might think they are crossing is called Avoided level-crossing (AC), which we will formally define in Chapter 2, Sec. 2.4. The gap thus reaches a minimum, which we note as $\Delta_{\min}$. Consequently, in the infinite time regime, the quantum system is guaranteed to remain at the same level from which it started its evolution. In our illustration, this means that the theorem guarantees that the baby will stay

asleep if the stroller moves infinitely slowly. The next question is: what happens if evolution proceeds faster than the adiabatic condition? In 1932, Landau and Zener independently derived the probability that the quantum system will jump from one energy level to the other. These transitions are called non-adiabatic transitions. In plain English, the smaller the gap, the greater the probability of jumping [Landau 1932, Zener 1932]. The best analogy to the small gap point in our example would be the position of the stroller where the baby could easily wake up, e.g. when the road is bumpy. The bumpy parts of the road should be crossed even more slowly than the flat ones. It happens that these bumpy passages can appear several times during the evolution. In the work of [Wilkinson 1987, Wilkinson 1989], the author undertook first a description of AC, then a statistical study of their number.

The first version of the adiabatic theorem has strong constraints on the Hamiltonian, limiting its potential applications. A more general and rigorous mathematical version was derived by [Kato 1950] in Japan. In particular, he got rid of two assumptions about the spectrum of the Hamiltonian. In his version, the latter can be continuous and show degeneracies. Around the same time, in [Anderson 1958], the author demonstrated a physical phenomenon occurring in a quantum system evolving according to Schrödinger’s equation. He asserted that, under certain conditions, no diffusion takes place in a lattice when it was expected. In simpler terms, it suggests that a quantum state can become localized within a specific region before the evolution concludes. This localization may occur in a “wrong” energy level – for instance, in this thesis, the main consequence of this phenomenon is that at some intermediate time of the evolution, the lowest energy state is close to the excited state at final time (Fig. 1.3).



Figure 1.3: Anderson’s localization phenomenon with our illustration. We represent the state inside the bra-ket notation [Nielsen & Chuang 2002]. The lowest energy state at an intermediate moment is the sleeping baby in the middle of the journey. This last state is akin to the crying baby at the end of the journey. It is a bit far-fetched and shows a limit to the comparison.

To understand the different events a quantum system can undergo, we should also mention that it can experience phase transitions. These transitions can be first- or second-order, depending on the continuity of the evolved state at the time of the transition. We refer the reader to [Sachdev 1999] for more details. Another important theoretical result for understanding quantum evolution is the Lieb-Robinson

bound. Informally speaking, it states that information travels at a finite speed in a quantum system [Lieb & Robinson 1972]. This means, among other things, that there is an evolution in total time T such that there is no correlation between two sufficiently distant sites of the quantum system. The “sufficient” distance is defined by the Lieb-Robinson velocity v_{LR} and is equal to $v_{LR}T$.

In the early 70s, shortly after research into theoretical computer science began with the Church-Turing thesis, Cook and Levin independently proved the first combinatorial NP-hard problem [Cook 1971, Levin 1973]. For now, it is enough to think of NP-hard problems as difficult problems that cannot be solved efficiently. This problem is called 3SAT for Boolean Satisfiability over clauses of size three. This problem marks the beginning of complexity theory. This field deals with the analysis of the resources required to compute certain given functions. In general, these are temporal resources. At the same time, Karp proved by reduction a zoo of 21 combinatorial problems that are NP-hard [Karp 1972]. For example, the Maximum Cut (MaxCut) problem is one of them. Let us illustrate this problem (Fig. 1.4).

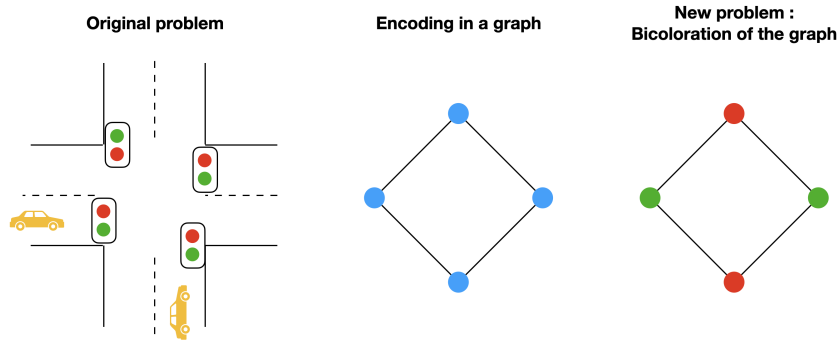


Figure 1.4: Illustration of the MaxCut problem. On the left the original problem. On the middle, its encoding in a graph instance. On the right, a possible solution to the MaxCut problem on the given graph. The latter solution is also solution to the original problem.

Let us imagine an intersection with traffic lights. The aim is to initialize the colors of the traffic lights to prevent as many undesirable events as possible from occurring. Let us assume we only have two colors, red and green. One way of solving this problem is to encode it in a graph; each traffic light is represented by a node in the graph, and we connect two nodes if the corresponding traffic lights should, ideally, be of different colors. The new goal in addressing the original problem is to find a red/green bicoloration of the generated graph. This is the maximum cut, because if we display green nodes on the right and red nodes on the left, all edges crossing the bipartition can be cut. We understand that with a more complex intersection like the one shown in Fig. 1.5, finding the best bicoloration is much less straightforward. The number of color combinations explodes exponentially with the number n of traffic lights as $(\text{number of colors})^n$.

In this thesis, we mainly study this type of problem, called NP-hard

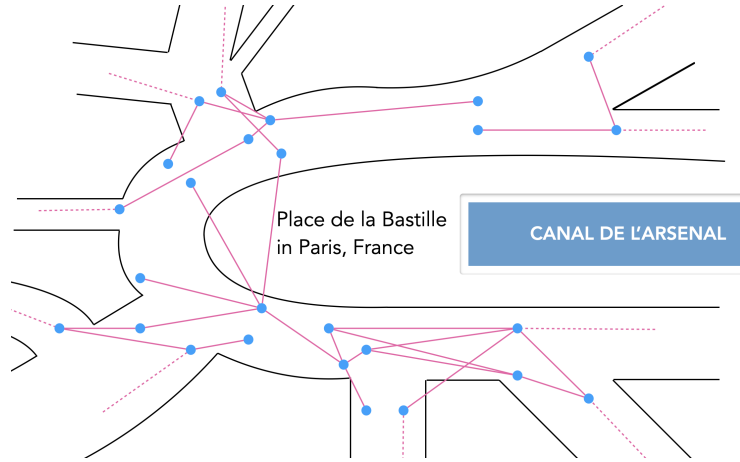


Figure 1.5: Example of a traffic lights problem on a complex instance. As the number of traffic lights increases, the number of combination explodes. The best solution might be *frustrated*, meaning that even two linked traffic lights have the same colors.

combinatorial graph problems. We refer the reader to the following review [Barahona *et al.* 1988] on the application of combinatorial problems. For example, MaxCut’s solution helps design electrical circuits and solves the Ising problem in physics. The NP-hardness of MaxCut means that we do not know of any algorithm that scales polynomially with the input size to solve the problem on an arbitrary input graph. However, we may still be interested in approximate solutions that can be reached efficiently, i.e. in polynomial time. Such algorithms are called *approximation algorithms*, and we evaluate their performance by calculating their approximation ratio. In a nutshell, the approximation ratio tells us how close we are to the optimal solution. The best approximation algorithm for MaxCut on a general graph is that of [Goemans & Williamson 1995], which achieves an approximation ratio of 0.87856, i.e. this algorithm guarantees that its output solution cuts more than 87,856% of all possible edges that can be cut in the optimal solution. If we restrict the graph to 3-regulars, i.e. each node is linked to exactly 3 others, the authors of [Halperin *et al.* 2002] proved that their algorithm achieves an approximation of 0.93 in polynomial time.

Meanwhile, the first idea for a quantum computer was launched independently by Benioff, Manin and Feynman [Benioff 1980, Manin 1980, Feynman 1982]. Feynman thought that the best way to simulate nature, which is quantum, was to use quantum hardware. The idea was formalized by Deutsch, who presented the universal quantum computer [Deutsch & Penrose 1985]. It is fundamentally similar to a “classical” computer, but the underlying physical laws governing the device are quantum laws. In the late ’80s, an Italian team came up with the idea of using quantum fluctuations to solve combinatorial problems [de Falco *et al.* 1988, Apolloni *et al.* 1989]. They introduced the Quantum Annealing (QA) algorithm, a general metaheuristic.

tic of the same type as simulated annealing to minimize a given cost function. It uses quantum tunneling to overcome potential barriers, whereas simulated annealing requires a high temperature to overcome these barriers (Fig. 1.6). In [Kadowaki & Nishimori 1998], the authors provided the first rigorous comparison of simulated annealing and quantum annealing.

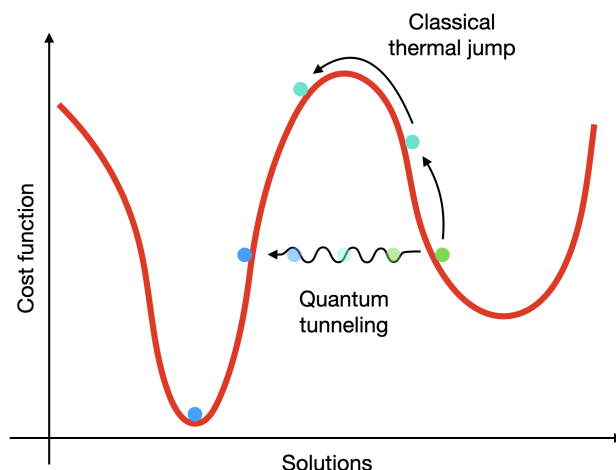


Figure 1.6: Schematic explanation of the physics that helps the state to get over potential barrier.

It was not until the 1990s that the first quantum algorithms formally demonstrated “quantum speed-up” compared with classical computers. This means that quantum computers can solve problems more efficiently than classical computers [Deutsch & Jozsa 1992, Bernstein & Vazirani 1997]. These are numerical algorithms, as the discrete property makes the setting more conducive to demonstrations. Among them, the two most famous are [Shor 1994]’s algorithm for factoring prime numbers and [Grover 1996]’s algorithm for finding a needle in a haystack. These works launched an unprecedented race towards the realization of quantum advantage in practice.

1.3 Theoretical foundation of analog quantum computing

In the digital framework, research is already well advanced, with several concrete proofs of quantum acceleration. The first hardware was beginning to emerge. On the analog side, the D-Wave Inc. launched its first hardware capable of quantum annealing. However, theoretical research into the model was not as advanced as in digital quantum computing. The year 2000 marked the beginning of the formalization of Adiabatic Quantum Computation (AQC) coined by the authors of [Farhi *et al.* 2001]. They formalized the quantum adiabatic computing framework with the guarantee of finding the desired state on the basis of the adiabatic theo-

rem. Basically, in [AQC](#), the initial state is a ground-state of some easy-to-capture Hamiltonian H_0 and the Hamiltonian is continuously modified so that the final Hamiltonian encodes some combinatorial problem. Indeed, as we mentioned earlier, the Maximum Cut problem, for example, is equivalent to solving an Ising problem which is described by the ground-state of a certain Hamiltonian H_1 . Consequently, finding the ground-state of H_1 is equivalent to finding the solution to the MaxCut problem. Thanks to the adiabatic theorem, we know that if evolution is sufficiently slow, the system will remain in the same energy level. So, if it starts from the ground-state (lowest energy state) of a certain initial Hamiltonian H_0 and we change the Hamiltonian slowly over time, at the end the state is the ground-state of H_1 , i.e. the desired solution. To visualize the role of H_0 and H_1 , we understand that H_1 depends on the problem and the input instance of the problem. It is as if we could choose between different final destinations to take the baby, the problem varies if we have to transport the baby to the nursery or to school. Conversely, the initial Hamiltonian depends on the problem and it should be easy to prepare the ground-state of such a Hamiltonian, just as it is easy to put the baby to sleep when he is at home.

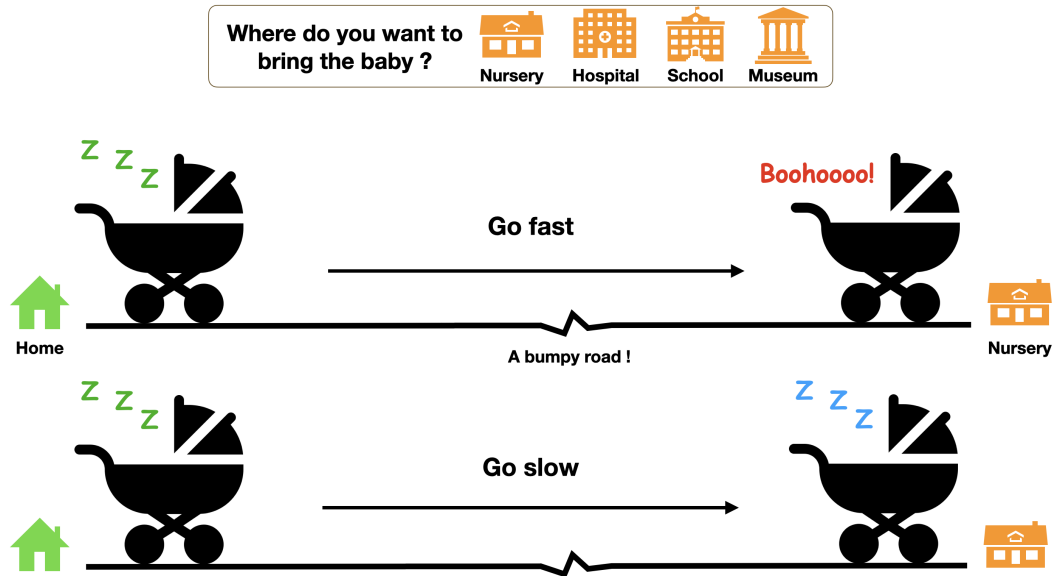


Figure 1.7: A schematic picture to explain [AQC](#) (inspired by Pauline Besserve). The state is the baby in the stroller. Its ground-state is the asleep state. The excited state is the awake crying state. The starting point is some place where it is easy to prepare the baby in the asleep state, e.g. “Home”. The final destination depends on the problem we want to solve. Here, the goal is to bring the baby to the nursery in its asleep state. The straight road is bumpy and by going too fast we increase the probability to awake the baby and finish the trip in the excited state.

Let us imagine for a moment that H_0 is a bathtub. If we compare H_1 to the shape

of the bottom of a bathtub, its ground-state is the deepest point of the bathtub. Evolution begins with a full bath and no knowledge of the shape of its bottom. At the start of evolution, we empty the bath and the state begins to “see” the shape of the bottom. It guesses where the deepest point is as the bath empties. Depending on H_1 ’s landscape, a potential barrier may have separated the bath into several parts. In order to transport part of the state of a well with local minima to the well including global minima, i.e. the ground-state, physics relies on quantum tunneling. This tunneling phenomenon needs time to be effective. In fact, it requires the rate of change of the Hamiltonian to be sufficiently slow in relation to the inverse square of the gap $\Delta_{\min}$. This requirement is exactly that imposed by the adiabatic condition, which then ensures the effectiveness of the tunneling effect and guarantees that the final state is the target state. This new formalism makes the study of the gap the main tool for analyzing the complexity of AQC.

Like the MaxCut problem, it seems that many other combinatorial problems can be encoded in an H_1 Hamiltonian. For example, in [Childs *et al.* 2002], the authors proposed a method for solving the k -clique problem with AQC. For this problem, we look for a clique of a given size k in a given input graph G , a clique being a complete subgraph. In general, combinatorial problems defined on graphs constitute a subset of problems called Quadratic Unconstrained Binary Optimization (QUBO) and go back to [Hammer & Rudeanu 1968]. We say “quadratic” because at most two variables are linked, like two nodes in a graph, and “binary” because the solution is a binary string that associates a bit with each node. Hamiltonians defined on a graph are said to be 2-local because the Hamiltonian’s operators generally propagate on a node and/or an edge (2 nodes). In a k -local Hamiltonian, at most k sites are involved in an operator. In [Cubitt & Montanaro 2016], the authors proved that finding the ground-state of a k -local Hamiltonian is NP-hard. One of the first theoretical results in this framework was proved by [Roland & Cerf 2002]. The authors mainly demonstrate that AQC has the same quantum acceleration for Grover’s problem (the needle in a haystack) by designing a specific Hamiltonian rate-of-change profile such that the gap has the best possible closure. In 2004, a major result of [Aharonov *et al.* 2004] proved the equivalence between AQC and the universal digital quantum computer, making AQC universal for computation. The fact that the digital quantum computer can simulate an analog quantum evolution was already proven, the main contribution focused on the other implication, namely that AQC can efficiently simulate any polynomial-depth quantum circuit.

The difficulty in proving the complexity result in AQC comes from its continuous nature. When the Hamiltonian is time-dependent, there is no general result for the Schrödinger equation. Determining the scale of the gap is also a difficult challenge. Previously, we mentioned that two physical phenomena have been observed to generate small gaps. With AQC, the study of the gap attracts a lot of attention with the aim of proving the tendency to scale with the size of the input instance. [Knysh & Smelyanskiy 2006, Schützhold & Schaller 2006] tackled the gap by observing the quantum phase transition. A first-order phase transition is characterized by an exponentially decreasing gap, while second-order transitions are generally ac-

accompanied by polynomial closure (exceptions can be found in [Tsuda *et al.* 2013]). Perturbation theory provides an approach to exponentially small gaps. The authors of [Amin & Choi 2009] used it to highlight the appearance of a first-order transition in an evolution of AQC for a specific case. In [Altshuler *et al.* 2010], the authors show that Anderson localization can also produce an exponentially small gap for an NP-hard problem. From these results, it is usual to conclude that AQC is inefficient, since the adiabatic guarantee is valid for times greater than $\Delta_{\min}^{-2}$. However, this conclusion is quickly relaxed by [Choi 2011], where the author shows on a particular example that by changing parameters of the final Hamiltonian, we can transform a first-order transition into a second-order one. We refer the reader to [Laumann *et al.* 2015]’s study of the effect of the Hamiltonian parameter on the efficiency of AQC.

Short of mathematical tools to study the analog framework, [Farhi *et al.* 2014]’s work introduced a quantum circuit inspired by AQC to approximate a combinatorial problem. It is called Quantum Approximation Optimization Algorithm (QAOA) and basically, they performed a continuous evolution digitization and then a parameterization to optimize the approximation ratio achieved. This algorithm is also parameterized by its depth. The depth of a quantum circuit is equivalent to the running time of the evolution. In the infinite depth regime, the adiabatic theorem guarantees the optimal solution. The authors proved that with a single layer, QAOA achieves an approximation ratio of 0.6925 for MaxCut on cubic (3 regular) graphs. This type of algorithm, known as a variational algorithm, attracted a lot of attention in the Near Intermediate Scale Quantum (NISQ) era [Preskill 2018], as it performs well in short constant time. In the NISQ era, we assume that quantum computers are still noisy, so the result of an algorithm is not exactly as predicted by theory. Also, the computer has a fairly short lifetime due to fast decoherence. We refer the reader to [Albash & Lidar 2018] for the most recent review on AQC.

More recently, over the last five years, a large number of research papers have been published motivating the use of QAOA as a promising candidate for approximating combinatorial graph problems [Shaydulin & Alexeev 2019, Herrman *et al.* 2021, Farhi *et al.* 2022]. A series of variants of this algorithm have been proposed to improve its performance in practice [Blekos *et al.* 2023]. In [Lykov *et al.* 2023], the authors even suggest that QAOA could provide a quantum advantage over classical algorithms in the NISQ era. Several numerical studies using classical simulation and also quantum hardware such as [Pelofske *et al.* 2024] maintain that QA outperforms QAOA. Both metaheuristics are well suited to approximate combinatorial optimization problems. QA has even been extended to continuous function optimization and industrial applications [Yarkoni *et al.* 2022, Arai *et al.* 2023]. Another analog approach to quantum computation, called Continuous-Time Quantum Walk (CTQW), is also a promising computational framework, as similar analytical results have been proven for its efficiency [Apers *et al.* 2022]. In CTQW, the Hamiltonian is independent of time. In this thesis, we do not cover this topic and refer the reader to a recent review by [Kadian *et al.* 2021]. In the same vein, the authors of [Banks *et al.* 2024] proved

that optimal state transfer techniques achieve an approximation ratio of 0.6003 for the MaxCut problem on cubic graphs. To improve the analog approaches of QA and AQC, one idea is to modify the Hamiltonian in the middle of the evolution by adding another Hamiltonian. This technique stems from the theory behind the idea of *Shortcut-to-Adiabaticity* [Guéry-Odelin *et al.* 2019]. With our example of the baby in the stroller, the idea would be to find a new route from the house to the nursery, avoiding the bumps (Fig. 1.8). From a numerical point of view, this approach looks promising for improving gap scaling [Seki & Nishimori 2012, Feinstein *et al.* 2022].

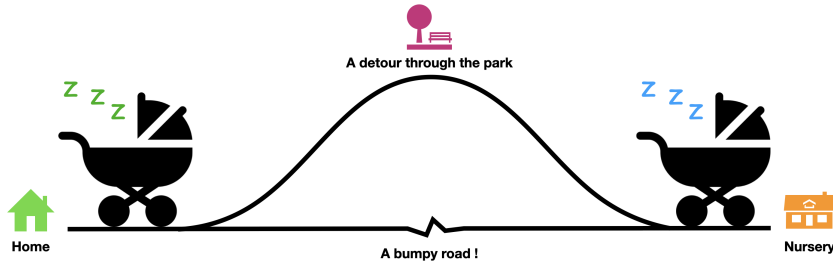


Figure 1.8: A schematic picture to explain the idea of Shortcut-to-Adiabaticity. To avoid the bumps part on the journey, the stroller can take a detour by some designed place, e.g. a square, so that going fast on this path is safe in terms of excitation probability.

Some researchers are focusing more on the optimal design of the total Hamiltonian rate of change, i.e. the combination function that links H_0 and H_1 . This latter function is called the *schedule*. Recall that in QA, the computation does not rely on the adiabatic theorem and has no general guarantee on the output state. With this in mind, any schedule is worth to be tested. It has long been thought that the “bang-bang” form of schedule is optimal [Yang *et al.* 2017]. The “bang-bang” schedule refers to the specific combination in which only H_1 is applied first, then only H_0 . This is exactly what the QAOA circuit implements. However, in [Brady *et al.* 2021], the authors demonstrate that a “bang-anneal-bang” scheme might be more efficient. In other words, a continuous part in the middle of the combination certainly improves the performance of the annealing process. In general, the task of schedule optimization is an important field of research in analog quantum computing (Fig. 1.9).

In terms of understanding anti-crossings, it has been proposed by [Choi 2020] a new definition of this phenomenon by explicitly describing the behaviors of certain quantities of the quantum system. This provides a better understanding of the phenomenon and enables it to be characterized more formally. Finally, we mention some recent uses of the Lieb-Robinson bound in quantum computing. In [Haah *et al.* 2021], the authors proved a bound Lieb-Robinson (LR) for Hamiltonian simulation on a quantum circuit. Their algorithm decomposes the full Hamiltonian acting on all qubits into local gates. The LR bound keeps track of the error made by truncating the gates. The complexity, i.e. the number of gates, of the algo-

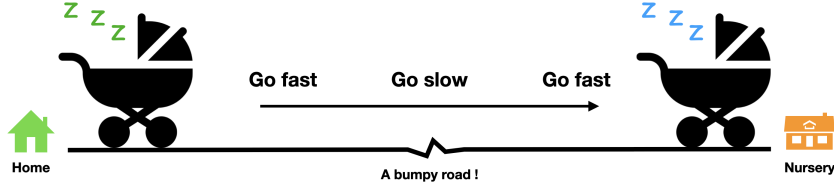


Figure 1.9: A schematic picture to explain the idea of schedule optimization. On flat part of the road the stroller can go fast. The knowledge of bumps positions can help to design optimal driving speed of the stroller. This idea was used by [Roland & Cerf 2002] to prove the Grover speed-up in AQC.

rithm is almost optimal in the worst case. This approach has also been developed in [Tran *et al.* 2019]. A major improvement in the tightness of the LR bound was proposed in [Wang & Hazzard 2020]. In [Moosavian *et al.* 2022], the bound was also used to study the limited performance of quantum annealing in a short running time. They proved that the approximation ratio achieved by QA for the MaxCut problem is lower than the best classical algorithm (Goemans-Williamson) if the running time is less than logarithmic in the input size.

1.4 Contribution of this thesis

This thesis manuscript is based primarily on peer-reviewed articles [Braidà & Martiel 2021, Braidà *et al.* 2022, Braidà *et al.* 2024a, Braidà *et al.* 2024b] and published or accepted in different scientific journals. We tried to leave them as published, we changed some of the notation to keep it harmonious in the manuscript and we added some unpublished work that is more exploratory in nature, but which could provide insights and future possibilities. In Chapter 2, we formally introduce the notations of analog quantum computing and the definition of the Hamiltonians. We detail different algorithms for the MaxCut problem. Then, we clearly define the notion of anti-crossings and the Lieb-Robinson bound.

Unless otherwise stated, all content in Part I and II is original work from the PhD candidate with his supervisors.

Part I focuses on the understanding of the avoided crossing phenomenon in adiabatic quantum computing. In Chapter 3, we first start from the recent characterization of [Choi 2020] and prove how her parametrization is related to the minimum gap. Then applying her definition to a toy model for the maximum k -clique problem, we highlight how the definition can miss an AC (Sec. 3.1). These observations lead us to suggest a new definition, more general than the former (see Fig. 1.10 and Sec. 3.2). It uses a Taylor expansion to justify our new definition and it provides in the same time the rigorous demonstration of the initial description of an AC (Sec. 2.4.2). We validate our definition with numerical evidence. Eventually, we

show some limitation cases that our definition will not capture either (Sec. 3.3). In Chapter 4, we prove a theorem that gives a condition of occurrences of an AC under the validity of the perturbation expansion at first order (Sec. 4.1). We then apply this theorem to the MaxCut problem to prove that AQC efficiently solves MaxCut over regular bipartite graphs and that enough irregularity leads to an exponentially closing gap (Sec. 4.2). Although the general implication of this observation would be to conclude on the failure of AQC in this case, using our new AC definition, a numerical study hints that it is probably not the case. For small instances, the probability to measure the optimal state seems to reach a constant value efficiently. This striking observation leads us to open a question on the possibility to prove an adiabatic theorem relaxed from its dependence in the gap. The last section of the Chapter (Sec. 4.3) suggests other interesting studies of the MaxCut problem using perturbative theory. It is quite remarkable that we can speak about the different coefficients in graph terminology. Would it allow to prove interesting results?

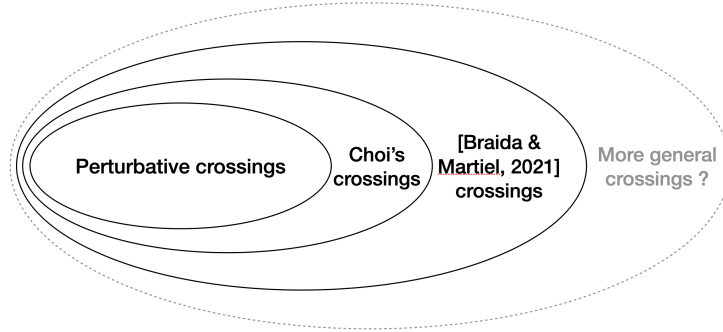


Figure 1.10: Inclusion of the anti-crossings definitions.

Part II targets the potentiality of QA as an approximation algorithm. More precisely, we tackle the study of short constant time QA for MaxCut over cubic graphs. Our first approach (Chapter 5) is a direct adaptation of the QAOA approximation ratio demonstration. We first describe the different steps in the analog regime (Sec. 5.1). Then, the work resides in the computation of a small enough LR bound at distance 1 to derive a non-trivial (above random guess) lower bound on the approximation ratio when QA is studied as a 1-local algorithm (0.5933 in Fig. 1.11 and Sec. 5.2). A tightness study of the LR type bound used leads us to bring some numerically convincing argument that the ratio should be above the QAOA one (0.6925). We suggest a non-implementable algorithm, in the sense that it is too costly to be run, whose ratio is probably close to 0.6963 (Sec. 5.3). In the last project detailed in Chapter 6, we develop a super tight LR bound achieving astonishing numerical values allowing us to reduce the time needed for the hinted previous algorithm. Along with the introduction of a parametrized version of QA, we were able to prove that the approximation ratio reached by QA with a 1-local analysis is above 0.7020, outperforming any known 1-local algorithm for this problem (see Fig. 1.11 and Sec. 6.2). We discuss the tightness of this new bound and

prove that the employed method almost reaches its optimality (Sec. 6.3). In a last chapter (Chapter 7), we develop some insight about the locality in QA, and what is needed to prove a state-dependent LR bound. Eventually, we give some argument on the effective approximation ratio by arguing on the worst graph and by cleverly taking advantage of the graph symmetry, thanks to which we are able to quickly simulate large graphs.

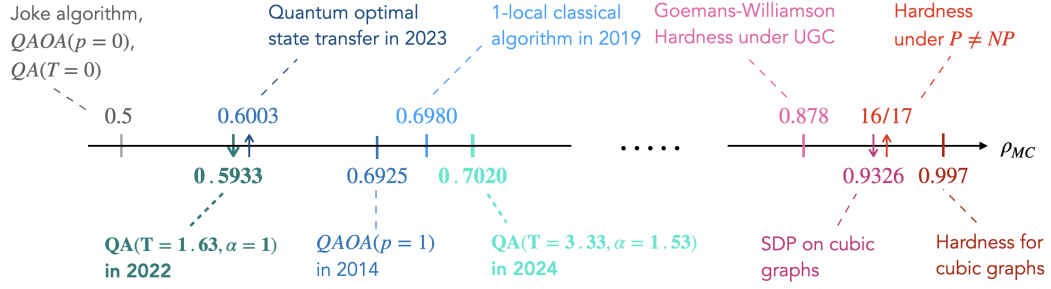


Figure 1.11: Similar to Fig. 2.9 where we added our contributions in bold. They corresponds to a 1-local analysis of QA.

Preliminaries

Contents

2.1	Theoretical foundation of analog quantum computing . . .	30
2.1.1	Complexity theory	30
2.1.2	Adiabatic quantum computing	32
2.1.3	Quantum annealing	36
2.1.4	Hardware and quantum annealer	37
2.2	Hamiltonians definition for graphs combinatorial problems	38
2.2.1	General optimization on graphs	38
2.2.2	Maximum Cut	40
2.2.3	Maximum Independent Set	42
2.2.4	Maximum weight k -clique problem	43
2.3	Approximation of MaxCut	46
2.3.1	Quantum Approximate Optimization Algorithm (QAOA) . .	46
2.3.2	Classical local algorithms	49
2.3.3	Overview of few results	50
2.4	Avoided level crossings (AC)	52
2.4.1	Perturbation theory	52
2.4.2	Initial description of an Anti-crossing	54
2.4.3	Perturbative crossings	55
2.4.4	Choi's crossing definition	58
2.5	Lieb-Robinson bound	60
2.5.1	Observation and intuition	60
2.5.2	Mathematical derivation	62
2.5.3	Some known applications	64

In this chapter, we will develop the mathematical formalism that will be useful for us in pursuing the research that is the subject of this thesis. As far as possible, we have tried to use common symbols for well-known physical quantities, or if we have reused some of the work of other authors, but to keep a harmonious notation in the manuscript, we may adapt some of them.

2.1 Theoretical foundation of analog quantum computing

2.1.1 Complexity theory

Before diving into the quantum formalism, it is essential to establish some fundamental definitions regarding complexity classes for decision and optimization problems. The complexity of an algorithm denotes the resources required for its execution. When unspecified, complexity typically refers to the time needed to run the algorithm, expressed as a function of the input size n , and called the running time T . The algorithm can be of different nature, namely, classical or quantum. The distinction is based on the hardware used to run the algorithm. In the classical case, we distinguish between deterministic algorithms, i.e. same input result in the same output, and probabilistic algorithms, i.e. same input potentially result in different outputs. The output of a quantum algorithm is defined by the probability distribution generated by the final state of the algorithm. Usually, when unspecified, algorithm refers to deterministic algorithm. In the thesis, we simply use algorithm to mention quantum algorithm when it is clear from the context.

Complexity classes were first defined for *decision problems* and then adapted for *optimization problems*. A *decision problem* is a yes-no problem, i.e. if g is an instance of a decision problem, the goal is to answer the question “is there a solution x to g ”? For example, MaxCut above k (“is there a bipartition that cuts at least k edges of the input graph?”) or k -clique (“does the input graph have a complete subgraph of size k ?”) are decision problems. The definition automatically splits the inputs into two subsets, “yes”-instances and “no”-instances, for which the above question is answered by “yes” and “no” respectively. In this thesis, we will restrict to problems whose solutions x can be checked efficiently, i.e. there exists a deterministic algorithm, called the verifier, that takes g and x as an input and verifies if x is a solution for input g . The verifier runs in polynomial time with respect to the input size (in particular the size of x is polynomial in the size of g). Note that for “yes”-instances, there exists x such the verifier accepts, while for no-instances the verifier rejects for any x . This class of problems is called NP:

Definition 2.1 (NP). *The class NP is defined as the set of all decision problems for which the solution of a “yes”-instance is verifiable in polynomial time and any candidate solution of a “no”-instance is rejected by the verifier in polynomial time.*

Among all problems in NP, some of them have solutions that are efficiently computable. It defines a subset of NP and it is called P:

Definition 2.2. (P) *The class P is defined as the set of all decision problems for which there is a polynomial-time deterministic algorithm that can decide if the input is a “no” or “yes”-instance.*

Although it is somewhat trivial to see that $P \subset NP$, it is widely believed that $P \neq NP$ since [Cook 1971, Levin 1973] but no one ever proved or disproved the

converse inclusion. It is widely believed that there exists problems in NP that are not in P. To class the ones that are very unlikely to be in P, we define the notion of NP-hardness. If solving a specific problem L in polynomial time, allows us to solve efficiently any other problem in NP, then L is said to be NP-hard. In addition, if L belongs to the class NP then L is NP-complete. This field of research started with the Cook-Levin theorem [Cook 1971, Levin 1973] where the authors, independently proved the first NP-complete problem. Then, by a technique called polynomial reduction, Karp showed 21 other NP-complete problems [Karp 1972].

All these above definitions were generalized to the probabilistic and quantum case. P in the probabilistic framework becomes BPP, and in the quantum one, it becomes BQP. B stands for “Bounded-error”, meaning that there exists a polynomial randomized/quantum algorithm that gives the correct yes-no answer with probability at least $\frac{2}{3}$. The same generalization exists for the class NP and it becomes MA, for Merlin-Arthur, in the probabilistic regime and QMA in the quantum one. We refer the reader to the following references [Bernstein & Vazirani 1997, Ausiello *et al.* 1999] for a more comprehensive presentation of computational complexity. On Fig. 2.1, we sum up the known inclusion results between the complexity classes (see [Albash & Lidar 2018] for more details).

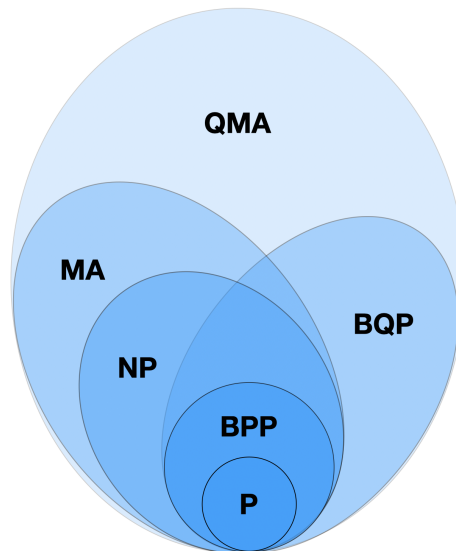


Figure 2.1: Known inclusion of some complexity classes

In this work, we are interested in the optimization versions of those problems. The goal is not to answer the yes-no question anymore but to compute the optimal solution x of the problem input g with respect to some cost function $C(g, x)$. Even though there exists well defined terminology for the optimization versions, for simplicity, we will abusively call them the same way. For example, as we will see in the

Section 2.3, we say that the maximum cut optimization problem is NP-hard.

NP-hard problems are more difficult to solve in terms of time complexity and usually the best known algorithm find the optimal solution in exponentially long time with respect to the input size. What happens if we only allow polynomial time? We only hope for an approximated solution. Those algorithms that run in polynomial time and guarantee that the solution is “close to the optimum” are called approximation algorithms (classical or quantum). To evaluate their performance, we compute their *approximation ratio*. Here, we only consider maximization functions. For a deterministic algorithm $\mathcal{A}$, it is defined as the minimum among all inputs g of the ratio of the cost of the output y of the approximation algorithm with the optimal cost $C_{opt}(g)$ of the input :

$$\rho_{\mathcal{A}} = \min_g \frac{C(g, y)}{C_{opt}(g)}$$

For example, we say that $\mathcal{A}$ is a 0.7 approximation algorithm for maximization problem C if, given any input g , it guarantees that the output solution cost is at least 0.7 times the best value $C_{opt}(g)$.

For probabilistic algorithms, we are interested in the expectation of the output cost value $\mathbb{E}_{y \sim \mathcal{A}}(C(g, y))$, i.e. it is guaranteed that given an input g , on average, the output cost value is ρ_C times the optimal cost value for g .

For quantum algorithms solving classical problems, we also look at the the average output value of the final state measure compared to the optimal one. To write it properly, let us first introduce the quantum computational model we will study in this thesis and its notations.

2.1.2 Adiabatic quantum computing

As in the classical setting, in quantum computing we differentiate between digital and analog computing based on the hardware used. In this thesis we mainly study the quantum analog framework to solve or approximate NP-hard optimization problems. Let us start with a formal definition of **AQC** as defined in the review [Albash & Lidar 2018].

Definition 2.3 (AQC). *A k -local adiabatic quantum computation is specified by two k -local Hamiltonians, H_0 and H_1 , acting on n p -state particles, $p \geq 2$. The ground-state of H_0 is unique and is a product state. The output is a state that is ε -close in the l_2 -norm to the ground-state of H_1 . Let $s(t) : [0, T] \rightarrow [0, 1]$ (the “schedule”) and let T be the smallest time such that the final state of an adiabatic evolution generated by $H(s) = (1 - s)H_0 + sH_1$ for time T is ε -close in l_2 -norm to the ground-state of H_1 . T is called the running time of the computation.*

In this work, we are only interested in Hamiltonians acting on quantum bits (qubits) so $p = 2$. Qubits are 2-states particles that can be represented by $|0\rangle$ the ground-state and $|1\rangle$ the excited state. For a comprehensive introduction to quantum computing, we refer the reader to [Nielsen & Chuang 2002]. Furthermore,

we will mainly be concerned with QUBO problems (Def. 2.4) which implies that our Hamiltonians are $k = 2$ -local. Unless specifically mentioned, we will not use p and k to represent those parameters as we are fixing them here for the rest of the thesis. For AQC to solve QUBO problems, it must be ensured that the final ground-state of H_1 encodes the solution x^* of the optimization problem. The choice of the two Hamiltonians H_0 and H_1 differs according to the problem under study. In this definition, both Hamiltonians are hermitian matrices in $\mathbb{C}^{2^n \times 2^n}$. In Sec. 2.2, we express the Hamiltonians for three problems: maximum k -clique, MaxCut and Maximum Independent Set (MIS). The last quantity to be designed is the schedule $s(t)$ such that $s(0) = 0$ and $s(T) = 1$, the choice being totally arbitrary.

Definition 2.4 (QUBO). *Let $x \in \{0, 1\}^n$ be an n -bitstring and $Q \in \mathbb{R}^{n \times n}$ a weight matrix. The goal of a Quadratic Unconstrained Binary Optimization problem is to find the assignment x^* that minimizes the following function:*

$$f_Q(x) = x^T Q x = \sum_{i,j} Q_{ij} x_i x_j$$

Now, let us call $|\phi_j(s)\rangle$ and $E_j(s)$ the instantaneous eigenvectors and eigenvalues of the total Hamiltonian $H(s)$ acting on n qubits, such that $\forall j \in [0, 2^n - 1], H(s)|\phi_j(s)\rangle = E_j(s)|\phi_j(s)\rangle$ where the eigenvalues are ordered with $E_0(s) \leq E_1(s) \leq \dots \leq E_{2^n-1}(s)$. With this notation, the ground-state is the state of minimum energy, namely $|\phi_0(s)\rangle$. The definition then requires that at $s = 0$, $E_0(0) < E_1(0)$, so that the initial ground-state $|\phi_0(0)\rangle$ is unique and that it is a product state. This last condition can be relaxed with an “easy-to-prepare state” which is the case of product states but in some cases it can be interesting to start from an easy-to-prepare entangled state (see Sec.2.2.4). The output state, which we do not know yet how to construct, is represented by $|\psi_T\rangle$, for now, and is ε -close to the final ground-state of H_1 , i.e. $|\langle\psi_T|\phi_0(1)\rangle|^2 \geq 1 - \varepsilon$. The final ground-state can be degenerate and so the output state can be in a superposition of states lying in the ground subspace of H_1 . We detail in Sec. 2.2 how to write the Hamiltonians for graph combinatorial problems. In [Lucas 2014], the authors show how to write Hamiltonians for many NP-hard problems.

Writing $|\psi(s)\rangle$ the current state of the algorithm, the computation runs as follows for runtime T . The initial quantum state $|\psi_0\rangle = |\psi(0)\rangle$ is the ground-state of the initial Hamiltonian H_0 , then the quantum system evolves under the effect of the defined Hamiltonian $H(s)$ according to Schrödinger’s equation:

$$\forall s \in [0, 1], \quad i\hbar \frac{\partial}{\partial s} |\psi(s)\rangle = T H(s) |\psi(s)\rangle \quad (2.1)$$

As it is standard in this field of research, $\hbar$ is taken as unity which sets the order of magnitude of the Hamiltonians strength. The output state, abusively written earlier $|\psi_T\rangle$ to preserve the dependence on the runtime, is the final state $|\psi(1)\rangle$ of the quantum evolution. It usually ends up in a superposition of classical states. The aim of

AQC is for the latter state to overlap significantly with the final ground-state to solve the problem at hand. The main theoretical support for this property is the adiabatic theorem, which states that a quantum system will remain in the same eigensubspace during the whole quantum process if the evolution is slow enough. “Slow enough” is characterized by the size of minimum spectral gap $\Delta_{\min} = \min_s |E_1(s) - E_0(s)|$ of the total Hamiltonian $H(s)$. In general, with few assumptions on the Hamiltonian, the gap dependence is $\mathcal{O}(\Delta_{\min}^{-3})$ [Jansen *et al.* 2007], and with smoothness hypothesis it can be improved to $\mathcal{O}(\Delta_{\min}^{-2})$ [Elgart & Hagedorn 2012]. In most cases of interest in computer science, the Hamiltonians respect the condition for an inverse square minimum gap dependence of the running time. Here we expose the version of [Morita & Nishimori 2008] for which the adiabatic condition is somewhat easier to capture. It is, however, more restrictive than the previous cited versions as it assumes a non-degenerate instantaneous ground-state throughout the evolution.

Theorem 2.1 ([Morita & Nishimori 2008]). *If the initial state is the ground-state at $s = 0$, i.e. $|\psi(0)\rangle = |\phi_0(0)\rangle$ and the instantaneous ground-state of $H(s)$ is not degenerated for $s \geq 0$, the quantum state $|\psi(s)\rangle$ has the following asymptotic form in the limit of large T :*

$$\begin{aligned} |\psi(s)\rangle &= \sum_j c_j(s) e^{-iT \int_0^s E_j(s') ds'} |\phi_j(s)\rangle, \\ c_0(s) &\simeq 1 + \mathcal{O}(T^{-2}), \\ c_{j \neq 0}(s) &\simeq \frac{i}{T} \left[A_j(0) - e^{iT \int_0^s \Delta_{j0}(s') ds'} A_j(s) \right] + \mathcal{O}(T^{-2}) \end{aligned} \quad (2.2)$$

where $A_j(s) = \frac{\langle \phi_j(s) | \dot{H} | \phi_0(s) \rangle}{\Delta_{j0}(s)^2}$ and $\Delta_{jk}(s) = E_j(s) - E_k(s)$.

Proof. Substituting state (2.2) in Schrödinger equation (2.1) gives us the following equation for the coefficient $c_j(s)$:

$$\dot{c}_j = \sum_{k \neq j} c_k(s) \frac{e^{iT \int_0^s \Delta_{jk}(s') ds'}}{\Delta_{jk}(s)} \langle \phi_j(s) | \dot{H} | \phi_k(s) \rangle \quad (2.3)$$

where we used that $\langle \phi_j(s) | \dot{\phi}_j(s) \rangle = 0$ because the state is of unit norm and we can fix the global phase of the state which is irrelevant in quantum computing. The derivative of the eigen-relation $H(s)|\phi_j(s)\rangle = E_j(s)|\phi_j(s)\rangle$ gives :

$$\langle \phi_j(s) | \dot{\phi}_k(s) \rangle = - \frac{\langle \phi_j(s) | \dot{H} | \phi_k(s) \rangle}{\Delta_{jk}(s)}$$

Then integrating Eq. (2.3) yields

$$c_j(s) = c_j(0) + \sum_{k \neq j} \int_0^s c_k(u) e^{iT \int_0^u \Delta_{jk}(s') ds'} \frac{\langle \phi_j(u) | \dot{H} | \phi_k(u) \rangle}{\Delta_{jk}(u)} du$$

By assumption, the initial state is the ground-state of $H(0)$, so $c_0(0) = 1$ and $c_{j \neq 0}(0) = 0$. By introducing $\alpha_{jk}(u) = e^{iT \int_0^u \Delta_{jk}(s') ds'} \frac{\langle \phi_j(u) | \dot{H} | \phi_k(u) \rangle}{\Delta_{jk}(u)}$ and inserting the former equation into itself, we can distinguish two cases $j = 0$ and $j \neq 0$ as

$$c_0(s) = 1 + 0 + \sum_{k \neq 0} \sum_{l \neq k} \int_0^s \int_0^u c_l(v) \alpha_{kl}(v) dv \alpha_{0k}(u) du \quad (2.4)$$

$$c_{j \neq 0}(s) = 0 + \int_0^s \alpha_{j0}(u) du + \sum_{k \neq j} \sum_{l \neq k} \int_0^s \int_0^u c_l(v) \alpha_{kl}(v) dv \alpha_{jk}(u) du \quad (2.5)$$

We then perform an integration by parts of $\alpha_{j0}(u)$ giving us :

$$\int_0^s \alpha_{j0}(u) du = \frac{1}{iT} \left[e^{iT \int_0^u \Delta_{j0}(s') ds'} A_j(u) \right]_0^s - \frac{1}{iT} \int_0^s e^{iT \int_0^u \Delta_{j0}(s') ds'} \frac{d}{du} A_j(u) du$$

where $[f(u)]_0^s = f(s) - f(0)$. Therefore the first term yields the order T^{-1} part of the $c_{j \neq 0}(s)$ coefficients. The next integration by parts will give the T^{-2} term along with the nested integrals of Eq. (2.5). For the ground-state coefficient $c_0(s)$, we see that the T^{-1} is null and the next one is of order T^{-2} with the nested integrals of Eq. (2.4). $\square$

The condition for adiabatic evolution is straightforward. We say that the quantum evolution is adiabatic if for every $s \in [0, 1]$, the state $|\psi(s)\rangle$ remains close to the instantaneous ground-state of $H(s)$. In other words, the condition translates into $\forall j \neq 0, \forall s, c_j(s) \ll 1$, i.e. $T \gg |A_j(s)|$. We recover the usual writing of the adiabatic validity condition, namely

$$T \gg \frac{\|\dot{H}\|_{\max}}{\Delta_{\min}^2}$$

where $\|\dot{H}\|_{\max}$ is the maximum norm of the time derivative of the Hamiltonian throughout the evolution. We note that this condition is consistent with the expansion presented in the proof of Theorem 2.1. In usual case of combinatorial problem, the numerator scales only polynomially with the input size, making the scaling of the minimum gap the main quantity to analyze to conclude on the complexity of [AQC](#).

The adiabatic theorem is the main theoretical foundation of [AQC](#). This theorem guarantees that the runtime is bounded by the complexity of the minimum spectral gap. In 2004, the authors of [[Aharonov et al. 2004](#)] showed that [AQC](#) is equivalent to the gate-based model of quantum computing making [AQC](#) a universal computational framework. Indeed, any continuous quantum evolution can be efficiently approximated by a quantum circuit using a discretization, a process known as Hamiltonian simulation. Conversely, it is possible to encode a quantum circuit into a local Hamiltonian such that its ground-state contains the output of the circuit. Based on this theorem, the task of analyzing the complexity of adiabatic computation is

restricted to understanding the behavior of Hamiltonian gap. It appears that this task is extremely difficult in the general case, and there are only few cases where an analytical derivation of the gap is possible. The main focus of the studies is to understand how $\Delta_{\min}$ scales with the input size, and to distinguish between hard problems with a gap that closes at least exponentially fast and easy problems with only a polynomially small gap or less. In Sec. 2.4, we will detail the interesting case where the gap closes exponentially fast at avoided level crossing. Two physical phenomena are known to be the source of these AC: first order quantum phase transitions [Sachdev 1999] and Anderson localization [Anderson 1958].

Overall, AQC seems well suited for some specific tasks such as finding the ground-state of an Hamiltonian and very little analytical work is known around its complexity, compared to promising numerical works. The gap dependence is the main hurdle toward analytical proofs of quantum advantage. However, it is still possible to study a quantum evolution by eliminating the gap dependence in the design of the algorithm as we see in the next section. For near-term devices (see Sec. 2.1.4), the runtime is limited by a constant and, in practice, the duration of quantum evolution should be less than this latter upper limit. This is why there is great interest in understanding quantum evolution performance on different time scales.

2.1.3 Quantum annealing

In this setting, the runtime T can be arbitrarily chosen and the initial state may not be the initial ground-state (although this is usually the case). Depending on the choice of T , the evolution is not necessarily adiabatic and the final state no longer has the same guarantee. We call such an evolution quantum annealing (QA) and it is a general meta-heuristic for approximating combinatorial problems. It has been introduced in [de Falco *et al.* 1988, Kadowaki & Nishimori 1998] with practical implementation in [Apolloni *et al.* 1989]. With this definition, QA is more general than AQC but the convergence of QA towards the optimal solution is still guaranteed by the adiabatic theorem. This opens the door to approximation study or to parametrization of AQC to optimize a certain target measure. To evaluate the performance of the evolution, we may want to study/optimize the probability of measuring the target state $|x^*\rangle$, i.e. $|\langle x^* | \psi_T \rangle|^2$. Nonetheless, as the runtime is arbitrarily chosen, away from the adiabatic regime, it is most likely that the output state differs from the target. Depending on T , the output state can still have a good enough cost value not too far from the target. The output state is an approximate solution. To capture “good enough”, another potential performance indicator is the approximation ratio (Def. 2.5), the goal being to optimize the average value of the final state with respect to the optimal value.

Definition 2.5 (Approximation ratio). *Given an input graph G on which we want to solve an maximization problem C , and a quantum algorithm $\mathcal{A}$ that outputs a potential solution x for the problem with value $C(G, x)$, the approximation ratio $r(C(G), \mathcal{A})$ is defined by the ratio of the expected output of $\mathcal{A}$ and the optimal value*

$C_{opt}(G)$.

$$r_{C(G),\mathcal{A}} = \frac{\mathbb{E}_{x \sim \mathcal{A}}(C(G, x))}{C_{opt}(G)} \quad (2.6)$$

In this thesis, we are interested in the worst case scenario. This means that we are looking for input-independent lower bounds of the worst possible ratio taken over all possible input graphs:

$$\rho_{C,\mathcal{A}} = \min_G \frac{\mathbb{E}_{x \sim \mathcal{A}}(C(G, x))}{C_{opt}(G)}$$

In our notation, the final expected output is written as $\langle \psi_T | H_1 | \psi_T \rangle$. To see this, let us decompose $|\psi_T\rangle$ in the computational basis of the bit-strings, and write

$$\begin{aligned} \langle \psi_T | H_1 | \psi_T \rangle &= \sum_{x,y \in \{0,1\}^n} \langle \psi_T | y \rangle \langle y | H_1 | x \rangle \langle x | \psi_T \rangle \\ &= \sum_{x \in \{0,1\}^n} \langle x | \psi_T \rangle \langle \psi_T | x \rangle C(x) \\ &= \sum_{x \in \{0,1\}^n} |\langle x | \psi_T \rangle|^2 C(x) \\ &= \mathbb{E}_{x \sim \mathcal{A}}(C(x)) \end{aligned}$$

where we use the fact that $\langle x | H_1 | y \rangle = C(x) \delta_{xy}$ and we remove the dependency in G of C as it is clear from the context. δ_{xy} represents the Kronecker delta symbol that evaluates to 1 when its indices are equal ($x = y$) and 0 otherwise ($x \neq y$).

This broader setting of [QA](#) offers room to analyze the quantum evolution as an approximation algorithm for shorter runtimes, away from the adiabatic condition. In this thesis (Part II), we undertake the specific runtime regime of constant time [QA](#) with the goal to formally prove lower bounds on the approximation ratio achieved by [QA](#) on some specific benchmark problems that we will detail in the next section. Similar results exist in the bounded depth quantum circuit model as well as in the classical case with constant round distributed algorithms. In Sec. 2.3, we will develop the construction of two other algorithms that formally prove lower bounds on the worst-case approximation ratio.

Quantum annealing seems to be even more suitable for NISQ use, as the hardware developed to perform this metaheuristic have a limited “quantum lifetime”. As we will see in the next section, this lifetime is called decoherence time.

2.1.4 Hardware and quantum annealer

The hardware available today for performing continuous quantum evolution is generally referred to as quantum annealer. As mentioned in the Introduction chapter 1, D-Wave was the first company to commercialize this type of hardware. Subsequently, many other companies began to develop technology to perform continuous

quantum evolution, such as Pasqal [Henriet *et al.* 2020], Qilimanjaro, QuEra Computing and others. The main limitation of current hardware is the total runtime it allows. This limitation means that adiabatic evolution cannot be guaranteed in the general case. This makes QA the main metaheuristic for analog quantum computing in the NISQ era. However, very few analytical results are known for this regime, so in this thesis we set out to prove the algorithmic performance of QA in constant time (Part II).

As an example, the hardware from the French company Pasqal uses Rydberg atoms as the physical building blocks of quantum information. Indeed, in [Saffman *et al.* 2010], the authors show how to use this technology to perform quantum computations. In [Serret *et al.* 2020], the authors provide quantitative requirements for running a QA algorithm with this hardware. A Rydberg atoms Hamiltonian naturally encode a specific combinatorial problem, called the MIS problem (see Sec. 2.2.3). A recent study [Ebadi *et al.* 2022] published in Science claimed a “quantum speed-up” using 287 Rydbergs atoms to solve a combinatorial graph problem compared with simulated annealing, the classical version of QA. Later in [Andrist *et al.* 2023], the result was mitigated by using a broader range of classical solvers.

2.2 Hamiltonians definition for graphs combinatorial problems

In both AQC and QA, the encoding of the target Hamiltonian H_1 is crucial. As stipulated by Def. 2.3, the ground-state of H_1 should encode the desired output state. As mentioned earlier, in this work we will mainly focus on graph optimization problems. In this section, we first develop the general form of the target Hamiltonian to solve such combinatorial problems. Then we explicit the expression of the Hamiltonians for three problems : Maximum Cut, Maximum Independent Set and Maximum (weight) k -clique. While the initial Hamiltonian H_0 remains independent of the input and the same one can always be used, its design can be tailored according to the nature of the problem, as in the maximum (weight) k -clique problem. For more examples of problems encoded in Hamiltonian, we refer the reader to the seminal work of [Farhi *et al.* 2001] and the comprehensive work of [Lucas 2014].

2.2.1 General optimization on graphs

We consider a restricted class of problems whose inputs are graphs and outputs are set of nodes, or two-partitions of the input node set, minimizing some global cost function. Moreover, we focus our interest on combinatorial graph problems where the global cost is a sum of cost functions localized on the nodes and the edges of the input graph.

Formally, we are given as input some graph $G = (V, E)$ on n nodes, and our cost function C associates any n -bit vector x (bit x_i corresponding to the Boolean value

of node i) with a value $C(x) = \sum_{i \in V} v(x_i) + \sum_{\langle i,j \rangle \in E} u(x_i, x_j)$, for certain functions $v : \{0, 1\} \rightarrow \mathbb{N}$ and $u : \{0, 1\} \times \{0, 1\} \rightarrow \mathbb{N}$. In general, the weighted version of each problem can be easily expressed by including the dependence of the weights in the cost functions u and v . The values taken from the cost function are then $\mathbb{R}$. Moreover, if we assume that the local cost functions are quadratic, it is easy to see that the resulting cost functions are a special case of Def 2.4, see [Hauke *et al.* 2020].

In quantum computing, we call the basis $\{|x\rangle, x \in \{0, 1\}^n\}$ the *computational basis*. A quantum annealing algorithm, *a fortiori* an adiabatic quantum computation algorithm, for minimizing a cost function C on n -node graphs is defined by three ingredients:

Final Hamiltonian H_1 . A target Hamiltonian H_1 (i.e. a hermitian operator) encoding the solution of the cost function C as its ground-state. The generic way to ensure this property is to encode the entire cost function, in the sense that for all $x \in \{0, 1\}^n$, $H_1|x\rangle = C(x)|x\rangle$. Notice that H_1 is diagonal in the computational basis. This completely defines H_1 and the target state x^* , which minimizes C , is the ground-state as $H_1|x^*\rangle = C(x^*)|x^*\rangle$. This construction is achieved by encoding each clause of C separately: $H_1 = -\sum_{i \in V} N^{(i)} - \sum_{\langle i,j \rangle \in E} M^{(i,j)}$. It follows that the structure of C is preserved by this Hamiltonian H_1 and local clauses become local terms (also called observables). Here, $N^{(i)}$ encodes the local cost function $v(x_i)$ of node i acting on qubit i and $M^{(i,j)}$ the local cost function $u(x_i, x_j)$ of edge $\langle i, j \rangle$ acting on qubits i and j . They can be defined as follows:

$$\begin{aligned} N^{(i)} &= (v(0)|0\rangle\langle 0| + v(1)|1\rangle\langle 1|) \otimes \mathbb{I}_{V \setminus \{i\}} \\ M^{(i,j)} &= (u(0,0)|00\rangle\langle 00| + u(0,1)|01\rangle\langle 01| \\ &\quad + u(1,0)|10\rangle\langle 10| + u(1,1)|11\rangle\langle 11|) \otimes \mathbb{I}_{V \setminus \{i,j\}} \end{aligned}$$

Initial Hamiltonian H_0 . An input-independent initial Hamiltonian H_0 to start our annealing. The ground-state of this Hamiltonian must be easy to prepare, as, for example, the product states. Apart from this technical constraint, to understand the choice of this Hamiltonian, we need to understand the role it plays in the evolution. Let's consider the evolution of the amplitude of a state $\langle x|$ in the quantum system $|\psi(s)\rangle$. The Schrödinger equation allows us to write

$$\begin{aligned} i \frac{d|\psi(s + ds)\rangle}{ds} &= TH(s)|\psi(s)\rangle \\ \Rightarrow |\psi(s + ds)\rangle &= (I - iTds((1-s)H_0 + sH_1))|\psi(s)\rangle \\ \Rightarrow \langle x|\psi(s + ds)\rangle &= (1 - iTsC(x)ds)\langle x|\psi(s)\rangle - iT(1-s)ds\langle x|H_0|\psi(s)\rangle \\ \Rightarrow \langle x|\psi(s + ds)\rangle &= (1 - iTsC(x)ds)\langle x|\psi(s)\rangle - iT(1-s)ds \sum_{y \underset{H_0}{\sim} x} (H_0)_{xy}\langle y|\psi(s)\rangle \end{aligned}$$

where I is the identity matrix and $y \underset{H_0}{\sim} x$ means that the matrix element $(H_0)_{xy}$ is non-zero, i.e. states x and y are related via H_0 . We understand from the last

equation that the amplitude at time $s + ds$ of state x is influenced by the amplitude of x at time s but also by the amplitude of “neighboring” states y via H_0 . Without saying more on the evolution, we understand that if we want to reach some state x^* , i.e. the amplitude of x^* at time $s = 1$ is large, it requires that this state “communicates” with other states. Otherwise, the probability of measuring the target state at the end of the evolution will be the same as the initial probability. Therefore, H_0 should connect all states that are potential solution to the optimization problem. Informally, physicists call this a driver in which the quantum system can evolve.

Schedule $s(t)$. The schedule is a function $s(t) : [0, T] \rightarrow \mathbb{R}$ with $s(0) = 0$ and $s(T) = 1$ so that $H(0) = H_0$ and $H(s(T)) = H_1$ and it rules the trajectory in time of the total Hamiltonian $H(s)$. It also fixes the runtime T and depending on the scaling of the minimum gap, the evolution will be adiabatic or not. This degree of freedom allows a wide range of possible functions and it includes the whole research area of optimal transport (see [Brady *et al.* 2021]). Freedom in the choice of the schedule also enables to add a catalyst Hamiltonian in the middle of the evolution. For example, the following total Hamiltonian is allowed

$$H(s) = (1 - s)H_0 + s(1 - s)H_{cat} + sH_1$$

The most standard choice of schedule is the linear interpolation with $s(t) = \frac{t}{T}$ which is a straight line between 0 and 1. Unless stated otherwise, in this work we mostly focus on this standard schedule without perturbative Hamiltonian and we leave this question for further work. It is interesting to point out that a local adiabatic condition can be stated as

$$\frac{ds}{dt} \ll \frac{\Delta_{01}(s)}{\langle \phi_1(s) | \dot{H} | \phi_0(s) \rangle}$$

enabling for an adjusted design of the schedule, i.e. to go fast when the gap is large and slow down when it is small. This idea has been deftly manipulated in [Roland & Cerf 2002] to prove the well-known Grover speedup in the adiabatic regime. The authors adjusted the speed of the evolution according to the gap leading to a particular form of schedule. We see that the standard linear schedule has a speed of $1/T$ which does not allow for intelligent control of the speed.

These three ingredients are both necessary and sufficient for initiating a quantum annealing process. In order to comprehensively understand the Hamiltonians formulation, we provide a detailed description of the Hamiltonians employed to investigate the computational complexity of solving three combinatorial graph problems. It is important to note that the complexity depends on the choice of your Hamiltonians [Choi 2011].

2.2.2 Maximum Cut

Maximum Cut as an optimization problem, is a combinatorial problem in which one has to find a bi-partition of the nodes of an input graph $G(V, E)$ such that the

number of edges across the bi-partition is maximized (see Fig. 2.2). The bi-partition is represented by a n -bit vector x with partition '0's and '1's (respectively green and red in Fig. 2.2). The associated cost function is $C_{\text{MaxCut}}(x, G) = \sum_{(i,j) \in E} x_i \oplus x_j$ where $\oplus$ is a XOR operator between two bits. It evaluates to 1 if $x_i \neq x_j$ and 0 otherwise. Using the notation of the previous section, we have that function $v(\cdot)$ is null and the cost function $u(\cdot, \cdot)$ is the XOR operation. It is interesting to note that this function has a $\mathbb{Z}_2$ symmetry meaning that by flipping all the bits of a bitstring x , it gives another bitstring $\bar{x}$ with the exactly the same cost value. Also, without any knowledge of the input labeling, any bitstring is a potential solution.

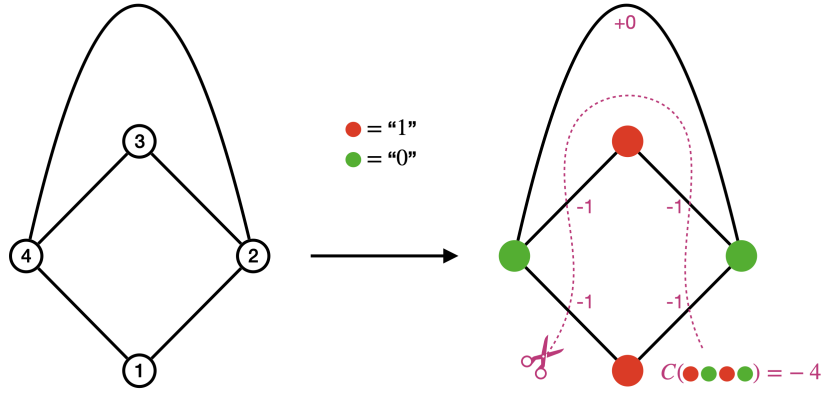


Figure 2.2: Example of Maxcut solving on the graph (left) with solution (right). The bi-partition green/red represents the bi-partition '0' and '1'. The solution proposed on the right has value $C(1010) = -4$.

To solve this problem using quantum annealing, we first look at the opposite cost function $C = -C_{\text{MaxCut}}$, as the setting of QA we defined earlier can be used to minimize some functions.

A standard choice for the **initial Hamiltonian** is

$$H_0 = - \sum_{i \in V} \sigma_x^{(i)}$$

where $\sigma_x^{(i)}$ is the bit-flip operator acting on qubit i . The ground-state (i.e. the eigenvector of minimum eigenvalue) of this Hamiltonian is the product state $|\psi_0\rangle = \left(\frac{|0\rangle + |1\rangle}{\sqrt{2}}\right)^{\otimes n}$, the uniform superposition of all possible bit-strings of length n . This state is particularly easy to prepare (it is not entangled) and will be the starting point of our computation. Now, if we look at the matrix elements of H_0 , we understand that for any bit-strings x and y , $(H_0)_{xy} = -1$ if x differs from y by exactly one bit and 0 otherwise. This observation tells us that the graph generated by $-H_0$ is the hypercube, e.g. see Fig. 2.3. This choice of Hamiltonian satisfies the connection condition between all the possible solutions.

For the **final Hamiltonian** H_1 , we observe that the cost function can be rewritten in terms of variables $z_i = 2x_i - 1 \in \{-1, +1\}$ as $C(z) = - \sum_{(i,j) \in E} \frac{1 - z_i z_j}{2}$. It

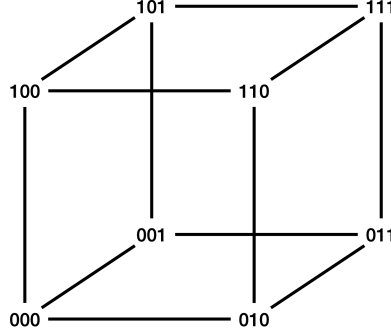


Figure 2.3: Example of the graph generated by $-H_0$ with $n = 3$. Each node of the graph represents a potential solution, i.e. a bi-partition of the input graph G .

computes exactly the same function. To encode it in a Hamiltonian, we use the Pauli σ_z operator where state $|0\rangle$ is the $+1$ eigenvector and state $|1\rangle$ is the -1 eigenvector. For quantum computing, this operator is the perfect basis transformation from $\{0, 1\}$ to $\{-1, +1\}$. Thus, the Hamiltonian writes

$$H_1 = - \sum_{\langle i, j \rangle \in E} \frac{1 - \sigma_z^{(i)} \sigma_z^{(j)}}{2}$$

so that it verifies for any bit-string $H_1|x\rangle = C(x)|x\rangle$. The optimal state x^* is the one that minimizes C , so the quantum state $|x^*\rangle$ is the ground-state of H_1 .

These two Hamiltonians are well defined as required by the setting of [AQC](#) or [QA](#), and the adiabatic theorem guarantees that the adiabatic evolution will find the solution to this problem. In Sec. 2.3, we review some of the known approximation ratios proved on this problem in the classical and quantum world.

2.2.3 Maximum Independent Set

Maximum Independent Set as an optimization problem, is a combinatorial problem where the goal is to find the largest set S of nodes of a graph $G(V, E)$ such that there is no edge of E between the nodes of S , called a *stable*. We encode the potential independent set in a vector $x = x_1 x_2 \dots x_n$ with $x_i = 1$ if $i \in S$ and $x_i = 0$ otherwise. The cost function is written in two parts because we want to maximize the Hamming weight of x while minimizing the number of edges between nodes i and j such that $x_i = x_j = 1$. We write $C_{\text{MIS}}(x) = \sum_{i \in V} x_i - \sum_{\langle i, j \rangle \in E} x_i x_j$. Even though this function only weakly encodes the independent set constraint, one can easily transform any bit-string x of cost $C_{\text{MIS}}(x)$ into a stable of size at least $C_{\text{MIS}}(x)$. This can be done considering each edge that violates the stable condition (i.e. each edge $\langle i, j \rangle$ such that $x_i = x_j = 1$), picking at random one of its extremities, and removing it from the solution. This effectively removes a node from x while “fixing” at least an edge, thus creating a new solution with an increased cost. Eventually, the maximum of $C_{\text{MIS}}(x)$ corresponds to a maximum independent set.

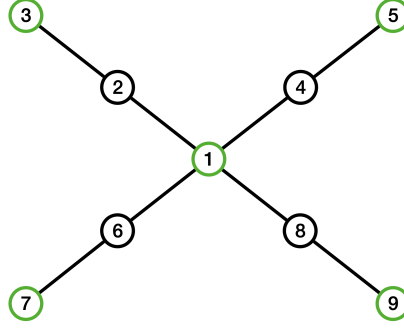


Figure 2.4: Example of an MIS in a graph. The MIS in green node is encoded in vector $x = 101010101$ with value $C(x) = -5$.

Like the MaxCut problem, to solve MIS via analog quantum computing, we will look at the cost function $C = -C_{\text{MIS}}$. The same initial Hamiltonian works as well for this problem, i.e. $H_0 = -\sum_{i \in V} \sigma_x^{(i)}$. In order to implement C into a Hamiltonian we proceed to the same change of basis as for MaxCut: $\{0, 1\}$ is transformed into $\{1, -1\}$ and C can be written as $C(z) = \sum_i \frac{1-z_i}{2} - \sum_{\langle i,j \rangle} \frac{1-z_i}{2} \frac{1-z_j}{2}$. Thus, the final Hamiltonian for MIS that QA will minimize is:

$$H_1 = -\sum_{i \in V} \frac{1}{2}(1 - \sigma_z^{(i)}) + \sum_{(i,j) \in E} \frac{1}{4}(1 - \sigma_z^{(i)})(1 - \sigma_z^{(j)})$$

Unlike MaxCut, we see here that we have a component on the nodes and a component on the edges; $N^{(i)} = \frac{1}{2}(1 - \sigma_z^{(i)})$ and $M^{(i,j)} = -\frac{1}{4}(1 - \sigma_z^{(i)})(1 - \sigma_z^{(j)})$. The ground-state of this Hamiltonian is the MIS.

2.2.4 Maximum weight k -clique problem

First, let us define the Maximum Clique problem. As an optimization problem, the goal is to find the clique of maximal size in an input random graph $G(V, E)$. It is directly equivalent to solving MIS problem in the complementary graph of G . A clique is a complete graph, so its complementary graph is an independent set. The complement of a graph G is a graph G' on the same nodes such that two distinct nodes of G' are adjacent if and only if they are not adjacent in G . Similarly, a bit-string $x = x_1 x_2 \dots x_n$ encodes a potential solution as it selects the subgraph generated by the set $\{i, x_i = 1\}$.

In the maximum k -clique optimization problem, we parametrized the Maximum Clique problem by the knowledge of the maximum clique size k . This parameter allows us to focus on bit-strings of Hamming weight equal to k as we are only interested in sub-graphs of size k . Thus the whole search space of size 2^n is reduced to the $\binom{n}{k}$ search space, that is the space spanned by bit-strings of Hamming weight k . Consequently, in the cost function, we can remove the part that deals with the Hamming weight and we are left with $C_{k\text{-clique}}(x) = \sum_{(i,j) \notin E} x_i x_j$. It counts the number of edges missing from the subgraph generated by x for it to be a clique. In

Fig. 2.5 for example, looking at the subgraph generated by nodes 1,2 and 3 results in the vector $x = 111000$ with cost value $C_{k\text{-clique}}(x) = 1$.

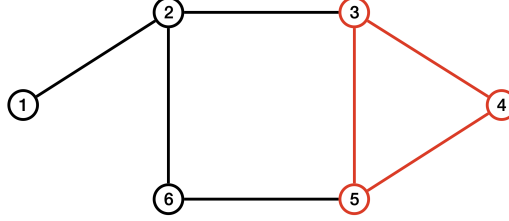


Figure 2.5: Example of a 3-clique in a graph. The solution in red is encoded in vector $x = 001110$ and its value is $C(x) = 0$.

The authors of [Childs *et al.* 2002] were the first to introduce the potential of AQC to solve the maximum k -clique problem. Cleverly and unlike the previously presented problems, the chosen initial Hamiltonian H_0 stabilizes the $\binom{n}{k}$ Hilbert space and thus preserves the Hamming weight of the computational basis vectors encoding our subgraphs. The S operator, exchanging the position of x_i and x_j in vector x , does not affect the Hamming weight and is the retained candidate in the original paper. Therefore, we write:

$$H_0 = - \sum_{(ij) \in E(G_{\text{driver}})} S^{(ij)}$$

where

$$S^{(ij)} = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}^{(ij)} \quad \text{swaps qubits } i \text{ and } j$$

and G_{driver} represents the graph of the allowed swaps between the qubits. In the original work [Childs *et al.* 2002], they only look at the complete driver graph, allowing the swap of any pairs of qubits. Then noticing that $2S^{(ij)} = \sigma_x^{(i)} \sigma_x^{(j)} + \sigma_y^{(i)} \sigma_y^{(j)}$, we can write H_0 with quantum operators like:

$$H_0 = -\frac{1}{2} \sum_{(ij) \in E(G_{\text{driver}})} (\sigma_x^{(i)} \sigma_x^{(j)} + \sigma_y^{(i)} \sigma_y^{(j)})$$

Now, $-H_0$ does not generate a hypercube anymore, but still connects every potential solution if G_{driver} is connected. Let us illustrate with a path graph as the driver graph. In Fig. 2.6, we plot on the top a line graph of size 5 and the corresponding graph with adjacency matrix $-H_0$ and $k = 3$. We can see that the vector x is linked to the vector y if, for a node $i \in V(G_{\text{driver}})$, the exchange of the positions of x_i and x_{i+1} gives y . For example, vector $x = 11100$ has only one neighbor because only x_3 and x_4 can swap. This is why the state that alternates between ‘0’ and ‘1’ is the highest degree node, since each swap of pairs (x_i, x_{i+1}) results in a new state. This driver graph gives us another degree of freedom to play with.

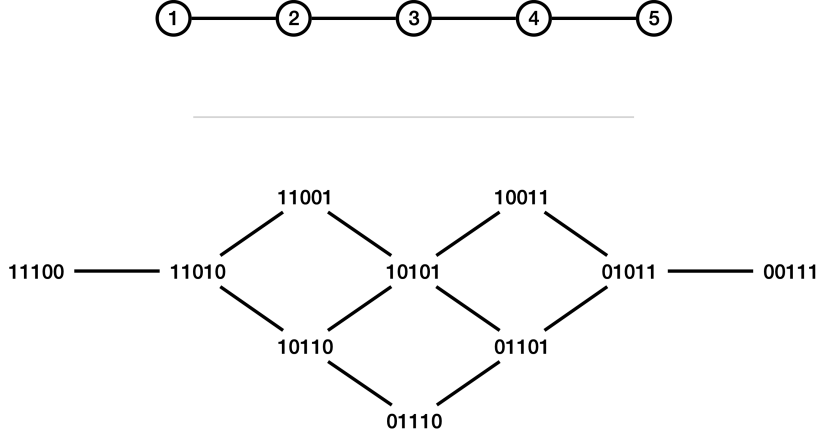


Figure 2.6: Example of a connected G_{driver} , a line on $n = 5$ nodes (top) and the generated search graph (down) with adjacency matrix $-H_0$ associated to the given driver graph where we implicitly fix $k = 3$.

For the final Hamiltonian, with a similar reasoning from the MIS problem, it is written as follows:

$$H_1 = \frac{1}{4} \sum_{i,j \notin E} (1 - \sigma_z^{(i)})(1 - \sigma_z^{(j)})$$

Now, we will introduce the weighted version of the k -clique problem¹. We suppose that each node i of the input graph has a weight w_i and the goal is to find the k -clique such that it maximizes the sum of the weights of the clique. The cost function can be updated to $C(x) = \sum_{(i,j) \notin E} x_i x_j - \alpha \sum_{i \in V} w_i x_i$ with a free parameter α that we can arbitrarily turn on to give more or less importance to the weights. The resulting final Hamiltonian for the weighted version is

$$H_1 = \frac{1}{4} \sum_{i,j \notin E} (1 - \sigma_z^{(i)})(1 - \sigma_z^{(j)}) - \alpha \sum_{i \in V} \frac{1 - \sigma_z^{(i)}}{2} w_i$$

The above presented problems are the three problems on which we applied the theoretical results we developed during this thesis. Now that we know how to write the Hamiltonian for different graph problems, we have every ingredient to study the QA evolution. Before moving on to the specific questions that we tackle in this thesis to provide some numerical and analytical results on the complexity of AQC and QA applied to these problems, in the next section, we present a quick overview of the state-of-the-art results on the MaxCut problem as well as the construction of two algorithms that can be compared to QA.

¹We draw the reader's attention to the fact that only the optimal solution is expected to be a clique, whereas approximate solutions may be only weighted dense k -subgraphs. We use the k -clique terminology as introduced in the original article [Childs *et al.* 2002].

2.3 Approximation of MaxCut

In this section, we first show the construction of two algorithms that have proven an approximation ratio on MaxCut over regular graphs and can be compared to quantum annealing. Then we review some known results about the approximability of the MaxCut problem in order to put the complexity of the problem into perspective. We also mention recently proved results on QA applied to MaxCut.

2.3.1 Quantum Approximate Optimization Algorithm (QAOA)

QAOA is a gate-based algorithm introduced by [Farhi *et al.* 2014]. The construction of this algorithm is inspired by the quantum adiabatic evolution. It works in two parts: first a digitization of the continuous evolution and then a Trotterization and parametrization of the gates.

The first idea starts with trying to approximate the unitary operator U_T of the continuous evolution. This operator is the solution of the Schrödinger evolution, i.e. the operator that transforms $|\psi_0\rangle$ in $|\psi_T\rangle$. We can write this as $|\psi_T\rangle = U_T|\psi_0\rangle$. There is no known closed form for this unitary when the Hamiltonian $H(s)$ is time-dependent. The continuous interval $[0, T]$ is decomposed into p pieces of size δ_t so that $p\delta_t = T$. For large enough p , $H(s)$ is well approximated by the function constant by pieces of values $H(j\delta_t)$ for $j \in [0, p]$. Now, the Schrödinger equation can be solved on every interval $[j\delta_t, (j+1)\delta_t]$ because the Hamiltonian is constant. The unitary operators are therefore equal to $e^{-i\delta_t H(j\delta_t)}$. The total evolution operator can be approximated by the product of all these operators, i.e.

$$U_T \simeq \prod_{j=0}^p e^{-i\delta_t H(j\delta_t)}, \quad \text{for large enough } p$$

We can already state that the limit when p tends to infinity of the right hand side is the unitary of the adiabatic evolution.

The next step is to decompose the exponential. We know that for any matrices A and B , $e^{A+B} = e^A e^B$ if and only if A and B commute. If we develop the Hamiltonian we have $H(j\delta_t) = (1 - j\delta_t)H_0 + j\delta_t H_1$ with $[H_0, H_1] \neq 0$ so the splitting of the exponential is not possible. But the Trotter-Suzuki [Trotter 1959, Suzuki 1976] formula

$$e^{A+B} = \lim_{n \rightarrow +\infty} \left(e^{\frac{A}{n}} e^{\frac{B}{n}} \right)^n$$

at first order allows us to write

$$U_T \simeq \prod_{j=0}^p e^{-i\delta_t(1-j\delta_t)H_0} e^{-ij\delta_t^2 H_1}$$

with the error in $\mathcal{O}(\|[H_0, H_1]\|)$. We see that the coefficients in front of the Hamiltonians in the exponential come from the schedule. As mentioned in the **Schedule** paragraph of Section 2.2.1, this function can be optimized to improve the efficiency. Here these coefficients are replaced by real parameters β_j, γ_j that can be optimized

with respect to a chosen target function. We call $U_{QAOA_p}(\gamma, \beta)$ the unitary generated at a given p with parameters vectors $\gamma = (\gamma_1, \dots, \gamma_p)$ and $\beta = (\beta_1, \dots, \beta_p)$ the following product:

$$U_{QAOA_p}(\gamma, \beta) = e^{-i\beta_p H_0} e^{-i\gamma_p H_1} \dots e^{-i\beta_1 H_0} e^{-i\gamma_1 H_1}$$

To see that these operators are quantum gates, we apply it to the MaxCut problem on some graph $G(V, E)$. We use the same expressions of the Hamiltonians as defined in Section 2.2.2, i.e. $H_0 = \sum_{i \in V} \sigma_x^{(i)}$ and $H_1 = \sum_{e \in E} O_e$ with $O_e = \frac{1 - \sigma_z^{(i)} \sigma_z^{(j)}}{2}$ for $e = (i, j)$, the signs of the Hamiltonians have no more importance as they are absorbed by the parameters. It is easy to see that every term in H_0 or in H_1 commutes so the corresponding exponential splits like:

$$e^{-i\beta_j H_0} = \prod_{i \in V} e^{-i\beta_j \sigma_x^{(i)}} \quad (2.7)$$

$$e^{-i\gamma_j H_1} = \prod_{e \in E} e^{-i\gamma_j O_e} \quad (2.8)$$

Each term within the product consists of local operators acting on either one or two qubits exclusively. Consequently, the resultant quantum gates are readily implementable (see Fig. 2.7).

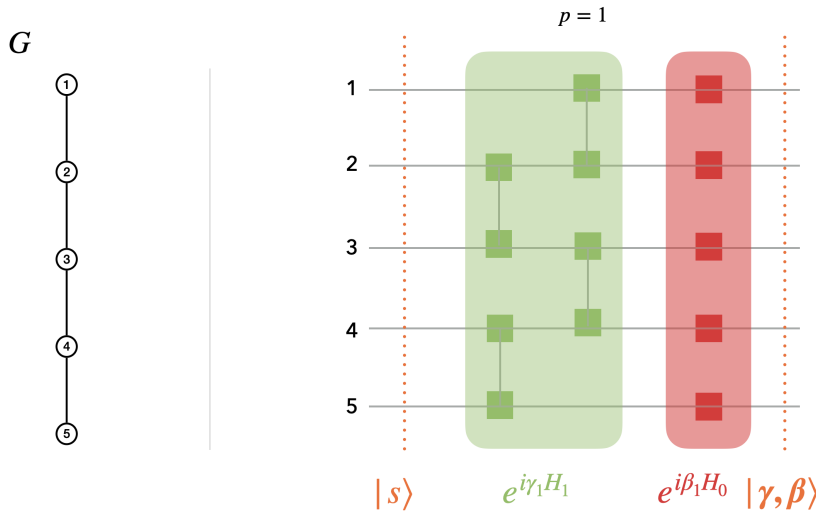


Figure 2.7: On the left the example of a graph G on which we solve MaxCut via QAOA. On the right a QAOA circuit for $p = 1$ applied to G where the two qubits gates (in green) are $e^{-i\gamma_1 O_e}$ for each edge e and the single qubit gate (in red) are $e^{-i\beta_1 \sigma_x^{(i)}}$ for each node i .

In practice to run a QAOA, we need to choose a starting state $|s\rangle$ and a target function on which optimizing the parameters. Let us call $|\gamma, \beta\rangle$ the final state of a QAOA circuit with p layers, i.e. $|\gamma, \beta\rangle = U_{QAOA_p}(\gamma, \beta)|s\rangle$. We know that in the

limit of an infinite large p , $U_{QAOA_p}(\gamma, \beta)$ can produce an adiabatic transformation of the starting state. So to keep the guarantee that the final state will converge toward a superposition of the optimal states $|x^*\rangle$, solutions of the MaxCut problem on G , the starting state is the same as in QA, meaning the ground-state of H_0 , the uniform superposition of all possible states:

$$|s\rangle = |\psi_0\rangle = \frac{1}{\sqrt{2^n}} \sum_{x \in \{0,1\}^n} |x\rangle$$

For finite p , the guarantee does not hold anymore, and the final state is a complex superposition. Complex in the sense that even for $p = 1$, it has been shown that under some complexity assumptions ($\text{PostBPP} \neq \text{PostBQP}$), finding the final probability of a given computational state classically is “Hard” [Farhi & Harrow 2019]. The chosen target function, like in the case of finite time QA, is the expected output cost value, i.e. $F_p(\gamma, \beta) = \mathbb{E}_{x \sim QAOA_p}[C(x)] = \langle \gamma, \beta | H_1 | \gamma, \beta \rangle$, where C is the classical cost function of MaxCut defined in Section 2.2.2. Let us look at how it simplifies for $p = 1$. By linearity of the expectation,

$$\begin{aligned} F_1(\gamma, \beta) &= \sum_{e \in E} \langle \gamma, \beta | O_e | \gamma, \beta \rangle \\ &= \sum_{e \in E} \langle s | e^{i\gamma_1 H_1} e^{i\beta_1 H_0} O_e e^{-i\beta_1 H_0} e^{-i\gamma_1 H_1} | s \rangle \end{aligned} \quad (2.9)$$

To simplify the formula we will use the following result. Given two observables O_X and O_Y supported on qubits in set X and Y respectively, we have that $e^{iO_Y} O_X e^{-iO_Y} = O_X$ as long as the observables commute. When the observables do not share their support, i.e. $X \cap Y = \emptyset$, it is easy to see that they commute. Using this with the expressions of the gates of Eq. (2.7), we see that for each summand, many terms cancel out ending in $\langle s | \prod_{X \sim e} e^{i\gamma_1 O_X} e^{i\beta_1 (\sigma_x^{(a)} + \sigma_x^{(b)})} O_e e^{-i\beta_1 (\sigma_x^{(a)} + \sigma_x^{(b)})} \prod_{X \sim e} e^{-i\gamma_1 O_X} | s \rangle$ with edge $e = (a, b)$ and $X \sim e$ meaning that the product is over every edges $X \in E$ that share a common node with e (see Fig. 2.8). Therefore, for a given edge e , the term only depends upon the structure at distance one from e . Using a similar reasoning for any p , we can show that the term will depends on the neighboring structure up to distance p from e .

Knowing this property of QAOA circuit at finite p , the rest of the reasoning to show the achieved approximation ratio for MaxCut over regular graphs is decomposed in a combinatorial argument followed by an optimization of the parameters (γ, β) . Indeed, a nice property of regular graphs is the finite number of local structures around a given edge. So, for a fixed degree, there is only a finite number of different terms in Eq. (2.9). This is how the authors of the original paper [Farhi et al. 2014] show the 0.6925 approximation ratio for cubic graphs. We intentionally stay quite blurry on this final step as we will follow a similar reasoning to prove our bound on the QA framework, so all details will be given later (see Part II). The main property of QAOA allowing for this straightforward proof is its **locality**.

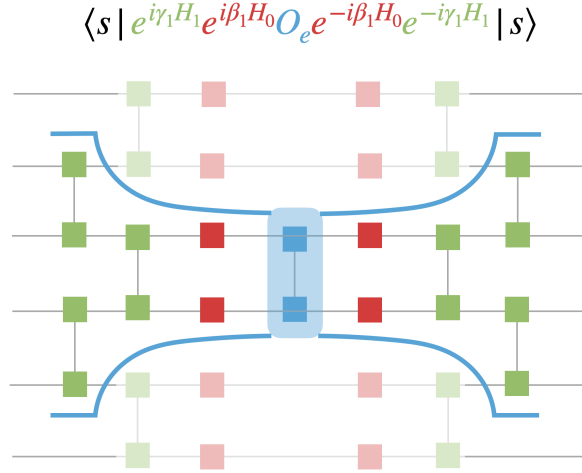


Figure 2.8: Schematic view of a term in the sum of Eq. (2.9) and how the different terms cancel each other. Two boxes of same colors in row is equal to the identity.

2.3.2 Classical local algorithms

In the section, we review the best known classical local algorithms for MaxCut. In short, for local algorithms on graphs used in distributed computing, “the output of a node in a local algorithm is a function of the input available within a constant-radius neighborhood of the vertex” (by [Suomela 2013], Suomela’s survey on local algorithms). Here we talk about an algorithm called the *threshold algorithm* [Hirvonen *et al.* 2017]. For simplicity of the analysis, the original paper restricts the input graphs to triangle-free instances. The algorithm works in one round: each node of the input graph G receives a random variable in $\{-1, +1\}$, then if the value of a node agrees with too many of its neighbors’ value, it changes its own variable. “Agree” means that they have the same initial value and “too many” is relative to a given threshold τ , a hyper-parameter that is fixed before running the algorithm. Formally, if $u_i \in \{-1, +1\}$ is the initial value of node u , the final value u_f is given by :

$$u_f = \begin{cases} -u_i & \text{if } |\{v, v \sim u \text{ and } v_i = u_i\}| \geq \tau \\ u_i & \text{otherwise.} \end{cases}$$

where $v \sim u$ means that $\langle u, v \rangle$ is an edge of G . The authors proved that it approximates MaxCut on cubic graphs with a ratio of 0.6875.

This algorithm has been relaxed to increase its performance [Hastings 2019]. Let $\mathbf{v}_0 \in [-1, 1]^n$ be the vector of the initial value for each node, i.e. node i has value $(\mathbf{v}_0)_i$. This vector is transformed into a vector $\mathbf{v}_1$ according to the following equation:

$$\mathbf{v}_1 = \mathbf{v}_0 - cJ\mathbf{v}_0$$

where J is the adjacency matrix of the input graph G . The output bi-partition is given by the sign of the final vector $\mathbf{v}_1$ for each node. By choosing $c = 0.6$, Hastings

numerically shows that it achieves an approximation ratio of 0.6980 for MaxCut over cubic graphs. Even though it is only a numerical argument, the author adds “it would also likely not be difficult to prove the performance of the classical algorithm rigorously”. It is believed that this value can be proved and we will take it as if it were. This value, obtained by a Monte Carlo sampling over 10^8 different graphs is also believed to be tight, like in the QAOA case, i.e. there are input graphs for which it achieves this value.

Remark. To understand how the Hastings algorithm is a relaxed version of the threshold algorithm, we can see that in the discrete case where $\mathbf{v}_0$ is sampled from $\{-1, +1\}^n$ and the same above update rule is used, the new value of a node i is equal to:

$$(\mathbf{v}_1)_i = (\mathbf{v}_0)_i \cdot (1 - c(2m - d))$$

where d is the degree of the graph, so $d = 3$ in the cubic case, and m is the number of i 's neighbors that agree with i . Therefore, the new value changes sign if $c(2m - d) \geq 1$, i.e. if m is large enough. This large enough m is the threshold.

This new algorithm is strictly local like QAOA and achieves a better performance than the quantum algorithm. Apparently, local quantum algorithms bring no speed-up upon the classical ones. In the next section, we review some other result about the MaxCut problem and sum up the known approximation value for cubic graphs.

2.3.3 Overview of few results

The MaxCut problem is known to be NP-hard even for regular cubic graphs [Alimonti & Kann 2000]. More precisely, it is NP-hard to approximate this problem up to some threshold. For arbitrary graphs, the best known algorithm is a semi-definite programming algorithm introduced by Goemans and Williamson [Goemans & Williamson 1995] and guarantees a 0.87856 approximation. Moreover, under the assumption of Unique Games Conjecture (UGC), the authors of [Khot *et al.* 2007] proved that for arbitrary graphs, it is NP-hard to approximate MaxCut above the Goemans-Williamson value of 0.87856. Assuming $P \neq NP$, the hardness of approximability is of $16/17 \simeq 0.94\dots$ [Håstad 2001]. When restricted to cubic graphs, or maximum degree bounded by 3, the hardness of approximability increases to 0.997 [Berman & Karpinski 1999] while the best known algorithm achieves an approximation ratio of 0.9326 [Halperin *et al.* 2002]. All of these classical algorithms run in polynomial time with respect to the input size. In Fig. 2.9, we sum up the known approximation values in different regimes.

Focusing on constant time algorithms, we have local algorithms as the one reviewed in the previous section. Basically, each node of the graph receives a random bit and updates it according to some information about its neighbors' bits. If we limit the number of update rounds to p , we call it a p -local algorithm. This approach runs in constant time only if the graph is regular so that the number of neighbors does not depend on the input size. With this method, the best classical algorithm achieves a ratio of 0.6980 MaxCut over cubic graphs

[Hastings 2019, Marwaha 2021] (see Sec. 2.3.2). It is easy to see that a random attribution of colors to the nodes gives a 0.5 approximation ratio.

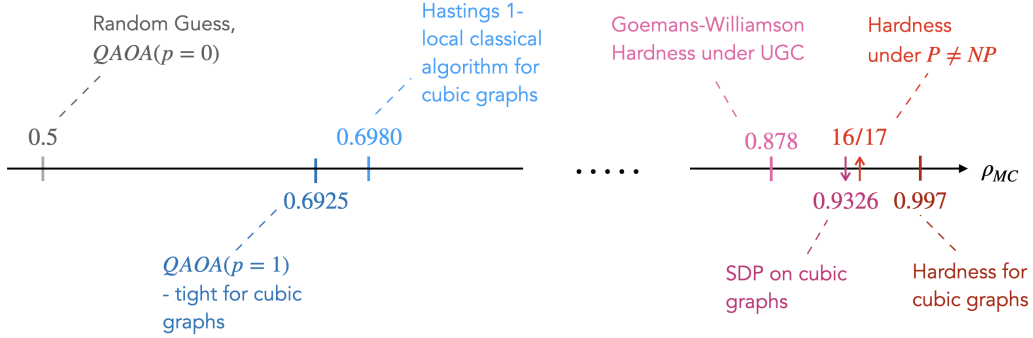


Figure 2.9: Sum up of the known approximation results and hardness value for MaxCut in general and in cubic graphs as well as the approximation ratios reach by local algorithm that run in constant time.

In the quantum framework, there are no known polynomial time approximation algorithms for combinatorial problems. If we restrict to constant time algorithms, we have the QAOA that we reviewed in Sec. 2.3.1 and its circuit depth is parametrized with p . With $p = 0$, it achieves the trivial value of 0.5 and with $p = 1$ it reaches a ratio of 0.6925 for MaxCut over cubic graph [Farhi *et al.* 2014]. This algorithm, by its construction, is a p -local algorithm. In the intermediate time scale, there are no proven results in polynomial depth, however, in logarithmic depth, the authors of [Farhi *et al.* 2020] proved that QAOA does not perform well even on random regular bipartite graphs.

In the QA setting, there was no known result apart from the adiabatic theorem that guarantees the optimal solution in exponentially long runtime. In [Moosavian *et al.* 2022], using Lieb-Robinson techniques (see Sec. 2.5), it was proved similar results to QAOA in the logarithmic time regime, i.e. QA with less than logarithmic time does not perform well, the approximation ratio for some regular bipartite graphs is upper bounded by a value below the Goemans-Williamson. In order to provide analytical proofs of the computational complexity in the AQC regime, the main theoretical tool remains the adiabatic theorem. The validity of this theorem is ruled by the minimum gap $\Delta_{\min}$ amplitude. It happens that two physical phenomena can give rise to exponentially small $\Delta_{\min}$: first-order quantum phase transition [Sachdev 1999] and Anderson localization [Anderson 1958]. These two phenomenons, when associated with exponentially closing gaps, have a typical signature when looking at the behavior of the eigen-energies during a quantum evolution and it is called “Anti-crossings” or “Avoided level-crossing” and sometimes “level repulsion”. This behavior is the subject of the next section.

2.4 Avoided level crossings (AC)

In this section, we discuss the literature around the ACs behavior as they are related to exponentially closing gaps. The early motivations behind the study of AC were the validity conditions of applicability of the adiabatic theorem [von Neumann & Wigner 1929, Wilkinson 1987, Wilkinson 1989]. Then with the rise of quantum annealing [de Falco *et al.* 1988] and adiabatic quantum computing [Farhi *et al.* 2000], proving their occurrences in a quantum evolution is a method to prove the inefficiency of AQC [Altshuler *et al.* 2010]. Inversely, proving that no such behavior occurs guarantee the efficiency of AQC [Braida *et al.* 2024a]. Also, knowing the time at which they occur during the evolution can still help to prove some quantum advantage [Roland & Cerf 2002, Amin & Choi 2009, Dalzell *et al.* 2023] as it allows to tune the schedule accordingly. The key technical tool to study avoided level-crossings is the perturbation theory, so we explain the general idea of the method and derive the main results as well as the original description. Both of them will be useful in this thesis.

2.4.1 Perturbation theory

General setting: In general, the perturbative analysis is used to study the effect a perturbation has on a system well-defined without this perturbation. For example, given two Hermitian matrices A and B , we know an eigen-pair (x, λ) of A , i.e. $Ax = \lambda x$ and we are interested in how a perturbation B will change this state. In other words, if (x, λ) represents the k^{th} eigen-pair of A , we are interested in the k^{th} eigen-pair (x_μ, λ_μ) of $A + \mu B$ for a small parameter μ . We suppose then that there exists a polynomial expansion in μ computing (x_μ, λ_μ) . We write these expansions as:

$$\begin{aligned} x_\mu &= x + x^{(1)}\mu + x^{(2)}\mu^2 + x^{(3)}\mu^3 + \dots \\ \lambda_\mu &= \lambda + \lambda^{(1)}\mu + \lambda^{(2)}\mu^2 + \lambda^{(3)}\mu^3 + \dots \end{aligned}$$

where $x^{(i)}$ and $\lambda^{(i)}$ represent the different coefficients of the polynomial expansion being respectively vectors and scalars. In practice, to be able to say something interesting, we stop the expansion at some order i . The validation of the truncation is justified by the ratio of the $(i+1)^{th}$ term over the i^{th} being small.

The different coefficients are derived iteratively by identification in the eigen-relation of the perturbed matrices. Namely, we identify each term in μ^j in the relation $(A + \mu B)x_\mu = \lambda_\mu x_\mu$. Finally, the obtained relations for each order in μ are vector equations. After choosing a right basis for the entire space (usually the eigen-vectors of A), we project along the different basis vectors each relation. Projecting along x gives the $\lambda^{(i)}$ terms and along others basis vectors gives the different coordinates of the vector $x^{(i)}$.

Application to quantum mechanics: We now apply the perturbative analysis to studying quantum system. For a full course on perturbation theory in quantum

mechanics, we refer the reader to the MIT lecture notes [Zwiebach 2018].

Let H be an Hamiltonian from which we have the full knowledge of its dynamics, meaning we know the whole spectrum respecting $H|n\rangle = E_n|n\rangle$ for the n^{th} state. We assume that the states are normalized, i.e. $\langle n|n\rangle = 1$. We are interested in the behavior of the eigen-energy and eigen-vector corresponding to the n^{th} eigen-state under the perturbation λV for a small parameter λ and an Hamiltonian V . In other words, we are looking for $E_n(\lambda)$ and $|n_\lambda\rangle$ solution of $\hat{H}_\lambda|n_\lambda\rangle = E_n(\lambda)|n_\lambda\rangle$ where $\hat{H}_\lambda = H + \lambda V$. Like presented above, we suppose that the eigen-energy and vector have an expansion in λ and we identify the different terms in the eigen-relation

$$(H + \lambda V)(|n\rangle + \lambda|n^{(1)}\rangle + \dots) = (E_n + \lambda E_n^{(1)} + \dots)(|n\rangle + \lambda|n^{(1)}\rangle + \dots)$$

Thus, by identification of the λ^j terms, we have the following relations:

$$\begin{aligned} \lambda^0: & H|n\rangle = E_n|n\rangle, \text{ already known,} \\ \lambda^1: & H|n^{(1)}\rangle + V|n\rangle = E_n|n^{(1)}\rangle + E_n^{(1)}|n\rangle \\ \lambda^2: & H|n^{(2)}\rangle + V|n^{(1)}\rangle = E_n|n^{(2)}\rangle + E_n^{(1)}|n^{(1)}\rangle + E_n^{(2)}|n\rangle \\ & \vdots \end{aligned}$$

Then we project those relations onto vector $|n\rangle$ to get the energy corrections:

- $E_n^{(1)} = \langle n|V|n\rangle$
- $E_n^{(2)} = \langle n|V|n^{(1)}\rangle + E_n^{(1)}\langle n|n^{(1)}\rangle$

The last term can be canceled because the expanded state is not assumed to be normalized and the following claim is true.

Claim. For $i \geq 1$, we can assume that vectors $|n^{(i)}\rangle$ have no component against $|n\rangle$.

Proof. Let us fix an $i \geq 1$ and suppose $|n^{(i)}\rangle$ has a component in $|n\rangle$, so we can write:

$$|n^{(i)}\rangle = |n^{(i)}\rangle_\perp + \alpha_i|n\rangle$$

where $|n^{(i)}\rangle_\perp$ is a vector orthogonal to $|n\rangle$. The expanded state $|n_\lambda\rangle$ can now be written like:

$$|n_\lambda\rangle = (1 + \alpha_1\lambda + \alpha_2\lambda^2 + \dots)|n\rangle + \lambda|n^{(1)}\rangle_\perp + \lambda^2|n^{(2)}\rangle_\perp + \dots$$

So this eigenstate will still be an eigenstate of $\hat{H}_\lambda$ if we divide it by the coefficient in front of $|n\rangle$. We call the latter state $|n_\lambda\rangle'$ and has the following expansion:

$$|n_\lambda\rangle' = |n\rangle + \lambda|n^{(1)}\rangle'_\perp + \lambda^2|n^{(2)}\rangle'_\perp + \dots$$

□

To get the first correction orders of the state, we project onto another basis state $|m\rangle$ to obtain the different coordinates:

- $\langle m|n^{(1)}\rangle = \frac{\langle m|V|n\rangle}{E_n - E_m}$
- $\langle m|n^{(2)}\rangle = \frac{1}{E_n - E_m}(\langle m|V|n^{(1)}\rangle - E_n^{(1)}\langle m|n^{(1)}\rangle)$

To develop an intuition of these expressions, we will see in Sec. 2.4.3 that they have been used to study the behaviors of the eigen-energies $E_0(s)$ and $E_1(s)$ at the end of a QA evolution. These latter anti-crossings are called perturbative crossings. Before that, we present in the next section the original description of an AC.

2.4.2 Initial description of an Anti-crossing

The goal is to retrieve, in our setting, the first results about AC given by [von Neumann & Wigner 1929] right after the first version of the adiabatic theorem [Born & Fock 1928] and later by [Wilkinson 1989].

For two Hamiltonians A and B , let $H(s)$ be a Hamiltonian with parameter $s \in [0, 1]$ of the form $(1-s)A + sB$. We note that we are in a more general setting as the standard $A + \mu B$. Let $(|\phi_j(s)\rangle, E_j(s))$ be the pairs of instantaneous eigenvector/eigen-values. We suppose that the eigen-values are ordered with $E_0(s)$ the smallest one.

Suppose that for a specific value of $s = s^*$, the gap $\Delta_{01}(s) = E_1(s) - E_0(s)$ is minimal, i.e. $\Delta_{01}(s^*) = \Delta_{\min}$ and the value of $E_0(s^*)$ and $E_1(s^*)$ are extremely close. At s^* , $H(s^*)$ is diagonal in the eigen basis span by $(|\phi_0^*\rangle, |\phi_1^*\rangle, \dots)$ where we removed the dependency in s^* and just added a star for clarity. The total Hamiltonian can be written like $H(s) = H(s^*) + (s - s^*)\dot{H}$. Here, $\dot{H} = B - A$ is the derivative of $H(s)$ with respect to s . This basis will be our working basis and the diagonal values of $H(s^*)$ are the eigen-energies $(E_0^*, E_1^*, \dots)$ and will be called the star basis.

$$H(s^*) = \begin{pmatrix} E_0^* & 0 & (\mathbf{0}) \\ 0 & E_1^* & \\ (\mathbf{0}) & & \ddots \end{pmatrix}$$

where $(\mathbf{0})$ means that the upper and lower parts are filled with zeros. We are interested the behavior of the energies $E_0(s)$ and $E_1(s)$ around $s = s^*$ when $H(s^*)$ is perturbed by the Hamiltonian $\delta_s \dot{H}$ for a small variation $\delta_s = s - s^*$. Let us call $H_{\delta_s} = H(s^* + \delta_s) = H(s^*) + \delta_s \dot{H}$. In [von Neumann & Wigner 1929, Wilkinson 1989], the authors explains that at first order, in the star basis, this Hamiltonian has the following structure:

$$H_{\delta_s} = \left(\begin{array}{cc|c} E_{0,\delta_s} & b & (\mathbf{0}) \\ b^\dagger & E_{1,\delta_s} & \\ \hline (\mathbf{0}) & & R \end{array} \right)$$

where R denotes the square matrix for the remaining Hilbert space. Importantly, we note that all matrix elements $\langle \phi_j^* | H_{\delta_s} | \phi_i^* \rangle = 0$ for $j \in \{0, 1\}$ and $i \geq 2$. In Chapter 3, we provide a rigorous proof of this statement in the demonstration of Theorem 3.1

as it does not appear in the original work. There are three last terms to explicitly describe the behavior around the crossing point s^* . It is straightforward to see that

$$\begin{aligned} E_{0,\delta_s} &= \langle \phi_0^* | H_{\delta_s} | \phi_0^* \rangle = E_0^* + \delta_s \dot{H}_{00} \\ E_{1,\delta_s} &= \langle \phi_1^* | H_{\delta_s} | \phi_1^* \rangle = E_1^* + \delta_s \dot{H}_{11} \\ b &= \langle \phi_0^* | H_{\delta_s} | \phi_1^* \rangle = \delta_s \dot{H}_{01} \end{aligned}$$

where we introduced the matrix element $\dot{H}_{ij} = \langle \phi_i^* | \dot{H} | \phi_j^* \rangle$. In Proposition 3.1, we also prove that $\dot{H}_{00} = \dot{H}_{11}$ at a crossing point. Solving the diagonalization of the two by two upper matrix yields the energy expressions:

$$E_{\delta_s}^{\pm} = \frac{E_0^* + E_1^*}{2} + \dot{H}_{00} \delta_s \pm \frac{\Delta_{min}}{2} \sqrt{1 + \left(\frac{2\dot{H}_{01}}{\Delta_{min}} \delta_s \right)^2} \quad (2.10)$$

These expressions are the first mathematical description of the eigen-energies' shape undergoing an anti-crossing. On Fig. 2.10, we draw the well-known plot of the energy behaviors around the crossing point, they take the shape of two hyperbola branches.

To close the loop, let us discuss about the different expressions we have for the approximation of the eigen-energies around the crossing point. Indeed, in this section we exploited a related but different perturbation theory, namely we reduced the problem to a 2×2 matrix and diagonalized the system. In Section 2.4.1, we presented a generic perturbation theory. Using the latter, we remark that the E_{j,δ_s} 's expressions are the first order approximation of $E_j(s)$ around s^* . How Eq.(2.10) is equivalent to the generic perturbation theory? Developing the square root like $\sqrt{1+x^2} \simeq 1 + \frac{1}{2}x^2$, under the condition of $\frac{2\dot{H}_{01}}{\Delta_{min}} \delta_s \ll 1$, yields an expression up the second order:

$$E_{\delta_s}^{\pm} \simeq E_{0,1}^* + \dot{H}_{00} \delta_s \pm \frac{\dot{H}_{01}^2}{\Delta_{min}} \delta_s^2$$

which is consistent with the second order generic perturbative expansion.

Now that the general idea of the AC phenomenon has been described properly, let us move on its application in quantum computing.

2.4.3 Perturbative crossings

In this section, we review the AC happening at the end of a QA or AQC evolution called *perturbative crossings*. There are usually attributed to the Anderson Localization effect as close to the end, the Hamiltonian $H(s)$ is close to the final one H_1 . In few words, this effect tends to localized the system in computational states (eigen-states of H_1) and if it localized in the “wrong” energy level, the gap shrinks as $\mathcal{O}(b^h)$ for some $b < 1$ and h the Hamming distance between the state where it localized and the right energy level state. Informally, let say that the ground-state of the system localized in state $y = 111000$ and the true ground-state of the final

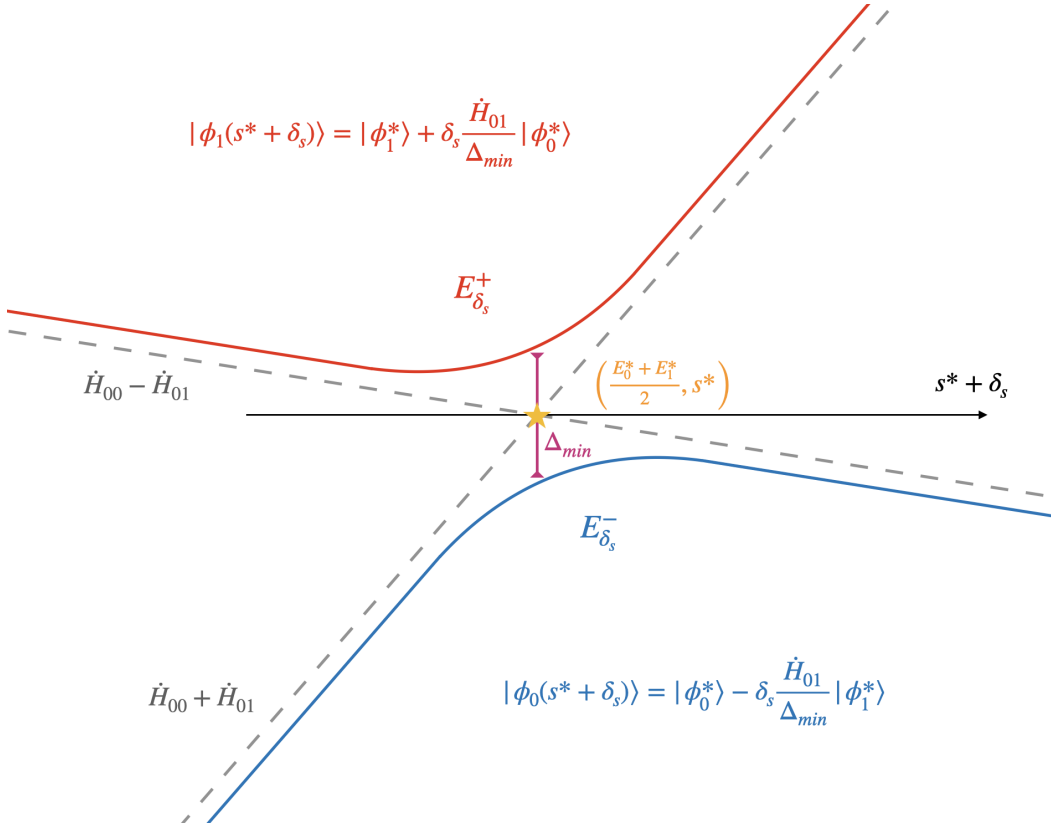


Figure 2.10: Plot of the energy shapes around the anti-crossings. Two branches of hyperbolas respecting Eq. (2.10). The orange star is the non-crossing point. In grey, the values of the asymptote slopes. The first order expansion of the associated eigen-states.

Hamiltonian is $x^* = 000111$, then it has to move from y to x^* through all states in the path that links y and x^* in H_0 . So if H_0 is the bit-flip mixing operator, then the distance is the Hamming distance between x^* and y . It results in a gap that closes exponentially fast with the system size. We will see a more detailed example in Chapter 3 and Appendix A.

The approach to show the occurrence of a perturbative crossing is to study the final Hamiltonian H_1 perturbed by the initial H_0 , i.e. the study of $\bar{H}(\lambda) = H_1 + \lambda H_0$. Without choosing a specific final Hamiltonian, we call $|\text{GS}\rangle = |\phi_0(1)\rangle$ and $|\text{FS}\rangle = |\phi_1(1)\rangle$ the ground-state and first excited state of H_1 of energy E_{gs} and E_{fs} respectively. Looking at the behaviors of $E_{0,\lambda}$ and $E_{1,\lambda}$ being respectively the perturbed expansion in λ of $E_0(1) = E_{gs}$ and $E_1(1) = E_{fs}$, can point out a real crossing of the perturbed expansion arising from an avoided crossing (see Fig.2.11)

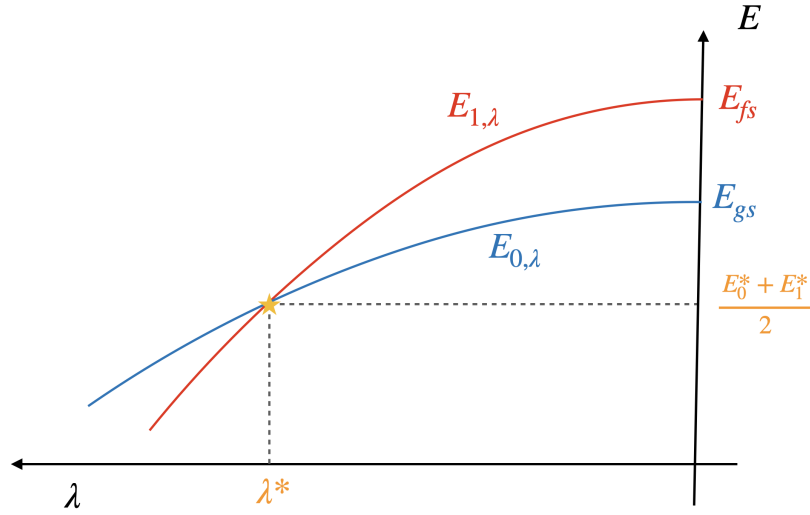


Figure 2.11: Schematic plot of a perturbative crossings. The perturbed expansions $E_{0,\lambda}$ and $E_{1,\lambda}$ cross at λ^* . For the plot, we stop at the second-order perturbation making the curves quadratic in λ .

In [Altshuler *et al.* 2010], the authors applied this method on the Exact Cover problem. This problem is NP-Hard and the encoded-problem Hamiltonian H_1 has a similar structure to the ones of Section 2.2.1. They expressed the eigen-values of $\bar{H}(\lambda)$, $E_{0,\lambda}$ and $E_{1,\lambda}$ as an expansion in λ :

$$E_{j,\lambda} = E_j(1) + \sum_{q=1}^{\infty} \lambda^q E_j^{(q)}, \quad j \in [0, 1]$$

and studied the behavior of these factors $E_j^{(q)}$ to conclude on the presence of a strong anti-crossing when AQc is used to solve Exact Cover. The difficulty when using perturbative theory is to prove the validity of the expansion. This means that if the expansion can be stopped at any finite order as long as the next term is negligible compared to the previous one. In this latter work, the authors carefully bound the

tail of the expansion allowing them to conclude on the theoretical implication of their result, i.e. AQC fails at solving Exact Cover, it gets stuck in a local minimum.

This type of perturbative crossing has also been highlighted in [Amin & Choi 2009] as a first-order quantum phase transition. It is said that this transition happens between two ordered phases only after localization. Is it the same thing as Anderson localization (AL)? How first-order phase transition and AL are related? We leave these questions to physicists. In any case, Amin and Choi used a perturbative expansion up to second order to point out the occurrence of an AC on a specific instance of the MIS problem. The condition for this expansion to hold is satisfied in their example. To understand the necessity of the second-order derivation, let us see the expressions of the eigen-values. Using the same perturbation and the formula of Section 2.4.1, we get:

$$E_{0,\lambda} = E_{gs} + \lambda \langle \text{GS} | H_0 | \text{GS} \rangle - \lambda^2 \sum_{x \neq \text{GS}} \frac{|\langle \text{GS} | H_0 | x \rangle|^2}{E_x - E_{gs}} \quad (2.11)$$

$$E_{1,\lambda} = E_{fs} + \lambda \langle \text{FS} | H_0 | \text{FS} \rangle - \lambda^2 \sum_{x \neq \text{FS}} \frac{|\langle \text{FS} | H_0 | x \rangle|^2}{E_x - E_{fs}} \quad (2.12)$$

where we abusively used the notation GS to represent the ground-state bit-string and for a bit-string x , we note E_x the energy of state $|x\rangle$. We observe that the first-order term vanishes because $\langle x | H_0 | y \rangle \neq 0$ only when x and y differ by exactly one bit. Therefore, the curves cross if the λ^2 coefficient of the first excited state is larger in norm than the coefficient of the ground-state, i.e:

$$\sum_{x \sim \text{FS}} \frac{1}{E_x - E_{fs}} > \sum_{x \sim \text{GS}} \frac{1}{E_x - E_{gs}}$$

The two sums have the same number of terms, so the inequality is satisfied if FS has lower energy neighbors than GS. This idea leads the author of [Choi 2020] to propose the “LENS” property, meaning “Low-Energy Neighboring State”. Using this insight the author suggested a new characterization of an anti-crossing that we present in the next section. Although the perturbative approach seems conclusive in some cases, one can not conclude in the absence of an AC during the evolution if not found by the perturbative method. Indeed, from [Laumann *et al.* 2015], we know that they can happen at any time. This is why V. Choi suggested a more general definition for an AC at any time in the annealing process.

2.4.4 Choi’s crossing definition

In this section, we review a new description of an avoided crossing in a QA process suggested in [Choi 2020] to tackle anti-crossings at any time in the evolution. From the states $|\phi_k(1)\rangle$ which can be degenerated and represent the states at the k^{th} energy level, V. Choi introduces the following quantities:

$$\begin{aligned} a_k(s) &= |\langle \phi_k(1) | \phi_0(s) \rangle|^2 \\ b_k(s) &= |\langle \phi_k(1) | \phi_1(s) \rangle|^2 \end{aligned}$$

They are the decomposition in the possible degenerated computational basis of the instantaneous eigenvectors corresponding to the two lowest eigenvalues. Especially, with $k = 0, 1$, we have $a_0(s)$ and $a_1(s)$ representing respectively how much of the final ground-state $|\phi_0(1)\rangle = |\text{GS}\rangle$ and the final first excited state $|\phi_1(1)\rangle = |\text{FS}\rangle$ overlaps with the instantaneous ground-state $|\phi_0(s)\rangle$, i.e. $a_0(s)$ is the probability of finding $|\text{GS}\rangle$ in $|\phi_0(s)\rangle$. The same is true for $b_0(s)$ and $b_1(s)$ with the instantaneous first excited state $|\phi_1(s)\rangle$. Let's restate her definition of a (γ, ε) -anti-crossing:

Definition 2.6. For $\gamma \geq 0$, $\varepsilon \geq 0$ we say there is an (γ, ε) -Anti-crossing if there exists a $\delta_s > 0$ such that

1. For $s \in [s^* - \delta_s, s^* + \delta_s]$,

$$\begin{aligned} |\phi_0(s)\rangle &\simeq \sqrt{a_0(s)}|\text{GS}\rangle + \sqrt{a_1(s)}|\text{FS}\rangle \\ |\phi_1(s)\rangle &\simeq \sqrt{b_0(s)}|\text{GS}\rangle - \sqrt{b_1(s)}|\text{FS}\rangle \end{aligned}$$

where $a_0(s) + a_1(s) \in [1 - \gamma, 1]$, $b_0(s) + b_1(s) \in [1 - \gamma, 1]$. Within the time interval $[s^* - \delta_s, s^* + \delta_s]$, both $|\phi_0(s)\rangle$ and $E_1(s)$ are mainly composed of $|\text{GS}\rangle$ and $|\text{FS}\rangle$. That is, all other states (eigenstates of the problem Hamiltonian H_1) are negligible (which sums up to at most $\gamma \geq 0$).

2. At the avoided crossing point $s = s^*$, $a_0, a_1, b_0, b_1 \in [1/2 - \varepsilon, 1/2 + \varepsilon]$, for a small $\varepsilon > 0$. That is, $|\phi_0(s^*)\rangle \simeq 1/\sqrt{2}(|\text{GS}\rangle + |\text{FS}\rangle)$ and $|\phi_1(s^*)\rangle \simeq 1/\sqrt{2}(|\text{GS}\rangle - |\text{FS}\rangle)$.
3. Within the time interval $[s^* - \delta_s, s^* + \delta_s]$, $a_0(s)$ increases from $\leq \gamma$ to $\geq (1 - \gamma)$, while $a_1(s)$ decreases from $\geq (1 - \gamma)$ to $\leq \gamma$. The reverse is true for $b_0(s), b_1(s)$.

This definition of an anti-crossing gives more insights toward the understanding of this physical phenomenon happening during an adiabatic evolution. Four new quantities, $a_0(s)$, $a_1(s)$ and $b_0(s)$, $b_1(s)$, and two parameters (γ, ε) are at stake here, to describe it. Definition 2.6 presents in a precise way how the quantities vary through the anti-crossing (see Fig.2.12) and implicitly suggests on the size of the parameters that the smaller they are, the stronger the anti-crossing will be. However, there is a missing result that directly links this definition of an anti-crossing to the min-gap. How the parameters (γ, ε) influence the min-gap? In Chapter 3, we answer this question by providing a rigorous proof the the minimum gap expressed in terms of a_k and b_k .

In this section, we sorted out the literature around the avoided crossing phenomenon after a general introduction to the perturbative theory which is the main technical tool to study the ACs. The understanding of the AC is an active research area as it indicates the computational complexity of the problem we are dealing with. There are still no general approaches to qualify the phenomena apart from the exponentially closing gap at the crossing point. Furthermore, the presence of such AC allows to conclude on the exponential time complexity to solve the problem

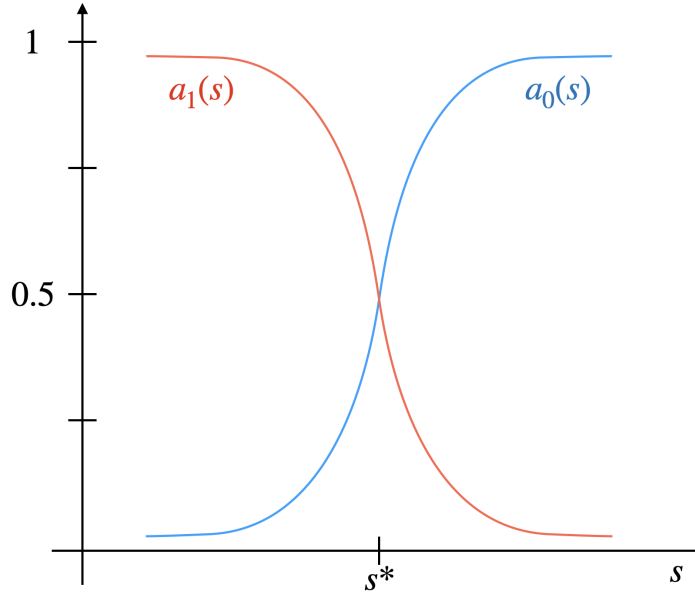


Figure 2.12: Evolution of $a_0(s)$ and $a_1(s)$ around a (γ, ε) –anti-crossing point s^* . The same goes for $b_0(s)$ and $b_1(s)$ but inverted so that $b_1(s)$ ends close to 1.

but does not give insights on the performance of a quantum evolution to approximate it. Can we still guarantee some performance of QA if one fixes the time away from the adiabatic regime? The other technical tool that is known to study a quantum evolution is the Lieb-Robinson bound.

2.5 Lieb-Robinson bound

In this section, we present a result from theoretical physics that has numerous application in computer science. Informally, it states as follows.

Theorem 2.2 (Lieb-Robinson bound). *There is an upper bound on the velocity v_{LR} at which information can propagate in a quantum system.*

It has been first proved in [Lieb & Robinson 1972], but it is only after a generalization of a result in higher dimension in [Hastings 2004] that uses the LR bound, that Theorem 2.2 gained more and more attention.

2.5.1 Observation and intuition

Let us first develop the intuition and some observations that are useful to understand how we will use this bound. By Theorem 2.2, the finite velocity v_{LR} implies at finite time T that the correlation of two distant subsystems is very small. Here the notion of distance is defined by the Hamiltonian connectivity acting on the quantum system. For simplicity, we suppose that the Hamiltonian H_G is 2–local and the connectivity

is described by a graph G . So the distance is relative to G . For instance, on a line graph, the qubits at the extremity of the line are weakly correlated for some time. On a general graph G , the runtime required to correlate every qubits is of order $\frac{D}{v_{LR}}$, where D is the diameter of the graph.

Another point of view of this result is to look at the evolution of an observable. If $|\psi_T^G\rangle$ is a quantum system that evolved under H_G for time T , the final expectation value of a local observable O_X supported on X depends mainly on the neighboring configuration Ω of X up to distance $p \simeq v_{LR}T$. This means that a quantum system that evolves under the Hamiltonian restricted to the subgraph Ω , noted H_Ω produces a final expected value of the observable O_X very close to the one with the whole quantum system.

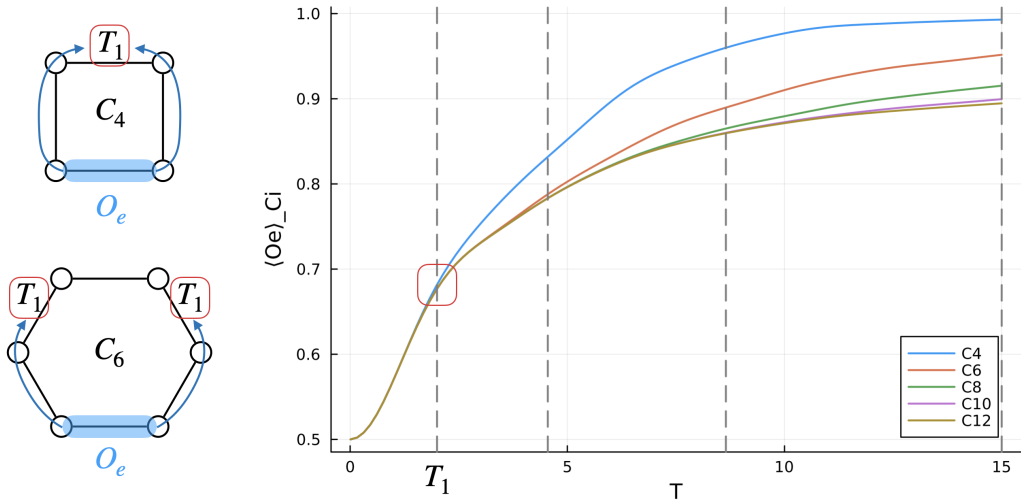


Figure 2.13: On the right, the plot of the evolution of the expected edge e energy for different size l of cycle C_l . The dashed lines in light grey point out the splitting time when the remaining curve splits from the others with a difference less than 0.005. On the left, cycle of size 4 and 6 with the representation of the time evolution of the support of the evolved observable O_e up to time T_1 , indicated by the red box.

To illustrate this phenomenon, let's put ourselves in the QA framework to solve the MaxCut problem over even cycles, as defined previously in Section 2.2.2. We restrict the input graphs to even cycle C_l of size l . The total Hamiltonian is written $H(C_l, s) = -(1-s) \sum_i \sigma_x^{(i)} - s \sum_e O_e$ where for any edge $e = (a, b)$, the observable O_e encodes the MaxCut cost function, i.e. for a bit-string state $|x\rangle \in \{0, 1\}^l$, $\langle x | O_e | x \rangle = 1$ if and only if $x_a \neq x_b$ and 0 otherwise. The initial state $|\psi_0\rangle$ of the evolution being the uniform superposition over all bit-strings of length l , the initial expected value of the edge e is $\langle \psi_0 | O_e | \psi_0 \rangle = 0.5$ as there is the same amount of computational states that cut edge e ($x_a \neq x_b$) and uncut it. After some time T , the final expected value is $\langle O_e \rangle_{C_l} = \langle \psi_T^{C_l} | O_e | \psi_T^{C_l} \rangle$. We note that in the limit of large T , this latter value converges toward 1 as every edge e can be cut in an even cycle, and the adiabatic theorem guarantees that the final state converges toward

the optimal state $|x^*\rangle$. On Fig. 2.13 we plotted the evolution of $\langle O_e \rangle_{C_l}$ for different values of l against the total runtime T . We observe that all curves start at 0.5 as expected and then follow the same trajectory up to some time T_1 where the value for C_4 starts to increase toward 1 faster than all the others that continue on the same trajectory for a while. Then, we can see similar splitting point later between the smallest remaining cycle and the others. We can infer that until time T_1 , $\langle O_e \rangle_{C_l}$ remains independent of the cycle size. Consequently, the quantum evolution has the capability to differentiate between C_4 and the other cycles. The underlying intuition drawn from this observation suggests that information must propagate around the cycle, and up to T_1 , the finite velocity v_{LR} implies that examining a local observable does not provide sufficient information to distinguish among all cycles. The knowledge of the graph structure at this point is limited. At time T_1 , edge e of C_4 starts to gain the knowledge that it is inside a cycle whereas the edge of the other cycle could potentially be in the middle of a long path. The drawings on the left of Fig. 2.13 try to convey this idea by showing that the edge e is looking around him and information comes at a finite speed, so the longer you let the system evolves the further in the graph e can “see”. The knowledge of belonging to an even cycle helps e to take a better decision faster, and thus converges toward 1.

This illustration is only a toy example with an informal tone to communicate some intuition behind Theorem 2.2. In the next section we dive into the mathematical derivation of the bound.

2.5.2 Mathematical derivation

The original derivation of the LR bound is an upper bound on the norm of a commutator between an evolved local observable $A(t)$ and a static one B . It is assumed that the support Y of B and the support X of $A = A(0)$ are at distance p (see Fig. 2.14). If U_t^G is the unitary solution of the Schrodinger equation under H_G (here G can be an hyper-graph), then the evolved observable is written $A(t) = (U_t^G)^\dagger A U_t^G$. Because X and Y are distant subsets of G , we have that $[A, B] = 0$. But for positive time t , in general $[A(t), B] \neq 0$. The LR bound gives an upper limit on the norm of this commutator and shows that it is exponentially small with the distant p , namely, there exists a constant v_{LR} and two real scalars a and b such that

$$\|[A(t), B]\| \leq a e^{-b(p-v_{LR}t)}$$

We see from this equation that the bound is exponentially small with p and exponentially large with t . This also means that for large t the bound becomes too loose to be meaningful. We will not prove the bound in details but we show how the different steps fit into each other by following these Lecture Notes [Hastings 2010].

The first derived LR bounds were in the time independent case. Let us work with a general time-independent Hamiltonian of the form $H = \sum_Z \gamma_Z$ where the sum is over the subset Z of qubits on which the local operator γ_Z acts. It can be

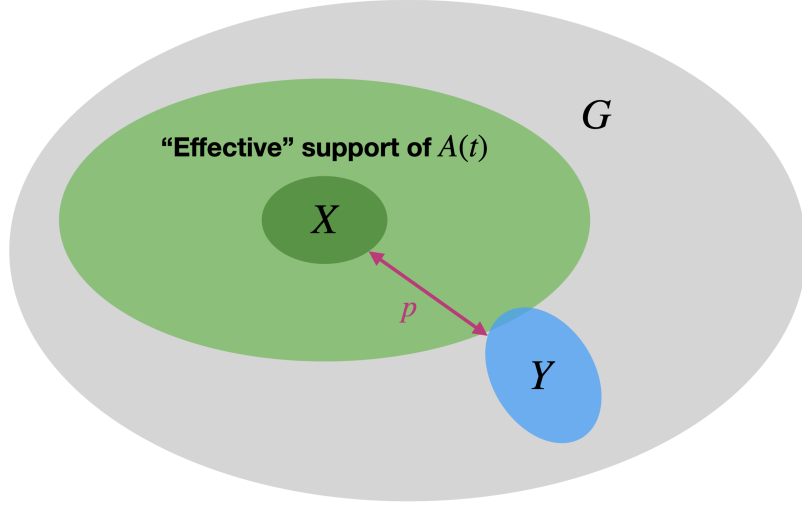


Figure 2.14: Setting for LR bound derivation.

shown that the derivative of the commutator of interest respects:

$$\frac{d\|[A(t), B]\|}{dt} \leq 2\|A\| \sum_{Z: Z \cap X \neq \emptyset} \|\gamma_Z(t), B\|$$

where $\gamma_Z(t)$ is the evolved operator γ_Z . Integrating this equation gives

$$\|[A(t), B]\| - \|[A(0), B]\| \leq 2\|A\| \sum_{Z: Z \cap X \neq \emptyset} \int_0^t dt_1 \|\gamma_Z(t_1), B\|$$

Now we see that we can apply this inequality repeatedly to bound the right-hand side. So knowing that $[A, B] = 0$, the first iterations give

$$\begin{aligned} \frac{\|[A(t), B]\|}{2\|A\|} &\leq \sum_{Z_1: Z_1 \cap X \neq \emptyset} \int_0^t dt_1 \|\gamma_{Z_1}(0), B\| \\ &+ 2 \sum_{Z_1: Z_1 \cap X \neq \emptyset} \sum_{Z_2: Z_2 \cap Z_1 \neq \emptyset} \|\gamma_{Z_1}\| \int_0^t dt_1 \int_0^{t_1} dt_2 \|\gamma_{Z_2}(t_2), B\| \\ &\leq 2t\|B\| \sum_{Z_1: X \sim Z_1 \sim Y} \|\gamma_{Z_1}\| \\ &+ 2^2\|B\| \frac{t^2}{2} \sum_{Z_1, Z_2: X \sim Z_1 \sim Z_2 \sim Y} \|\gamma_{Z_1}\| \|\gamma_{Z_2}\| \\ &+ 2^3\|B\| \frac{t^3}{3!} \sum_{Z_1, Z_2, Z_3: X \sim Z_1 \sim Z_2 \sim Z_3 \sim Y} \|\gamma_{Z_1}\| \|\gamma_{Z_2}\| \|\gamma_{Z_3}\| + \dots \end{aligned}$$

where the notation $X \sim Z_1 \sim \dots \sim Z_k \sim Y$ means that the intersection two by two is not the empty set, i.e. $Z_1 \cap X \neq \emptyset, \forall i \in [1, k-1], Z_i \cap Z_{i+1} \neq \emptyset$ and $Z_k \cap Y \neq \emptyset$.

Now to pursue the proof we need to add an assumption about the interaction in the Hamiltonian of interest. From a physical point of view, in the most general case one can use a law of interaction to bound the different sums appearing in the expression. Being exponential or power decay interaction, it is enough to finish the demonstration [Haah *et al.* 2021, Tran *et al.* 2019]. In the case of interest in this thesis, namely the combinatorial graph problems, the interactions are finite range as we only have bounded 2-local operators in the Hamiltonian (see Section 2.2). So like in [Moosavian *et al.* 2022], the sum of the r^{th} term is over paths of length r from X to Y and the first $p - 1$ terms are null. We are left with the last term, i.e.

$$\frac{\| [A(t), B] \|}{2\|A\|} \leq 2^p \|B\| \frac{t^p}{p!} (2(d-1))^p$$

where we bounded the number of paths of length p from X to Y by $(2(d-1))^p$ if G is a d -regular graphs. Then using a Stirling formula or an astute manipulation of the exponential like in [Chen *et al.* 2023], we get the wanted expression of the LR bound.

2.5.3 Some known applications

The Lieb-Robinson bound has brought a growing interest since the result of [Hastings 2004] as it is used to generalize a physical result at higher dimension, namely the Lieb-Schultz-Mattis theorem [Lieb *et al.* 1961].

From a computer science point of view, the first application of this bound is in [Haah *et al.* 2021] and later generalized in [Tran *et al.* 2019] to produce a quantum circuit for Hamiltonian simulation. It improves on the gates complexity and the depth of the circuit upon the previous best algorithms based on the Lie-Trotter-Suzuki formula [Berry *et al.* 2006], or truncated Taylor series [Low & Chuang 2017].

More recently, in [Moosavian *et al.* 2022], the authors used the Lieb-Robinson bound to prove some upper bound on the reachable approximation ratio on MaxCut by a quantum annealing evolution when the runtime is at most logarithmic in the input size. They claim to prove the first time-dependent LR bound. The outcome of their result is an upper bound on the performance of short time QA closely related to the result in [Farhi *et al.* 2020] about QAOA. They show that on a specific class of bipartite regular graphs, the approximation ratio for MaxCut is upper bounded by a value that goes below the Goemans-Williamson ratio of 0.87856 when the degree exceeds 6. In plain English, this means that QA in logarithmic time cannot beat Goemans-Williamson's algorithm for MaxCut.

For a more extensive review about the Lieb-Robinson bound, we refer the reader to [Chen *et al.* 2023].

Part I

Anti-crossings in Adiabatic Quantum Computing

Anti-crossings characterization

Contents

3.1 Results and limitations of Def. 2.6	67
3.1.1 Minimum gap in AC of Def. 2.6	68
3.1.2 AC in Maximum-Weight k -clique problem	71
3.2 New characterization of anti-crossing	75
3.2.1 Variation of $ \phi_0(s)\rangle$ at s^*	75
3.2.2 New anti-crossing definition	77
3.2.3 Thorough comparison of ACs definitions	78
3.3 Limitation and other potential type of AC	82

In this chapter, we present our contribution in the understanding of the avoided-crossing phenomenon in [AQC](#). Recall that in [AQC](#) we suppose that the state is evolved adiabatically, meaning, among other things, that the runtime is ruled by the minimum gap closing, while in [QA](#) the runtime is chosen arbitrarily. For this reason, [AC](#) phenomenon are of first interest when studying [AQC](#) performance. Here, we show some results and limitations of the most recent definition by [\[Choi 2020\]](#) (Sec. 3.1). Those observations lead us to propose a new definition of an [AC](#) encompassing more cases. We support our new characterization with a numerical analysis on a toy example for the maximum (weight) k -clique problem (Sec. 3.2).

3.1 Results and limitations of Def. 2.6

In this section, we discuss the recent definition of an anti-crossing introduced in [\[Choi 2020\]](#), that we detailed in the Preliminaries Chapter in Sec. 2.4.4. Recall that the study of this phenomenon is interesting in [AQC](#) as it comes with an exponentially closing minimum gap of the total Hamiltonian. Improving the characterization of an [AC](#) is a major challenge in this setting to understand the time complexity of [AQC](#). We will show how Def. 2.6 is indeed linked to a small gap. Then using a toy model of the maximum weighted k -clique problem, we show how this definition is limited to some specific [AC](#).

As defined in Sec. 2.2, we focus on total Hamiltonian of the form $H(s) = (1 - s)H_0 + sH_1$, with H_0 the initial Hamiltonian and H_1 the problem Hamiltonian. Let's state a useful lemma that applies in this setting:

Lemma 3.1. *Let $\lambda_i, |n_i\rangle$ be an eigenvalue/eigenvector orthogonal pair of a real-valued symmetric time dependent matrix operator B such that $\ddot{B} = 0$, we have:*

$$\dot{\lambda}_i = \langle n_i | \dot{B} | n_i \rangle \quad (3.1)$$

$$|\dot{n}_i\rangle = \sum_{j \neq i} \frac{\langle n_j | \dot{B} | n_i \rangle}{\lambda_i - \lambda_j} |n_j\rangle \quad (3.2)$$

$$\ddot{\lambda}_i = 2 \sum_{j \neq i} \frac{|\langle n_j | \dot{B} | n_i \rangle|^2}{\lambda_i - \lambda_j} \quad (3.3)$$

Proof. We take the derivative of the eigen relation $B|n_i\rangle = \lambda_i|n_i\rangle$:

$$\dot{B}|n_i\rangle + B|\dot{n}_i\rangle = \dot{\lambda}_i|n_i\rangle + \lambda_i|\dot{n}_i\rangle$$

and knowing that $\langle n_i | \dot{n}_i \rangle = 0$ because $\langle n_i | n_i \rangle = 1$, we compose by $\langle n_i |$ on the left to get the first expression. Then, composing the same expression by another eigenvector $|n_j\rangle$ with eigenvalue λ_j we obtain $(\lambda_i - \lambda_j)\langle n_j | \dot{n}_i \rangle = \langle n_j | \dot{B} | n_i \rangle$. From that we get the second expression. Eventually, we take the second derivative of the eigen relation:

$$\ddot{B}|n_i\rangle + 2\dot{B}|\dot{n}_i\rangle + B|\ddot{n}_i\rangle = \ddot{\lambda}_i|n_i\rangle + 2\dot{\lambda}_i|\dot{n}_i\rangle + \lambda_i|\ddot{n}_i\rangle$$

by hypothesis $\ddot{B} = 0$, then projecting onto $|n_i\rangle$ and using (3.2) we obtain the third result. $\square$

In our setting, the second derivative of the Hamiltonian $H(s)$ with respect to s is zero, so the results of this latter lemma are true for the Hamiltonian $H(s)$.

3.1.1 Minimum gap in AC of Def. 2.6

Let us first recall quickly the idea of Def. 2.6 crossing. Around anti-crossing point s^* , the instantaneous ground-state $|\phi_0(s^*)\rangle$ and first excited state $|\phi_1(s^*)\rangle$ are mainly localized in the final ground-state $|\text{GS}\rangle$ and final first excited state $|\text{FS}\rangle$. With $a_k(s) = |\langle \phi_0(s) | \phi_k(1) \rangle|^2$ and $b_k(s) = |\langle \phi_1(s) | \phi_k(1) \rangle|^2$, the definition states that there exists δ_s such that in the interval $[s^* - \delta_s, s^* + \delta_s]$:

$$\begin{aligned} |\phi_0(s)\rangle &\simeq \sqrt{a_0(s)}|\text{GS}\rangle + \sqrt{a_1(s)}|\text{FS}\rangle \\ |\phi_1(s)\rangle &\simeq \sqrt{b_0(s)}|\text{GS}\rangle - \sqrt{b_1(s)}|\text{FS}\rangle \end{aligned}$$

where the four quantities a_0 , a_1 , b_0 and b_1 have well defined behaviors parametrized by γ and ε . γ represents the maximal amount of states other than $|\text{GS}\rangle$ and $|\text{FS}\rangle$, i.e. in the same interval, $\sum_{k \geq 2} a_k(s) < \gamma$ (similarly for the b_k 's). ε represents the value by which the latter four quantities deviate from 1/2 at crossing point s^* (see Fig. 2.12). Typically, $a_0(s)$ will grow from below γ to above $1 - \gamma$ passing through $\frac{1}{2} \pm \varepsilon$ at $s = s^*$, and inversely for $a_1(s)$. The same goes for b_1 and b_0 .

Now, recall that the quantity of interest for AQCC is the behavior of the minimum gap $\Delta_{\min}$. Few examples are known where the expression of the gap can be found

explicitly [Albash & Lidar 2018]. It is interesting to understand the influence of parameter γ and ε on the minimum gap value. We derive here the exact expression (Proposition 3.1) of $\Delta_{\min}$ using the quantities introduced in [Choi 2020].

Proposition 3.1. $\Delta_{\min} = \sum_k E_k(1) [b_k(s^*) - a_k(s^*)]$ where the sum runs over indices k corresponding to distinct $E_k(1)$.

Proof. The gap reaches a minimum at s^* so its derivative is null at s^* , thus using Eq. (3.1) of Lemma 3.1, we get:

$$\begin{aligned} 0 &= \frac{d\Delta}{ds}(s^*) \\ &= \frac{dE_1}{ds}(s^*) - \frac{dE_0}{ds}(s^*) \\ &= \langle \phi_1(s^*) | \dot{H} | \phi_1(s^*) \rangle - \langle \phi_0(s^*) | \dot{H} | \phi_0(s^*) \rangle \end{aligned}$$

$$\begin{aligned} \Rightarrow \langle \phi_1(s^*) | H_0 | \phi_1(s^*) \rangle - \langle \phi_0(s^*) | H_0 | \phi_0(s^*) \rangle &= \langle \phi_1(s^*) | H_1 | \phi_1(s^*) \rangle \\ &\quad - \langle \phi_0(s^*) | H_1 | \phi_0(s^*) \rangle \end{aligned}$$

In the setting of linear interpolation, we have $\dot{H} = H_1 - H_0$. The eigen relation takes the form of $s\dot{H}|\phi_k(s)\rangle = E_k(s)|\phi_k(s)\rangle - H_0|\phi_k(s)\rangle$. Then projecting onto $\langle \phi_k(s) |$ gives $s\langle \phi_k(s) | \dot{H} | \phi_k(s) \rangle = E_k(s) - \langle \phi_k(s) | H_0 | \phi_k(s) \rangle$. Applying this relation with $k = 0, 1$ on the previous calculation, gives another expression of the min-gap:

$$\begin{aligned} \Delta_{\min} &= E_1(s^*) - E_0(s^*) \\ &= \langle \phi_1(s^*) | H_0 | \phi_1(s^*) \rangle - \langle \phi_0(s^*) | H_0 | \phi_0(s^*) \rangle \\ &= \langle \phi_1(s^*) | H_1 | \phi_1(s^*) \rangle - \langle \phi_0(s^*) | H_1 | \phi_0(s^*) \rangle \end{aligned}$$

Now, let's write $|\phi_1(s^*)\rangle$ and $|\phi_0(s^*)\rangle$ in the computational basis $|x\rangle$. There exists α_x and β_x such that

$$|\phi_0(s^*)\rangle = \sum_{x \in \{0,1\}^n} \alpha_x |x\rangle \quad \text{and} \quad |\phi_1(s^*)\rangle = \sum_{x \in \{0,1\}^n} \beta_x |x\rangle$$

By definition of a_k and b_k , the different eigen subspace can be degenerate, so we have that

$$\sum_{x \text{ s.t. } E_x = E_k(1)} |\alpha_x|^2 = a_k(s^*) \quad \text{and} \quad \sum_{x \text{ s.t. } E_x = E_k(1)} |\beta_x|^2 = b_k(s^*)$$

Then knowing that $H_1|x\rangle = E_x|x\rangle$ and $E_x = E_k(1)$ is the same for all vectors in the subspace span by $|\phi_k(1)\rangle$, we have:

$$\begin{aligned} \langle \phi_0(s^*) | H_1 | \phi_0(s^*) \rangle &= \sum_x \sum_y \alpha_x^\dagger \alpha_y \langle x | H_1 | y \rangle \\ &= \sum_x |\alpha_x|^2 E_x \\ &= \sum_k a_k(s^*) E_k(1) \end{aligned}$$

The same goes with $|\phi_1(s^*)\rangle$ and $b_k(s^*)$ therefore we end up with:

$$\Delta_{min} = \sum_k E_k(1) [b_k(s^*) - a_k(s^*)]$$

□

This result is a general expression of the minimum gap when linear interpolation is used. We can see that it depends only on the new variables and the final energies. The minimum gap will be exponentially small if all terms a_k and b_k are small or if $a_k \simeq b_k$. We can now upper-bound Δ_{min} using the parameter ε that quantifies the strength of the anti-crossing.

Corollary 3.1. *For a (γ, ε) -anti-crossing of definition 2.6, we have:*

$$\Delta_{min} = \mathcal{O}(\varepsilon)$$

Proof. For a (γ, ε) -anti-crossing of definition 2.6, $a_0(s^*), a_1(s^*), b_0(s^*), b_1(s^*) \in [1/2 - \varepsilon, 1/2 + \varepsilon]$. Thus, for $k = 0, 1$, $|b_k(s^*) - a_k(s^*)| \leq 2\varepsilon$. This also means that $\sum_{k \geq 2} a_k(s^*) \leq 2\varepsilon$, and the same goes for the $b_i(s^*)$'s. The final energy $E_k(1)$ is upper-bounded by a constant M and without loss of generality we assume $E_k(1) \geq 0$, therefore:

$$\begin{aligned} \Delta_{min} &\leq E_0(1)|b_0(s^*) - a_0(s^*)| + E_1(1)|b_1(s^*) - a_1(s^*)| \\ &\quad + M \sum_{k \geq 2} [b_k(s^*) + a_k(s^*)] \\ &\leq 2\varepsilon E_0(1) + 2\varepsilon E_1(1) + 4\varepsilon M \\ &\leq 2(E_0(1) + E_1(1) + 2M)\varepsilon \end{aligned}$$

□

The smaller ε is, the smaller the minimum gap will be. Therefore, if the instantaneous ground-state is exponentially close to a linear combination of $|\text{GS}\rangle$ and $|\text{FS}\rangle$ (i.e. ε is exponentially small), the minimum gap will be exponentially small. This new result quantifies the strength of a Def. 2.6 anti-crossing.

Discussion: Let us discuss some intuition we can get from the quantities a_k and b_k based on Def. 2.6. Around crossing point, a_1 is dominant before s^* , meaning that the instantaneous ground-state is localized in the first final excited state before the anti-crossing. After s^* , a_0 is dominant, meaning that $|\phi_0(s)\rangle$ is now localized in the final ground-state. This exchange of localization happens in pairs with the instantaneous excited state $|\phi_1(s)\rangle$ as it jumps from $|\text{GS}\rangle$ toward $|\text{FS}\rangle$. This implies that before s^* , $|\phi_0(s)\rangle$ was heading toward $|\text{FS}\rangle$, and $|\phi_1(s)\rangle$ toward $|\text{GS}\rangle$.

One can understand the a_k 's as being the direction toward which the instantaneous ground-state is evolving. For a particular j , if a_j is becoming dominant at some point in the evolution, it means that $|\phi_0(s)\rangle$ is going toward the j^{th} final

energy state. The description imposed by Def. 2.6 seems restrictive as it is only focused on a_0 crossing with a_1 , but in theory, a_0 could cross with any other level a_j . One could question if the imposed property is universal for an AC in general. The same intuition holds for the b_k 's and the instantaneous first excited state $|\phi_1(s)\rangle$. Although Def. 2.6 relates the anti-crossings to the gap closing via Prop. 3.1, is it possible to exhibit a counter-example to the definition helped by the intuition we provided on the role of the a_k 's? We answer this question in the following section.

3.1.2 AC in Maximum-Weight k -clique problem

To validate or invalidate Def. 2.6 of an AC, we look at the maximum-weight k -clique problem, with the Hamiltonians described in Sec. 2.2.4. We recall that the goal is to find a clique of size k in a given input weighted graph $G = (V, E, w)$ that maximizes the weight of the clique. w is a vector such that w_i is the weight of node i . The initial Hamiltonian H_0 stabilizes the Hilbert space span by bit-strings of Hamming weight k , and we choose the path graph as the driver graph, meaning that H_0 links two classical states if they differ by exactly one swap. We choose the path graph as it seems that it makes AQC struggle more than other tested driver graph (see Appendix A). In the weighted setting, the cost of a bit-string (necessarily of Hamming weight k) is the number edges missing to form a clique minus the weights of the corresponding nodes. As introduced in Sec. 2.2.4, we add a parameter α in front of the total weight to play with the importance of the weights.

The toy example we constructed to illustrate the anti-crossing phenomenon is a graph on $n = 6$ vertices and $|E| = 7$ edges and the size of the clique we search is of size $k = 3$ (see Fig. 3.1). Let $w = [1.0, 1.0, 1.0, 1.5, 1.5, 1.5]$ be the list of weights of the six nodes. Using this weight vector, for $\alpha < 2/3$, the ground-state of H_1 is $|\text{GS}\rangle = |111000\rangle$ with energy $0 - 3\alpha$ and the first excited state $|\text{FS}\rangle = |000111\rangle$ with energy $1 - 4.5\alpha$ (for $\alpha = 0$, there are many first excited states). The ground-state is degenerate for $1 - 4.5\alpha = -3\alpha$ i.e. $\alpha = 2/3$. It is interesting to note that these latter states are far apart in the graph G_{H_0} generated by $-H_0$. Indeed with a path graph as driver graph, G_{H_0} is similar to the one presented in Fig. 2.6 with a kite configuration. The distance in this graph between $|\text{GS}\rangle$ and $|\text{FS}\rangle$ is the diameter of the graph, i.e. the largest distance between two states. As we will see in Sec. 4.2, this property is an indicator for small minimum gap.

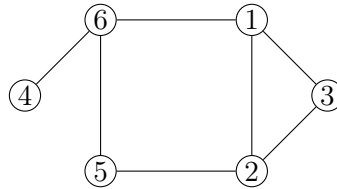


Figure 3.1: Toy example 1 with weights $[1,1,1,1.5,1.5,1.5]$ for each node. For $\alpha < 2/3$, the solution is the triangle with nodes labeled 1, 2 and 3. For $\alpha > 2/3$, the solution is the subgraph $\{4,5,6\}$.

Now let us have a look at the evolution of the different quantities occurring during an anti-crossing to compare two instances, $\alpha = 0$ and $\alpha = 0.5$. First, let us explicitly give the final energies of the different states (Fig. 3.2). We see that in the case of $\alpha = 0$, the first excited state is degenerate with 8 states whereas, with $\alpha = 0.5$, both the ground and first excited states are non-degenerate.

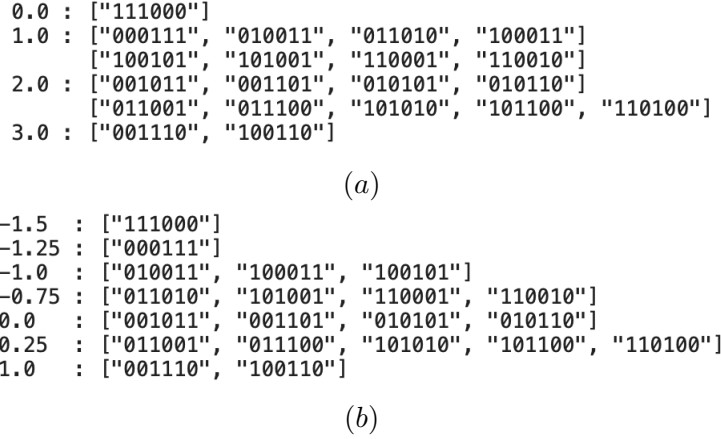
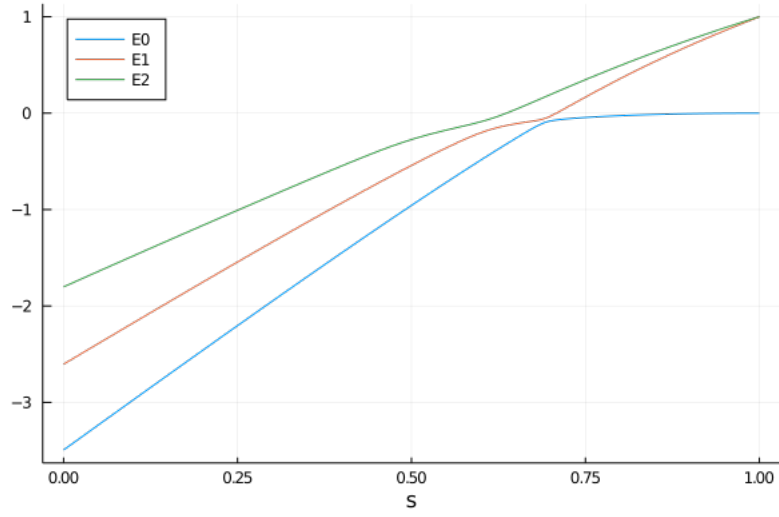


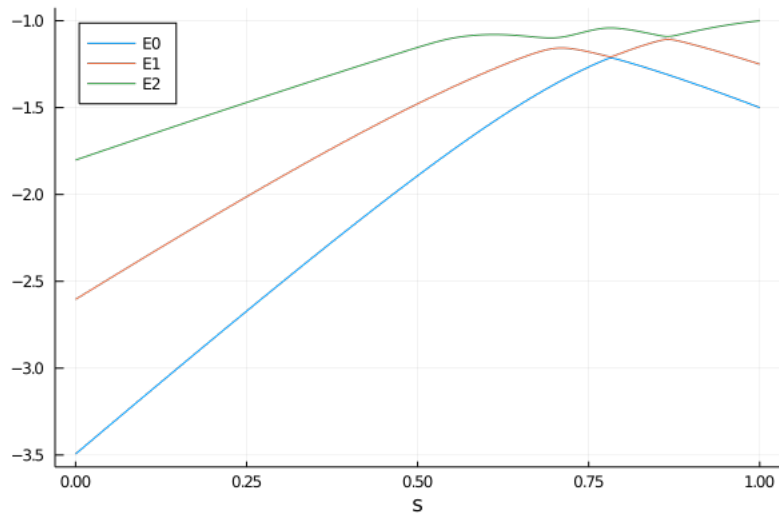
Figure 3.2: States with energies for toy example 1 for $\alpha = 0$ (a) and $\alpha = 0.5$ (b)

In the following analysis, we call anti-crossing the two hyperbola branches described by von Neuman and Wigner (see Sec. 2.4.2) that can happen between any two successive eigenvalues and we denote by s^* the anti-crossing point between the two lowest eigenvalues. We plot the evolution of the first three eigenvalues for the two cases of interest. On Fig. 3.3 (a), two types of anti-crossings are distinguishable: the one between E_2 and E_1 which is quite weak and the one between E_1 and E_0 which is strong and of interest in AQC. Here, the slope of $E_0(s)$ before s^* is the slope of $E_1(s)$ after s^* . Noticing that E_1 does not undergo another AC after, we can say that the lowest energy was going toward $E_1(1)$ before bouncing against the second lowest energy to redirect toward $E_0(1)$. If we follow the intuition given in the previous section, we can expect to see a_0 crossing a_1 .

Now, looking at Fig. 3.3 (b), we see that E_2 and E_1 are getting closer before and even more after the anti-crossing between E_1 and E_0 . The slope of E_0 before s^* is jumping from one level to the other ending in $E_2(1)$. Our intuition says that a_2 is becoming dominant before s^* as the direction of E_0 is $E_2(1)$, and crosses with a_0 . The latter case does not follow the parametrization definition 2.6. The plots of a_k 's and b_k 's for these two instances on Fig. 3.4 validate our expectations on the behavior of these curves. On the left, figures 3.4(a) and (c) show the evolution as predicted by definition 2.6, however, figures 3.4(b) and (d) show other variations of the quantities a_k and b_k for an anti-crossing. Thus, the toy model we introduced highlights the limitation of Def. 2.6. We further explained the intuition behind this



(a)



(b)

Figure 3.3: Eigenvalues evolution for toy example 1 with $\alpha = 0$ (a) and $\alpha = 0.5$ (b)

limitation and a thorough example is detailed in Sec. 3.2.3.

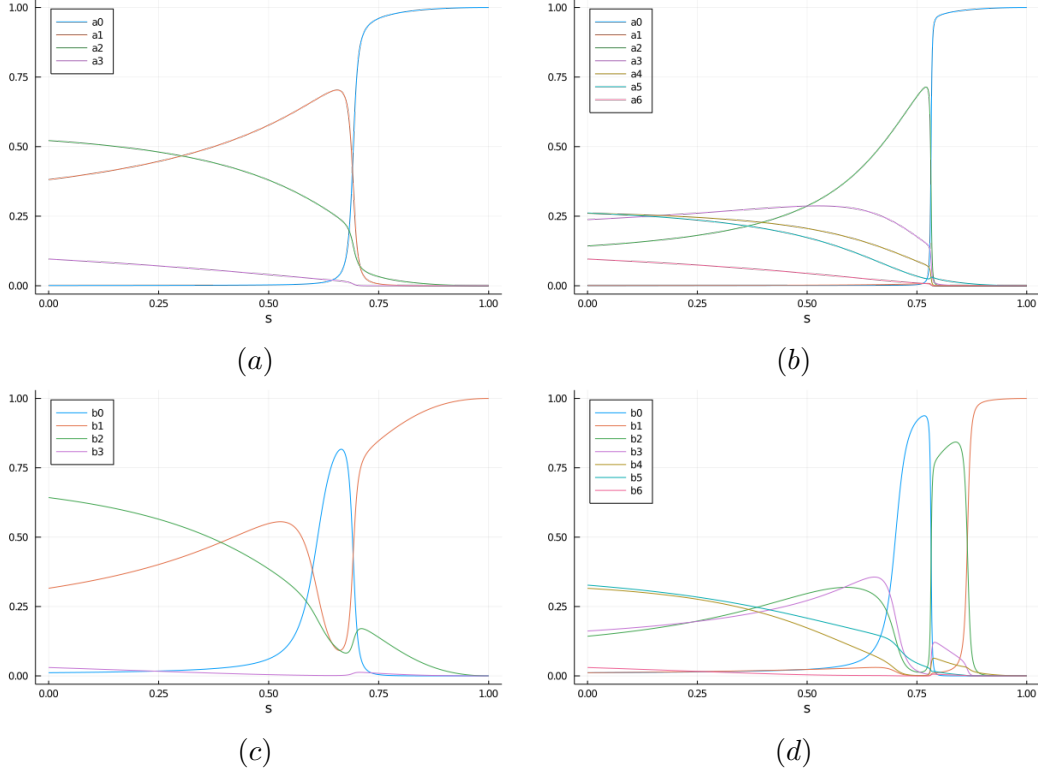


Figure 3.4: a_k (top) and b_k (bottom) during evolution for toy example 1 with $\alpha = 0$ (left) and $\alpha = 0.5$ (right). Values indexed by 0 are in blue, those by 1 in red, 2 in green and 3 in purple.

Here we gave more insights on the quantities introduced in Def. 2.6 to describe an anti-crossing and we explicit the link with the min-gap. A strong anti-crossing will end up in a small min-gap. However, Def. 2.6 is not general enough as the a_k 's and b_k 's don't explain exactly the behavior described by the definition. In the next section, still motivated by [Choi 2020], we give a new parametrization definition of an anti-crossing between the two lowest energies based only on $a_0(s)$ and $b_0(s)$ and derive a result that quantifies the min-gap.

In this section, we studied the anti-crossing of Def. 2.6 and proved that it is indeed related to a minimum gap closing but with a toy model of the maximum-weight k -clique we were able to show the limitation of this definition and explained which energy landscape could invalidate Def. 2.6. Based on our observations, in the next section we introduce a relaxed definition of an anti-crossing to encompass more cases.

3.2 New characterization of anti-crossing

In this section, we give a new general result on the derivatives of the instantaneous vectors involved in the anti-crossing and then we introduce a more general characterization of an anti-crossing than the one studied in the previous section.

3.2.1 Variation of $|\phi_0(s)\rangle$ at s^*

In AQC, we are interested in the evolution of the ground-state as it encodes the solution of the given problem at the end of the computation. By the adiabatic theorem, the state stays close to the instantaneous ground-state. Why is it hard to follow it sometimes? What is happening to the instantaneous ground-state and first excited state at crossing point? We give one element of answers in Theorem 3.1, proving that the variation of the two states that crossed are symmetrically similar. The amplitude of this variation is inversely proportional to the minimum gap, the direction of the variation is only supported by the other vector state involved in the anti-crossing.

Theorem 3.1. *Assuming that around the s^* anti-crossing point between $E_0(s)$ and $E_1(s)$, all other higher energy levels are well separated from these two levels, the following relations hold :*

$$\frac{d}{ds}|\phi_0(s^*)\rangle \simeq -\beta|\phi_1(s^*)\rangle \quad (3.4)$$

$$\frac{d}{ds}|\phi_1(s^*)\rangle \simeq \beta|\phi_0(s^*)\rangle \quad (3.5)$$

where $\beta = \frac{\langle \phi_0(s^*) | \dot{H} | \phi_1(s^*) \rangle}{\Delta_{min}}$ and the $\simeq$ symbol hides some negligible $\mathcal{O}(\delta_s^2)$ terms for a small δ_s around s^* .

Proof. Let us write the expression we get using Eq. 3.2 of Lemma 3.1, for $i = 0, 1$:

$$\frac{d}{ds}|\phi_i(s)\rangle = (-1)^{i+1} \frac{\langle \phi_0(s) | \dot{H} | \phi_1(s) \rangle}{E_1(s) - E_0(s)} |\phi_{i\oplus 1}(s)\rangle - \sum_{j \geq 2} \frac{\langle \phi_j(s) | \dot{H} | \phi_i(s) \rangle}{E_j(s) - E_i(s)} |\phi_j(s)\rangle$$

The right-hand side is composed of a first term that is β and a second term which is a sum over energy levels greater than 2. Let us show that the sum is zero. To do so, we will show that for $i = 0, 1, \forall j \geq 2, \langle \phi_i(s^*) | \dot{H} | \phi_j(s^*) \rangle = 0$. We start from the observation that the slope of $E_0(s)$ before s^* is the one from $E_1(s)$ after s^* , meaning that they satisfy $\frac{dE_0}{ds}(s^* - \delta_s) = \frac{dE_1}{ds}(s^* + \delta_s)$ for a small $\delta_s > 0$. The slopes are exchanged. This observation is consistent with the original description in [von Neumann & Wigner 1929, Wilkinson 1989].

Then we use Taylor expansion at the first order in δ_s :

$$\frac{dE_0}{ds}(s^*) - \delta_s \frac{d^2 E_0}{ds^2}(s^*) = \frac{dE_1}{ds}(s^*) + \delta_s \frac{d^2 E_1}{ds^2}(s^*) + \mathcal{O}(\delta_s^2)$$

We will neglect the $\mathcal{O}(\delta_s^2)$ terms in the rest of the proof and we hide them behind the $\simeq$ symbol. In the neighborhood of s^* , the gap $\Delta(s)$ is minimal in s^* , thus $\frac{dE_1}{ds}(s^*) - \frac{dE_0}{ds}(s^*) = \frac{d\Delta}{ds}(s^*) = 0$. We are left with :

$$\frac{d^2 E_0}{ds^2}(s^*) + \frac{d^2 E_1}{ds^2}(s^*) \simeq 0$$

We are in a setting with linear interpolation, so $\ddot{H} = 0$. We can use Eq. 3.3 of Lemma 3.1 to get:

$$2 \sum_{j \neq 0} \frac{|\langle \phi_0(s^*) | \dot{H} | \phi_j(s^*) \rangle|^2}{E_0(s^*) - E_j(s^*)} + 2 \sum_{j \neq 1} \frac{|\langle \phi_1(s^*) | \dot{H} | \phi_j(s^*) \rangle|^2}{E_1(s^*) - E_j(s^*)} \simeq 0$$

Pulling out from the first sum the term $j = 1$ and the term $j = 0$ from the second one, they cancel each other and we end up with:

$$\sum_{j \geq 2} \left[\frac{|\langle \phi_0(s^*) | \dot{H} | \phi_j(s^*) \rangle|^2}{E_j(s^*) - E_0(s^*)} + \frac{|\langle \phi_1(s^*) | \dot{H} | \phi_j(s^*) \rangle|^2}{E_j(s^*) - E_1(s^*)} \right] \simeq 0$$

The sum of positive summands is equal to zero, so each summand is equal to zero. Because the denominators are strictly positives, we obtain:

$$\forall j \geq 2, \begin{cases} \langle \phi_0(s^*) | \dot{H} | \phi_j(s^*) \rangle \simeq 0 \\ \langle \phi_1(s^*) | \dot{H} | \phi_j(s^*) \rangle \simeq 0 \end{cases}$$

This is enough to conclude. □

Actually, with some manipulations of the eigen relation like in the proof of Prop. 3.1, we have that for $i \neq j$, $\langle \phi_i(s) | \dot{H} | \phi_j(s) \rangle = \frac{1}{s} \langle \phi_i(s) | -H_0 | \phi_j(s) \rangle = \frac{1}{1-s} \langle \phi_i(s) | H_1 | \phi_j(s) \rangle \geq 0$. Thus, one can show that:

$$\forall j \geq 2, \begin{cases} \langle \phi_0(s^*) | H_1 | \phi_j(s^*) \rangle = \langle \phi_0(s^*) | H_0 | \phi_j(s^*) \rangle \simeq 0 \\ \langle \phi_1(s^*) | H_1 | \phi_j(s^*) \rangle = \langle \phi_1(s^*) | H_0 | \phi_j(s^*) \rangle \simeq 0 \end{cases}$$

Furthermore, if H_1 is positive semi-definite (which is easy to achieve by adding an identity matrix scaled by the minimum value of H_1), we have $\beta \geq 0$.

This theorem also fills a blank in Chapter 2, Sec. 2.4.2 in which we assumed the reduced matrix expression of the Hamiltonian around s^* . The matrix element $\langle \phi_0^* | H_{\delta_s} | \phi_j^* \rangle = \langle \phi_0^* | H(s^*) | \phi_j^* \rangle + \langle \phi_0^* | \dot{H} | \phi_j^* \rangle \simeq 0$ for $j \geq 2$, and the same holds with $|\phi_1^*\rangle$.

Here, we understand that the smaller the minimum gap is, the more brutal the variation of the instantaneous ground-state will be. This result can explain why it is hard to follow the trajectory of the ground-state vector. Furthermore, the direction of the variation is purely supported by the other vector involved in the anti-crossing meaning that $|\phi_0(s^*)\rangle$ and $\frac{d}{ds}|\phi_0(s^*)\rangle$ form the same plane as $|\phi_0(s^*)\rangle$ and $|\phi_1(s^*)\rangle$ where these two vectors rotate.

3.2.2 New anti-crossing definition

During adiabatic evolution, each level of energy will undergo some anti-crossings. We aim to explain the anti-crossings happening to one specific level namely the ground-state. In AQC, the purpose is to find the ground-state of the final Hamiltonian, thus knowing from which higher energy level $|\text{GS}\rangle$ comes, will help to understand if the solution has to jump many levels before ending in the ground-state. This information is present in the coefficient of $|\phi_i(s)\rangle$ corresponding to $|\text{GS}\rangle$. Therefore, instead of looking at the decomposition of the instantaneous ground-state $|\phi_0(s)\rangle$ in the computational basis, we can look at the decomposition of the final ground-state $|\text{GS}\rangle$ in the instantaneous basis by introducing :

$$g_k(s) = |\langle \text{GS} | \phi_k(s) \rangle|^2 \quad (3.6)$$

The value of the g_k during the evolution corresponds to the probability to measure the solution if the system is in the k^{th} energy level. In AQC, generally, we start from the ground-state of the initial Hamiltonian H_0 , so the state stays close to the instantaneous ground-state according to the adiabatic theorem. Therefore, we hope that g_0 is dominant at some point.

Notice that $g_0 = a_0$ and $g_1 = b_0$. The idea is to relax Def. 2.6 to include a better description of an anti-crossing between E_0 and E_1 . The intuition behind those variables are quite similar, a dominant g_j during the evolution means that the instantaneous basis vector $|\phi_j(s)\rangle$ is in direction toward the solution before s^* . We can restate the parametrization of an (γ, ε) -anti-crossing based only on g_0 and g_1 .

Definition 3.1. For $\gamma \geq 0$, $\varepsilon \geq 0$ we say there is an (γ, ε) -Anti-crossing if there exists a $\delta_s > 0$ such that

1. For $s \in [s^* - \delta_s, s^* + \delta_s]$,

$$|\text{GS}\rangle \simeq \sqrt{g_0(s)}|\phi_0(s)\rangle + \sqrt{g_1(s)}|\phi_1(s)\rangle \quad (3.7)$$

where $g_0(s) + g_1(s) \in [1 - \gamma, 1]$. Within the time interval $[s^* - \delta_s, s^* + \delta_s]$, $|\text{GS}\rangle$ is mainly composed of $|\phi_0(s)\rangle$ and $|\phi_1(s)\rangle$. That is, all other states (eigen states of the Hamiltonian $H(s)$) are negligible (which sums up to at most $\gamma \geq 0$).

2. At the avoided crossing point $s = s^*$, $g_0 = g_1 \in [1/2 - \varepsilon, 1/2]$, for a small $\varepsilon > 0$. That is, $|\text{GS}\rangle \simeq 1/\sqrt{2}(|\phi_0(s^*)\rangle + |\phi_1(s^*)\rangle)$.
3. Within the time interval $[s^* - \delta_s, s^* + \delta_s]$, $g_0(s)$ increases from $\leq \gamma$ to $\geq (1 - \gamma)$, while $g_1(s)$ decreases from $\geq (1 - \gamma)$ to $\leq \gamma$.

Our new definition is quite similar to Def. 2.6 and trivially includes all anti-crossings described by Choi's definition. We see from point 1 and 3 that we can remove point

2 by setting $2\varepsilon = \gamma$. Nevertheless, one may want to keep the independence of this two parameters to encompass more potential AC. Additionally, we have the following corollary of Theorem 3.1 that links it to the minimum gap and properly characterizes the strength of an anti-crossing.

Corollary 3.2. *Still neglecting terms in $\mathcal{O}(\delta_s^2)$,*

$$\frac{dg_0}{ds}(s^*) + \frac{dg_1}{ds}(s^*) \simeq 0 \quad (3.8)$$

$$\frac{dg_0}{ds}(s^*) - \frac{dg_1}{ds}(s^*) \simeq 4g_{0,1}(s^*)\beta \quad (3.9)$$

Proof. We have $\frac{dg_i}{ds}(s) = \langle \text{GS} | \phi_i(s) \rangle \langle \text{GS} | \frac{d\phi_i}{ds}(s) \rangle + \langle \phi_i(s) | \text{GS} \rangle \langle \frac{d\phi_i}{ds}(s) | \text{GS} \rangle$, $g_0(s^*) = g_1(s^*)$ and theorem 3.1. □

For a strong anti-crossing, $g_0(s^*) = g_1(s^*) = g_{0,1}(s^*) \simeq 1/2$. Using our definition (Def. 3.1), along with Corollary 3.2, we have that

$$4g_{0,1}(s^*)\beta = \frac{dg_0}{ds}(s^*) - \frac{dg_1}{ds}(s^*) \simeq \frac{1 - 2\gamma}{\delta_s}$$

which gives the order of the minimum gap: $\Delta_{\min} = \mathcal{O}(2\dot{H}_{01} \frac{\delta_s}{1-2\gamma})$. The main property characterizing an avoided-crossing according to our definition is the sharp variation of g_0 and $g - 1$ around s^* . We provide numerical evidence on the previous toy model (Fig. 3.1) by plotting the derivative of g_0 and g_1 against $\Delta_{\min}$ for $\alpha \in [0, 0.66]$ close to the threshold $2/3$ (Fig. 3.5). The numerical result looks consistent with Corollary 3.2. We clearly see that the derivative of g_0 and g_1 have similar opposite variation. The plain lines express the tendency of the variations which are clearly inversely proportional to the minimum gap.

In this section, we introduced a new characterization of an anti-crossing which is a relaxed version of Def. 2.6. We supported this definition by an analytical result and numerical evidence on AC that the previous definition (Def. 2.6) does not capture. In the next section, we further detail an example and explain each expected behavior of the a_k , b_k and g_k .

3.2.3 Thorough comparison of ACs definitions

In this section, we illustrate how the two definitions (Def. 2.6 and Def. 3.1) differ and interpret this difference through the previous intuitions. We use a new toy model for the maximum-weight k -clique problem with the graph in Fig. 3.6. It is the same structure of the graph Fig. 3.1 where nodes 1 and 3 are swapped as well as nodes 5 and 6. We keep the same weights vector $w = [1, 1, 1, 1.5, 1.5, 1.5]$.

For $\alpha = 0.2$, this produces the following final states according to their energy (Fig. 3.7 (a)) and the eigenvalues evolution (Fig. 3.7 (b)).

With this instance, we observe that the final slope of $E_0(s)$ comes from the slope of $E_2(s)$, therefore we expect that $g_2(s)$ becomes dominant before transmitting to

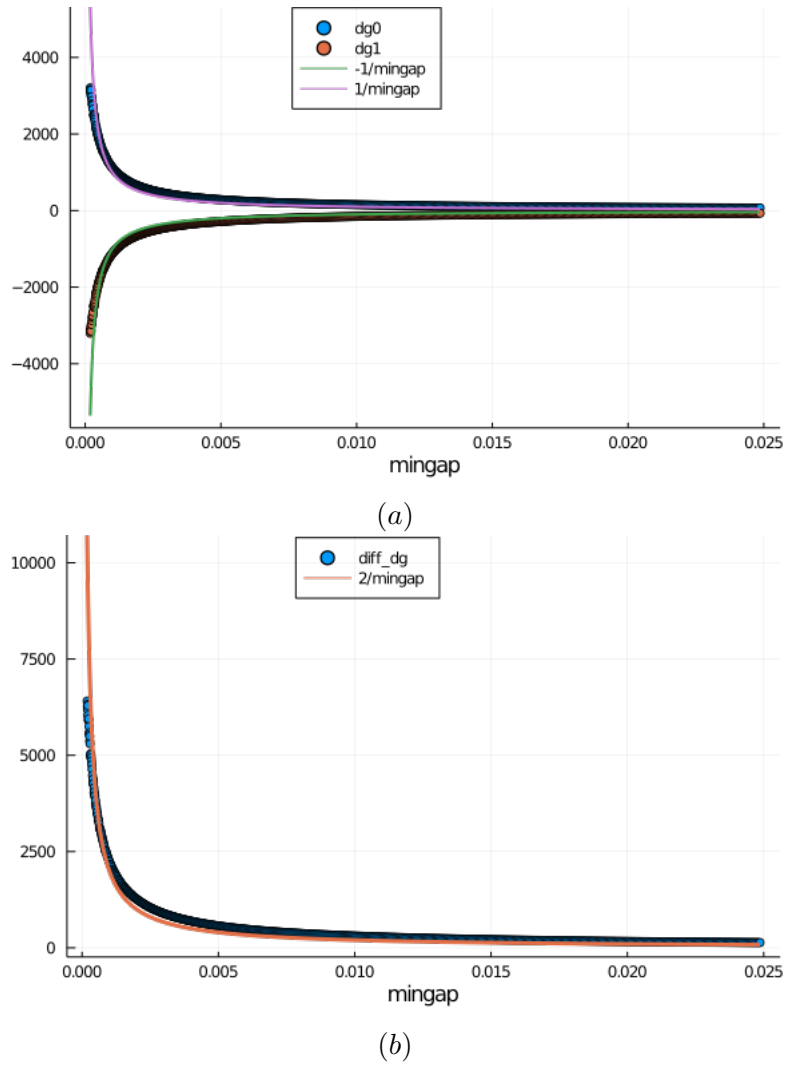


Figure 3.5: Evolution of the derivative of g_0 and g_1 (a) and the difference (b) against the min-gap for α varying from 0 (large gap) to 0.66 close to the threshold $2/3$ (small gap).

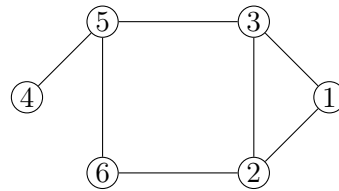


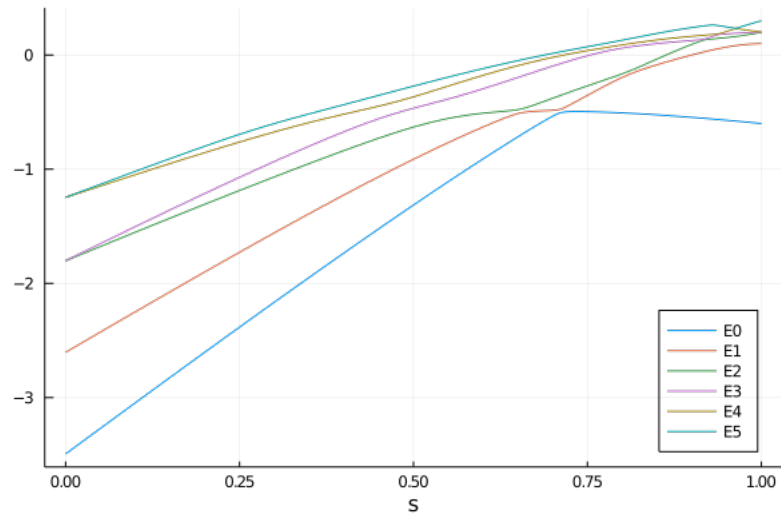
Figure 3.6: Toy example 2

```

-0.6 : ["111000"]
0.1  : ["000111"]
0.2  : ["001011", "001110", "010011"]
0.3  : ["011001", "011010", "101010", "110001"]
1.2  : ["010101", "010110", "100011", "100110"]
1.3  : ["011100", "101001", "101100", "110010", "110100"]
2.2  : ["001101", "100101"]

```

(a)



(b)

Figure 3.7: States (a) with their energy and $E_i(s)$ (b) during evolution for toy example 3.6 with $\alpha = 0.2$

$g_1(s)$ after the first anti-crossing between E_2 and E_1 . Eventually, the anti-crossing of E_1 and E_0 will produce a crossing of g_1 and g_0 . The latter will become dominant and will become equal to 1 at the end. Now, focusing on the slope of $E_0(s)$ before the anti-crossing, following the different successive anti-crossings, the jumps end up in the 4th energy level. In terms of a_k 's and b_k 's, this means that $a_3(s)$ becomes important just before the anti-crossing and crosses $a_0(s)$ at the anti-crossing. The same goes for $b_3(s)$ and $b_0(s)$. The plots below supports these previous analyses.

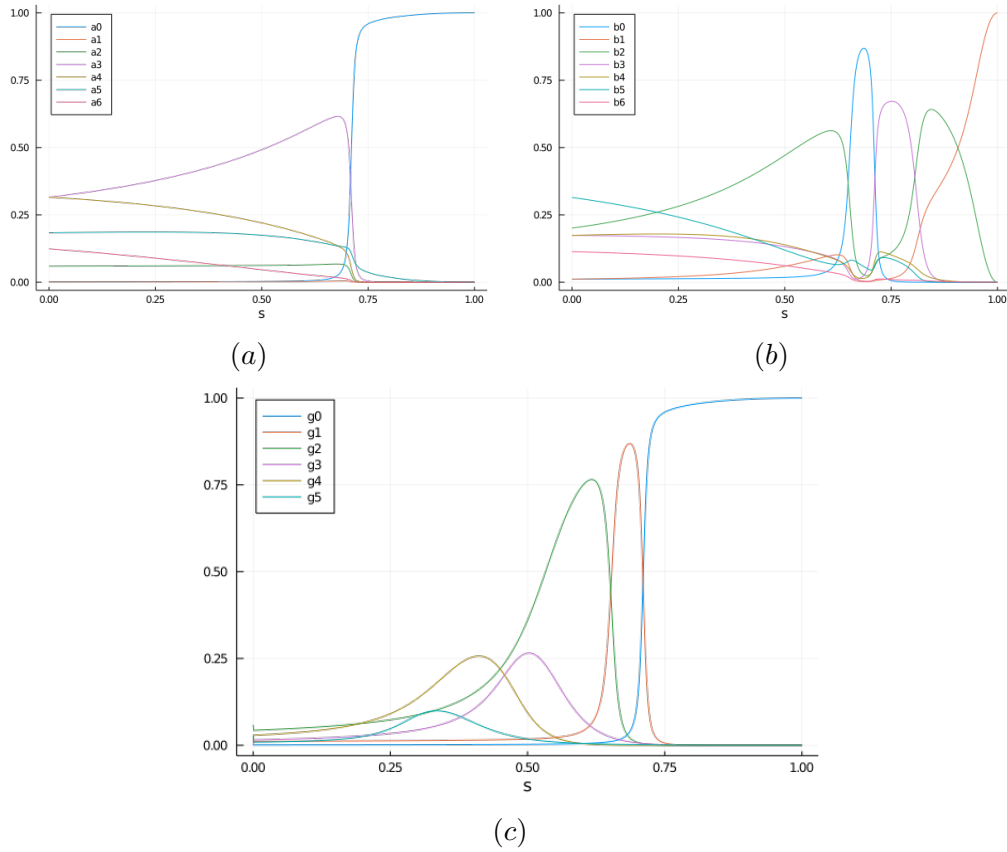


Figure 3.8: a_k (a), b_k (b) and g_k (c) during evolution for toy example 3.6 with $\alpha = 0.2$. Values indexed by 0 are in blue, those by 1 in red, 2 in green and 3 in purple.

Remarks:

1. On Fig. 3.8 (a), we only see one specific situation, namely the crossing of a_3 with a_0 at the anti-crossing point s^* between E_0 and E_1 . This means that the curve $E_0(s)$ was going toward the 4th energy level in terms of the slope before s^* and immediately change its slope toward its final direction of the 1st energy level. Hypothetically, we could observe a_3 crossing a_1 , which will indicate that $E_0(s)$ change its direction toward the 2nd energy level, then a_1 crossing a_0 . In this hypothetical case, $E_0(s)$ undergoes two anti-crossings.

2. On Fig. 3.8 (b), we focus on the behavior of $|\phi_1(s)\rangle$. Here, we understand that $E_1(s)$ first went toward the 3rd energy level before changing direction toward the lowest energy level. This is b_2 crossing b_0 when the first anti-crossing between E_1 and E_2 occurs. Then, it takes the direction of the 4th energy level when b_0 crosses b_3 . Indeed, it fetches the direction of E_0 before anti-crossing (remember it was a_3 which was dominant at this point). Then again takes back its initial direction with b_2 becoming dominant at the second anti-crossing between E_1 and E_2 . Eventually, smoothly change its direction toward its final goal.
3. On Fig. 3.8 (c), the point of view is quite different as we look from the final lowest energy position $E_0(1)$ and see from where it comes. We see in Fig. 3.7 (b) that the final blue slope undergoes two anti-crossing before becoming blue. Indeed, it starts green, then jumps to red and finally blue. These successive anti-crossings appear on the plot of g_k , first, g_2 is dominant, then at the first anti-crossing between E_2 and E_1 , g_2 crosses with g_1 . Now, the final slope of E_0 is transported by E_1 . Eventually, E_1 anti-crosses E_0 so g_1 crosses g_0 and the evolution (at least for the ground-state) can finish peacefully.

From this detailed example, we understand that avoided crossings are a place where slopes of eigen energies are swapping. As hinted in the first bullet point of the remarks, the new introduced definition has its own limitation as we see in the next section.

3.3 Limitation and other potential type of AC

In this section, we highlight potential limitations in our AC definition. In Definition 3.1, emphasis is placed on the significance of the $|\text{GS}\rangle$ coefficient in the instantaneous eigenvectors. As defined, $|\phi_0(s)\rangle$ will ultimately converge to $|\text{GS}\rangle$, and if an AC occurs between E_1 and E_0 , then g_0 will inevitably undergo a noticeable increase. However, this observation holds true only for the final AC in the evolution. Notably, if E_0 experiences multiple AC events (as in [Somma *et al.* 2012, Feinstein *et al.* 2022]), the initial AC may not conform to the behaviors outlined in our definition.

For instance, considering an energy spectrum akin to Fig. 3.9, our definition remains applicable for the second AC of $E_0(s)$ but not for the first one. In this scenario, g_2 begins to dominate before the first anti-crossing between E_2 and E_1 . Subsequently, g_2 intersects with g_1 , which then becomes the dominant coefficient. The crossing between g_1 and g_0 occurs at the second anti-crossing between E_1 and E_0 , aligning with the predictions of our definition.

Now the question is “How can we describe the first AC”. We have two perspectives. Either we take the point of view of looking where E_0 is going. Here, following the different AC brings the initial slope of E_0 to E_3 , so a_3 is becoming dominant, then after the crossing point, E_0 is heading toward E_2 so a_2 is becoming dominant.

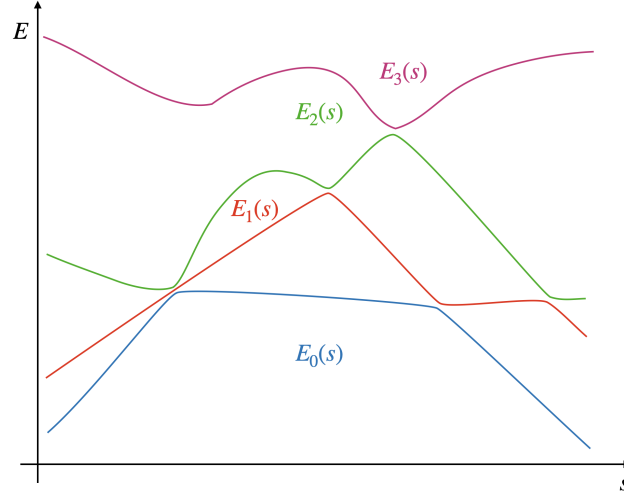


Figure 3.9: Example of an energy spectrum in which the ground-state undergoes two anti-crossing. In this situation, the first AC does not follow our definition but the second one does.

We can expect a crossing of a_3 and a_2 and inversely between b_2 and b_3 as well. Or, similarly to our definition, we also expect a_2 and b_2 to cross. We see with this example that it might be more complicated to find a general approach to describe an AC. Another potential AC is one where E_2 anti-crosses with E_0 leaving E_1 in the middle without changing its trajectory.

Conclusion

In this chapter, we questioned the definition of an AC given by [Choi 2020] and shown a limitation case. Supported by a theoretical approach, we suggested a relaxed definition to encompass more cases. We provided a complete example along with a discussion based on our intuition on the evolution of the different quantities a_k , b_k and g_k . Eventually we suggested further directions to describe even more potential AC. The main ingredient of all these observations are the conservation of the slope at an AC. Is it always verified? We leave this question for future work. Now that we have a better understanding of the phenomenon that can generate exponentially small minimum gap during a quantum evolution, we move on the presence of such closure during a quantum evolution. In the next chapter, we prove a necessary condition on the occurrence of exponentially closing gaps in AQC. As it happens, these gaps do not correspond to the definition we have introduced in this chapter, and we will study their impact on the efficiency of AQC.

Exponentially closing gaps in AQC

Contents

4.1 Perturbative Analysis to AQC	85
4.1.1 Initial perturbation	89
4.1.2 Final perturbation	90
4.1.3 Energy crossing	91
4.2 Application to MaxCut over bipartite graphs	93
4.2.1 d -regular bipartite graphs	93
4.2.2 General bipartite graphs	99
4.2.3 Numerical study: AC and other observations	103
4.2.4 Validation of perturbative expansion and limitations	106
4.3 Higher order perturbative expansion for MaxCut	108

In this chapter, we develop a perturbative first-order expansion at the beginning and end of the adiabatic process to prove a necessary condition for the appearance of first-order quantum phase transitions. In this chapter, we refer to AC as first-order quantum phase transitions that are characterized by the presence of an exponentially closing gap. We apply this result to the MaxCut problem on bipartite graphs. We show that AQC efficiently solves the problem when the graph is also regular. Furthermore, we construct a family of non-regular bipartite graphs that satisfies the AC occurrence condition, and numerically confirm the appearance of an exponentially decreasing gap. This last interesting case leads us to question the correlation between the appearance of an exponentially small gap and the difficulty of solving the problem since, despite the very small gap, AQC efficiently finds the optimal solution with a fairly good probability (Sec. 4.2). Not all exponentially small deviations are equivalent in terms of the efficiency of AQC. Using the definition of AC we proposed in chapter 3 (Def. 3.1), we are able to distinguish the latter case with a first-order quantum phase transition that is a source of failure for AQC. Finally, we suggest a more advanced perturbative expansion for the MaxCut problem (Sec. 4.3).

4.1 Perturbative Analysis to AQC

In this section, we apply the perturbative analysis presented in Sec. 2.4.1 to the AQC process. This idea has already been explored by other authors [Altshuler *et al.* 2010,

[Amin & Choi 2009, Werner *et al.* 2023] to derive different results and intuitions about the occurrences of an anti-crossings during the evolution. The perturbative analysis can be naturally applied at the beginning ($s(0) = 0$) and at the end ($s(T) = 1$) of the evolution. Typically, we know the eigen basis of H_0 and H_1 which allows us to deduce relevant features about the process. Building on the work of [Werner *et al.* 2023], we develop here an expansion of the energy $E_0^I = E_0(0)$ of the initial state $|\psi_0\rangle = |\phi_0(0)\rangle$, i.e. the ground-state of H_0 and for the energies $E_{gs} = E_0(1)$ of the final ground-state $|\text{GS}\rangle = |\phi_0(1)\rangle$ and $E_{fs} = E_1(1)$ of the final first excited state $|\text{FS}\rangle = |\phi_1(1)\rangle$ of H_1 . We further assume that the first excited subspace of H_1 is degenerate, meaning that there are many solutions with second best cost value. We already know from [Amin & Choi 2009] that the second-order perturbation can be enough to prove the presence of an AC. In Sec. 2.4.3, we also saw that at first-order, without the degeneracy condition, it was impossible to conclude on the crossing of the perturbed final energy. In [Werner *et al.* 2023], the authors show that adding the condition of a first degenerate excited subspace of H_1 lifts the degeneracy at first order. We place ourselves in this setting to derive powerful technical results on the occurrence of ACs. Recall that AC refers to the point where the gap is closing exponentially fast, i.e. when the two lowest instantaneous eigen-energies are getting exponentially close to each other. Intuitively, the energy curves almost cross but change directions right before (Sec. 2.4 and 3.2).

Before deriving the different expansions (in the next sections), let us set again the considered time-dependent Hamiltonian $H(s) = (1-s)H_0 + sH_1$. We need to define H_0 , H_1 and the trajectory $s(t)$. We choose to stay in the standard setting of AQC for solving classical optimization problems defined over the bit-strings of size n where $s(t)$ can be any function from 1 to 0 (e.g. $s(t) = \frac{t}{T}$), $H_0 = -\sum_i \sigma_x^{(i)}$ where the sum is over the n qubits of the considered quantum system and $H_1 = \sum_{x \in \{0,1\}^n} E_x |x\rangle\langle x|$. E_x is the value for a classical n -bit-string x of the function we want to optimize, i.e. if C is a cost function to minimize, $C(x) = E_x$. We refer the reader to Sec. 2.2 for more details on the construction of the Hamiltonians. From this setting, we know that $|\psi_0\rangle$ is the uniform superposition over all bit-strings and the associated eigen-space is non-degenerate. We further assume that the ground space of H_1 is non-degenerate as well, i.e. $\exists! x^*, E_{x^*} = E_{gs}$ while the first excited subspace is degenerate, i.e. $\exists x \neq y, E_x = E_y = E_{fs} > E_{gs}$.

We now introduce different graphs that help us to better visualize some quantities. As described in Sec. 2.2.2, H_0 can be seen as the negative adjacency matrix of an n -regular graph. If each node represents a bit-string x , this state is connected to another one y via H_0 if y is exactly one bit-flip (σ_x operation) away from x . We express this relation as $x \underset{H_0}{\sim} y$. For any bit-string of size n , there are exactly n possible bit-flips. $-H_0$ represents the search graph which is the hypercube in dimension n among all possible solutions x . We can isolate the nodes that belong to the degenerate first excited subspace of energy E_{fs} among all x , i.e. $V_{\text{loc}} = \{y \in \{0,1\}^n | E_y = E_{fs}\}$ and we can define the graph induced by those states V_{loc} in $-H_0$. We call G_{loc} this subgraph that corresponds to the local minima of

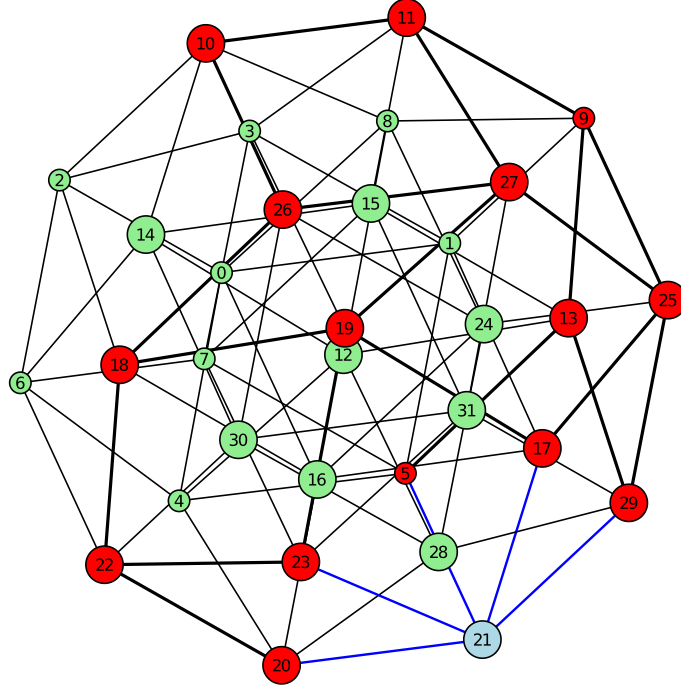


Figure 4.1: A 5-cube with G_{loc} highlighted with red nodes and thick black edges. Light-blue node is the unique ground-state and blue edges show the connection between G_{loc} and the ground-state. Green nodes are all the other possible states with higher energies. The labels, once converted in binary, represent the state configuration.

the optimization problem. An example of G_{loc} in the 5-cube is shown on Fig. 4.1. We use MaxCut on a cycle to generate this example, we give the details in the next section. To visualize the landscape of such a graph, we draw in Fig. 4.2 a schematic two-dimensional plot of the objective function $C(x)$ which is also the energy landscape of H_1 . In the example of Fig. 4.1, we see that the optimal state $x^* = |\text{GS}\rangle$ is only connected to states in G_{loc} and there is no component of G_{loc} far from it, i.e. with a potential barrier in between. This idea is conveyed in Fig 4.2 by the absence of green parts between the red and blue sections.

In the rest of the section, we detail the perturbation expansions and how we can articulate them to derive a condition on the occurrence of the anti-crossing during AQC. More precisely, we will prove the following theorem:

Theorem 4.1. *Under perturbative expansion validity, if $\lambda_0(loc)$ is the unique largest eigenvalue of the adjacency matrix of G_{loc} and H_1 has a unique ground-state and a degenerate first eigen-space, with a linear interpolation in s between H_0 and H_1 as*

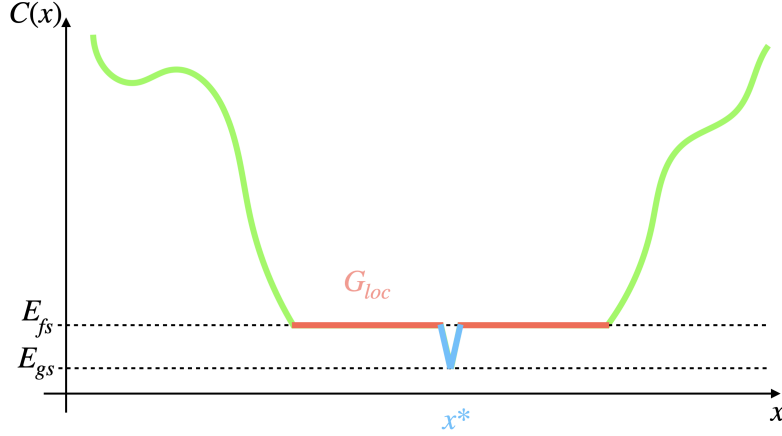


Figure 4.2: Schematic energy landscape of H_1 corresponding to Fig. 4.1. G_{loc} has only one component and is strongly connected to the optimal state x^* .

defined above, and by defining

$$s_{lg} = \frac{\lambda_0(loc)}{\Delta H_1 + \lambda_0(loc)} = \frac{1}{1 + \frac{\Delta H_1}{\lambda_0(loc)}}$$

and

$$\alpha_{H_1} = \frac{\Delta H_1}{\langle H_1 \rangle_0 - E_{gs}}$$

where $\Delta H_1 = E_{fs} - E_{gs}$ and $\langle H_1 \rangle_0$ is the mean of H_1 's eigenvalues, we can say that an anti-crossing happens at s_{lg} if $\lambda_0(loc) > n\alpha_{H_1}$. No anti-crossing occurs if $\lambda_0(loc) < n\alpha_{H_1}$.

The assumption on $\lambda_0(loc)$ is always satisfied when G_{loc} has a unique major component. This forms a general condition on the occurrence of an anti-crossing during a quantum process with the assumptions of the theorem. We see that the α_{H_1} parameter depends only on the problem H_1 while $\lambda_0(loc)$ is mixing H_0 and H_1 . We observe from this result that the potential occurrence time of an AC around s_{lg} is ruled by the ratio $\frac{\Delta H_1}{\lambda_0(loc)}$. In practice, this result can help computer scientists to design appropriate schedules by slowing the evolution around the AC. However, the $\lambda_0(loc)$ parameter can be complicated to compute. It encodes the centrality of G_{loc} and can be interpreted as the importance of this graph. To tackle this we use a result from graph theory [Zhang 2021] that bounds the largest eigenvector of a graph by : $\mathbf{d}_{avg}(loc) \leq \lambda_0(loc) \leq \mathbf{d}_{max}(loc)$. Where $\mathbf{d}_{avg}(loc)$ and $\mathbf{d}_{max}(loc)$ denote the average and maximum degree of G_{loc} respectively. We can derive the following more practical corollary:

Corollary 4.1. *By introducing,*

$$s_{lg}^+ = \frac{\mathbf{d}_{avg}(loc)}{\Delta H_1 + \mathbf{d}_{avg}(loc)}$$

$$s_{lg}^- = \frac{\mathbf{d}_{max}(loc)}{\Delta H_1 + \mathbf{d}_{max}(loc)}$$

we can distinguish three regimes :

- *AC occurs in the interval $[s_{lg}^+, s_{lg}^-]$ if $\mathbf{d}_{avg}(loc) > n\alpha_{H_1}$;*
- *No-AC occurs if $\mathbf{d}_{max}(loc) < n\alpha_{H_1}$;*
- *Undefined if $\mathbf{d}_{max}(loc) > n\alpha_{H_1} > \mathbf{d}_{avg}(loc)$.*

This corollary gives an interval where an AC may occur. Furthermore, it will help anyone who wants to study the different regimes when applied to a specific problem as we do with MaxCut in the next section. In any case, this analytical result is derived from the perturbative theory and the validity of the truncation used needs to be checked. We suggest a validation of this approach applied to MaxCut over bipartite graphs, through analytical and numerical evidence in Sec. 4.2.4. Now let us detail the proof of the theorem.

4.1.1 Initial perturbation

At the beginning of the evolution, we know that we start from the ground-state of H_0 with energy E_0^I , i.e. $H_0|\psi_0\rangle = E_0^I|\psi_0\rangle$. We are interested in how it changes while perturbing H_0 with some H_1 . More formally, let us look at the modified Hamiltonian $\tilde{H}(\varepsilon) = H_0 + \varepsilon H_1$ which is obtained by dividing the original Hamiltonian by $(1-s)$ and setting $\varepsilon = \frac{s}{1-s}$. If we call $\tilde{E}_{deloc}(\varepsilon)$, “deloc” for delocalized state, the ground-state energy expansion at first-order of $\tilde{H}(\varepsilon)$, by perturbative analysis with non-degenerate subspace, the first-order expansion is :

$$\begin{aligned} \tilde{E}_{deloc}(\varepsilon) &= E_0^{(0)} + \varepsilon E_0^{(1)} \\ &= \langle \psi_0 | H_0 | \psi_0 \rangle + \varepsilon \langle \psi_0 | H_1 | \psi_0 \rangle \\ &= E_0^I + \varepsilon \langle H_1 \rangle_0 \end{aligned}$$

where $E_0^I = -n$ and the associated state $|\psi_0\rangle$ is a uniform superposition among all bit-strings. Hence, $\langle H_1 \rangle_0$ represents the mean of all possible values of the optimization problem, encoded in H_1 . Therefore, in the s frame, we end up with :

$$E_{deloc}(s) = -(1-s)n + s\langle H_1 \rangle_0 \quad (4.1)$$

This change of frame is valid as one could have studied directly $H(s) = H(s_\times) + (s - s_\times)\dot{H}$ for $s_\times = 0$ as we do when looking at the AC point $s_\times = s^*$ in Sec. 2.4.2.

4.1.2 Final perturbation

At the end of the evolution, we know that the ideal case is one where the state largely overlaps with the final ground-state. However, the appearance of an anti-crossing can lead to a significant overlap with the first excited state, or even with higher levels (see Sec. 3.3 for ACs cascade). We are therefore interested in the behavior of the energies ending in E_{gs} and E_{fs} when perturbed by H_0 . More formally, consider the modified Hamiltonian $\bar{H}(\lambda) = H_1 + \lambda H_0$ which is obtained by dividing the original Hamiltonian by s and setting $\lambda = \frac{1-s}{s}$. We choose this point of view because it is more usual when applying perturbation theory, but as the initial case, setting $s_\times = 1$ leads to the same expressions.

First, we focus on the behavior of the ground-state. We know that $H_1|\text{GS}\rangle = E_{gs}|\text{GS}\rangle$. If we call $\bar{E}_{\text{glob}}(\lambda)$, where “glob” stands for global minima, the ground-state energy expansion at first-order of $\bar{H}(\lambda)$, by perturbative analysis with a non-degenerate subspace, the first-order expansion is :

$$\begin{aligned}\bar{E}_{\text{glob}}(\lambda) &= E_{gs}^{(0)} + \lambda E_{gs}^{(1)} \\ &= \langle \text{GS} | H_1 | \text{GS} \rangle + \lambda \langle \text{GS} | H_0 | \text{GS} \rangle \\ &= E_{gs}\end{aligned}$$

Recall that E_{gs} is the optimal value of the optimization problem we are studying and the associated eigenspace is non-degenerate. Thus $|\text{GS}\rangle$ is a quantum state that encodes a classical bit-string, the optimal solution to the problem. In other words, $|\text{GS}\rangle$ is a vector of the canonical basis of the Hilbert space and $\langle \text{GS} | H_0 | \text{GS} \rangle$ is a diagonal element of H_0 which are all zero. Therefore in the s frame, we obtain :

$$E_{\text{glob}}(s) = sE_{gs} \quad (4.2)$$

Secondly, we focus on the evolution of the first excited state. However, we supposed that this subspace is degenerate so we need to be more precise about which state we want to study. Let $|\text{FS}, k\rangle$ be the k^{th} eigenstate of the degenerate eigenspace of H_1 , by definition $H_1|\text{FS}, k\rangle = E_{fs}|\text{FS}, k\rangle$. If we keep the usual bit-string basis among the degenerate subspace, the first order term $\langle \text{FS}, k | H_0 | \text{FS}, k \rangle$ will always be zero and the degeneracy is not lifted. The states $|\text{FS}, k\rangle$ can be ordered by continuity of the non-degenerate instantaneous energy landscape of $H(s)$ and thus $\bar{H}(\lambda)$ as well. We therefore focus on the energy evolution of the state $|\text{FS}, 0\rangle$. If we call $\bar{E}_{\text{loc}}(\lambda)$ the first excited state energy expansion at first-order of $\bar{H}(\lambda)$, by perturbative analysis with non-degenerate subspace, the first-order expansion is :

$$\begin{aligned}\bar{E}_{\text{loc}}(\lambda) &= E_{fs,0}^{(0)} + \lambda E_{fs,0}^{(1)} \\ &= \langle \text{FS}, 0 | H_1 | \text{FS}, 0 \rangle + \lambda \langle \text{FS}, 0 | H_0 | \text{FS}, 0 \rangle \\ &= E_{fs} + \lambda \langle \text{FS}, 0 | H_0 | \text{FS}, 0 \rangle\end{aligned}$$

To lift the degeneracy at first-order, we need to find a “good” basis $|\text{FS}, k\rangle$ for which $\forall k \geq 1, \langle \text{FS}, 0 | H_0 | \text{FS}, 0 \rangle < \langle \text{FS}, k | H_0 | \text{FS}, k \rangle$. We take as basis vectors $|\text{FS}, k\rangle$ of the degenerate eigenspace the eigenvectors of adjacency matrix A_{loc} of G_{loc} . With this notation, $A_{loc}|\text{FS}, k\rangle = \lambda_k|\text{FS}, k\rangle$ where we order $\lambda_0 > \lambda_1 \geq \lambda_2 \geq \dots$ and finally $\langle \text{FS}, k | H_0 | \text{FS}, k \rangle = -\lambda_k$ by construction. This ensures that the degeneracy is lifted if the largest eigenvalue of A_{loc} is unique. This happens if G_{loc} has a unique major component, which we assume. Note that if G_{loc} consists only of isolated nodes, intuitively they become as difficult as the ground-state to find by AQC unless there are exponentially many of them [Ebadi *et al.* 2022], we assume from now that this is not the case. Hence, λ_0 is unique and in the s frame, we end up with :

$$E_{loc}(s) = sE_{fs} - (1-s)\lambda_0 \quad (4.3)$$

From [Zhang 2021], we can bound the largest eigenvector of a graph by : $\mathbf{d}_{\text{avg}}(\text{loc}) \leq \lambda_0 \leq \mathbf{d}_{\text{max}}(\text{loc})$, where $\mathbf{d}_{\text{avg}}(\text{loc})$ and $\mathbf{d}_{\text{max}}(\text{loc})$ denote the average and maximum degree of G_{loc} respectively. Consequently, we can use the following more practical bounds on $E_{loc}(s)$:

$$E_{loc}(s) \geq sE_{fs} - (1-s)\mathbf{d}_{\text{max}}(\text{loc}) = E_{loc}^-(s) \quad (4.4)$$

$$E_{loc}(s) \leq sE_{fs} - (1-s)\mathbf{d}_{\text{avg}}(\text{loc}) = E_{loc}^+(s) \quad (4.5)$$

4.1.3 Energy crossing

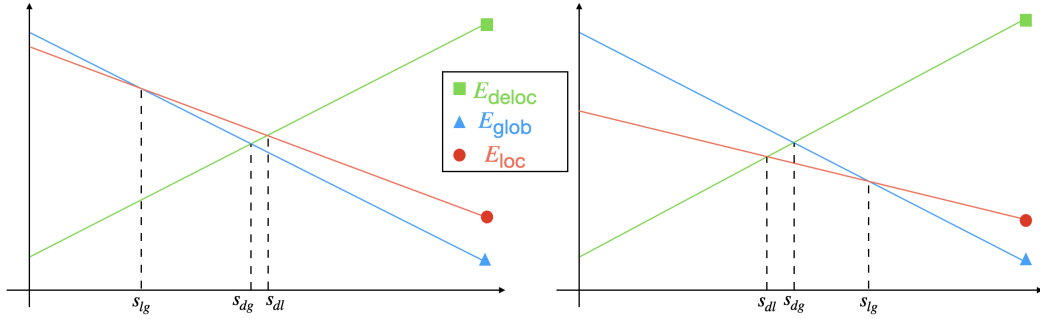


Figure 4.3: Schematic behavior of the three energy expansions: (left) a case where the condition of AC is not satisfied and (right) a case where the condition is satisfied.

We are in position to distinguish different regimes during which an avoided-crossing does or does not occur. The state begins in the delocalized situation, since $|\psi_0\rangle$ is the uniform superposition, with energy E_{deloc} . If it first crosses E_{glob} , it then follows trajectory of the instantaneous ground-state to “easily” reach the final optimal state. If it crosses E_{loc} first, then it follows the trajectory of the first instantaneous excited state, and at some point it will cross E_{glob} later and an anti-crossing will occur at that instant. Hence, the two instants of interest in the dynamics are s_{dg} , defined such that $E_{\text{deloc}}(s_{dg}) = E_{\text{glob}}(s_{dg})$, and s_{dl} , defined such that $E_{\text{deloc}}(s_{dl}) = E_{\text{loc}}(s_{dl})$. If $s_{dl} < s_{dg}$, then an anti-crossing occurs at time s_{lg}

verifying that $E_{\text{loc}}(s_{lg}) = E_{\text{glob}}(s_{lg})$. Fig. 4.3 shows the possible behavior of the energy expansions. In this plot, we assume that $E_{gs} < E_{fs} < 0$. The slope of the curve E_{loc} also depends on λ_0 , the largest eigenvalue of G_{loc} . A larger λ_0 value shifts the sign of the slope toward positives values, so that E_{loc} crosses E_{deloc} before E_{glob} , all else being equal. This situation (right) will create an AC during the annealing. It is important to note that a large value of λ_0 means high connectivity in the graph G_{loc} (or at least in its largest component). In other words, this means that the local minima are large in the mixing graph H_0 making it difficult for AQC to converge toward the global minimum.

We can derive the explicit expression for s_{dg} , s_{dl} and s_{lg} as follows:

$$\begin{aligned} s_{dg} &= \frac{n}{n + \langle H_1 \rangle_0 - E_{gs}} \\ s_{dl} &= \frac{n - \lambda_0}{n - \lambda_0 + \langle H_1 \rangle_0 - E_{fs}} \\ s_{lg} &= \frac{\lambda_0}{\Delta H_1 + \lambda_0} = \frac{1}{1 + \frac{\Delta H_1}{\lambda_0}} \end{aligned}$$

We note

$$\alpha_{H_1} = \frac{\Delta H_1}{\langle H_1 \rangle_0 - E_{gs}}$$

where $\Delta H_1 = E_{fs} - E_{gs}$, a parameter that depends only on the problem H_1 we want to solve. And so AC occurs at s_{lg} if $s_{dl} < s_{dg}$ i.e. if $\lambda_0 > n\alpha_{H_1}$. This concludes the proof of our theorem.

The corollary immediately follows by using $E_{\text{loc}}^-(s)$ and $E_{\text{loc}}^+(s)$. The undefined regime is then when $s_{dg} \in [s_{dl}^-, s_{dl}^+]$ because we cannot discriminate which curve the delocalized energy will cross first.

□

This result is quite general for many targets Hamiltonians, but we still need several conditions: the ground-state must be unique and the first excited subspace is degenerate, the validity of the first-order perturbative expansion and the unique largest component in G_{loc} .

Remark. From a physical point of view, as developed in Sec. 2.4.3 and more in [Amin & Choi 2009], we understand that in the AC-case the instantaneous ground-state first localized in the first excited state $|\text{FS}\rangle$ (or in the eigen subspace) corresponding to a quantum phase transition of second order generating only a polynomially small gap. The AC happens when the state then experiences a transition from a localized state $|\text{FS}\rangle$ to another localized state $|\text{GS}\rangle$ related to a first-order quantum phase transition.

In this section, we applied the perturbative analysis to AQC and showed under assumptions about H_1 , G_{loc} and the expansion truncation, that anti-crossings may occur during annealing given a condition to be satisfied that depends on G_{loc} and

H_1 . We have also given a corollary that relaxes the assumptions of the theorem so that it is more useful when applied to a specific problem. In the next section, we show such an application in the case of MaxCut on bipartite graphs.

4.2 Application to MaxCut over bipartite graphs

In this section, we apply Corollary 4.1 to the MaxCut problem over bipartite graphs. Given a graph $G(V, E)$, recall that the goal of MaxCut is to partition its node set V into two parts L and R in order to maximize the number of cut edges, i.e., of edges with an endpoint in L and the other in R . Such partitions are classically encoded by a bit-string of size $n = |V|$, the i^{th} bit being set to 0 if node $i \in L$, and to 1 if $i \in R$. We define our target Hamiltonian as $H_1 = -\sum_{\langle ij \rangle \in E} \frac{1 - \sigma_z^{(i)} \sigma_z^{(j)}}{2}$. This Hamiltonian (and the corresponding MaxCut cost function, see Sec. 2.2.2) has a trivial symmetry : any solution can be turned into a solution with the same energy by bit-flipping all its entries. Consequently, H_1 has a degenerate ground-state. We can break down this symmetry by forcing an arbitrary bit (say the first one) to 0 and updating H_1 accordingly. In Sec. 4.2.4, we provide some analytical tools to justify the validation of the perturbation expansion in such cases.

To ensure that the two conditions on H_1 are met, we need to choose a class of graphs such that the ground-state is non-degenerate (after breaking the trivial symmetry). Connected bipartite graphs obviously respect this property and we focus on them in the rest of the section. We will in particular show that the first excited subspace is degenerate. Also this class allows us to explicitly determine the parameter α_{H_1} and the graph G_{loc} . This will help us to determine the existence (or not) of ACs while solving MaxCut on these graphs with AQC.

4.2.1 d -regular bipartite graphs

We first restrict the bipartite graphs on being d -regular and we will show that no AC appears during the evolution by using the result from Corollary 4.1 : $\mathbf{d}_{\max}(\text{loc}) < n\alpha_{H_1}$. Leading to the following theorem :

Theorem 4.2 (No AC - d -regular bipartite graphs). *AQC efficiently solves MaxCut on d -regular bipartite graphs, assuming the validity of the first-order expansion in this case.*

First, we show the two following claims to give a value to $n\alpha_{H_1}$, then we show the No-AC conditions with Lemma 4.1 if $d \notin \{2, 4\}$. The latter two cases are tackled separately where we directly use the technical Theorem 4.1 to prove the desired result.

Claim 1. For d -regular bipartite graphs we have, $n\alpha_{H_1} = \frac{4l}{d}$, where $l \in [1, d]$ denotes the number of uncut edges in the first excited state, i.e. $E_{fs} = E_{gs} + l$.

Proof of claim 1. For bipartite graphs we have that $\langle H_1 \rangle_0 = -\frac{|E|}{2}$, $E_{gs} = -|E|$ and $\Delta H_1 = l \in [1, d]$. For regular graphs, we also have that $|E| = \frac{dn}{2}$. This is enough to prove Claim 1. $\square$

So $n\alpha_{H_1} = \frac{4l}{d}$ and we need to look at how $\mathbf{d}_{\max}(\text{loc})$ and $\mathbf{d}_{\text{avg}}(\text{loc})$ behave compared to $\frac{4l}{d}$.

Claim 2. There exist graphs with $\mathbf{d}_{\max}(\text{loc}) > 0$ only if $l = d$. Therefore $n\alpha_{H_1} = 4$.

Proof of claim 2. Recall that G_{loc} is the subgraph induced by solutions of energy E_{fs} in the hypercube $-H_0$. In other words, the vertices of G_{loc} are configurations (bit-strings) of energy E_{fs} (so “second best” solutions for MaxCut), and two vertices are adjacent if the corresponding bit-strings differ in exactly one bit, i.e., each one is obtained by bit-flipping a single bit of the other. We denote by $\mathbf{d}_{\max}(\text{loc})$ the maximum degree of G_{loc} . We know that, in the input graph G , there exists a left/right partition of its vertices such that all edges lie across the partition (by bipartiteness). Looking at one configuration of the first excited subspace, it specifies another bipartition, this time with all but l edges lying across it. We are interested in configurations that are not isolated in G_{loc} because these nodes as mentioned in Sec. 4.1 do not play a role in AC occurrence. In such a configuration x , we want that by flipping one node (i.e. moving it to the other side of the partition), the number of uncut edges stays the same, in order to obtain a configuration y that is also a vertex of G_{loc} . So this specific node needs to have half of its edges that are uncut and the other half that are cut in this particular configuration x of the first excited subspace. This automatically restricts l to be both even and larger than $d/2$.

Case $l = d/2$: Let us suppose $l = d/2$. We are in a situation similar to Fig. 4.4 (left). By supposing that $l = d/2$, it means, in the configuration of one excited state, all other edges must go from left (L) to right (R). This splits the configuration in the classical L/R partition of a cut. Then we show the following claim that node 1 is a minimal separator of the graph which creates another split up (U) and down (D) (Fig 4.4 - right).

Claim 3. Assume that $l = d/2$ and let us consider a configuration corresponding to a non-isolated vertex of G_{loc} . Then there is a node of the input graph G , say node 1, having $d/2$ neighbors on each side of the configuration. Moreover, this node is a minimal separator of the graph (see Fig. 4.4).

Proof of claim 3. The configuration x is such that all edges but $l = d/2$ are cut, and this also holds after the bit-flip of one of its bits. Assume without loss of generality that this is the first bit, corresponding to node 1, and that 1 is on the left-hand side of the configuration, i.e., $1 \in L$. Since flipping node 1 from left to right maintains the number of cut edges, it means that 1 has exactly $d/2$ neighbors in set L and $d/2$ in set R . Since $l = d/2$, it also means that the l uncut edges are precisely the $d/2$ ones incident to node 1, from 1 to vertices of L .

Let $N_D(1)$ denote the set of neighbors of 1 in L , and $N_U(1)$ denote the set of neighbors of 1 in R . We prove that $N_D(1)$ and $N_U(1)$ are disconnected in graph $G - 1$, obtained from the input graph G by deleting vertex 1. By contradiction, assume there is a path P from $a \in N_U(1)$ to $b \in N_D(1)$ in $G - 1$. Path P together with vertex 1 form a cycle in graph G . By bipartiteness, this cycle is even, so at least one edge of the cycle, other than $\{1, b\}$, is contained in L or R . This is

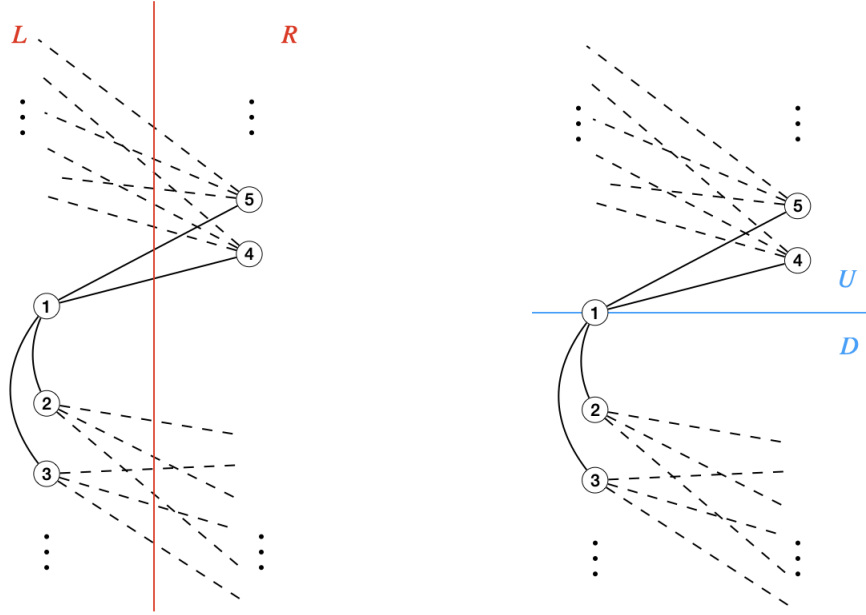


Figure 4.4: Construction of a specific first excited configuration with $d = 4$. The L/R partition (left) is natural in MaxCut. The U/D partition (right) is relevant if 1 is a minimal separator.

in contradiction with the assumption that $l = d/2$ and all of of these specific $d/2$ edges are linked to the same node 1. Therefore, $G - 1$ is disconnected. This proves claim 3. $\square$

This creates four quadrants UL , UR , DL and DR as follows: U is the subset of nodes of G formed by the union of connected components of $G - 1$ intersecting $N_U(1)$, and D is its complement. Then UL , UR , DL and DR are defined as the respective intersections of U and D with L and R ($UL = U \cap L$ and similar). The above considerations tell us that all edges of $G - 1$ go either from UL to UR or from DL to DR . Now, we call n_{DL} and n_{DR} the numbers of nodes in parts DL and DR (other than the labeled ones, i.e., the neighbors of node 1). By counting the edges from DL to DR , observe that these variables must satisfy the following equation:

$$\frac{d}{2}(d-1) + dn_{DL} = dn_{DR}$$

Because we know that d is even and, n_{DL} and n_{DR} are integers, the above equation cannot be satisfied.

Case $l > d/2$: l must be strictly larger than $d/2$, i.e. $l \in [\frac{d}{2} + 1, d]$. All these l uncut edges can be split between r_L and r_R , the ones on the left side and right side respectively and without loss of generality we choose that already $d/2$ of them are on the left side. So $r_L \in [\frac{d}{2}, d]$, $r_R \in [0, \frac{d}{2}]$ and $l = r_L + r_R$. Again we can count the

number of edges that lie across L and R and we end up with:

$$dn_L - 2r_L = dn_R - 2r_R$$

where $n_L = |L|$, $n_R = |R|$ and $n_L + n_R = n$ the total number of nodes. In a d -regular bipartite graph, n is necessarily even, so we have that

$$r_L - r_R = 2\left(\frac{n}{2} - n_R\right)\frac{d}{2} = kd \quad \text{for } k \in \mathbb{Z}$$

The potential values for r_L and r_R bring the interval for $r_L - r_R$ to $[0, d]$. So only $k = 0$ and $k = 1$ are possible. If $k = 0$, then $r_L = r_R = d/2$ so $l = d$. If $k = 1$, then $r_R = 0$, $r_L = d$ so $l = d$. In any case, the only possibility is to have $l = d$ which concludes the proof of claim 2. $\square$

These two claims simplify the expression of the different AC occurrence conditions, as follows:

- AC if $\mathbf{d}_{\text{avg}}(\text{loc}) > 4$;
- No-AC if $\mathbf{d}_{\text{max}}(\text{loc}) < 4$;
- Undefined if $\mathbf{d}_{\text{max}}(\text{loc}) > 4 > \mathbf{d}_{\text{avg}}(\text{loc})$.

We are left with a last thing to show to ensure that no AC occurs while solving MaxCut on d -regular bipartite graphs with AQC. To this purpose, we show this final lemma:

Lemma 4.1. *If $d \notin \{2, 4\}$ then $\mathbf{d}_{\text{max}}(\text{loc}) < 4$.*

Proof. Recall that odd values for d are already disregarded as d must be even. Suppose it is possible to have $\mathbf{d}_{\text{max}}(\text{loc}) \geq 4$ then it means that we need at least four nodes in a configuration such as Fig. 4.4, where half of their edges are uncut. Let us call F the set of these latter nodes, i.e. $|F| = \mathbf{d}_{\text{max}}(\text{loc})$. It means that there are at least $|F|\frac{d}{2} \geq 2d$ outgoing uncut edges from the nodes in F . By outgoing edge from a node, we mean the extremity of the edge that leaves the node (each edge contributes to two outgoing edges, one for each of its nodes). So here we count the number of edges that leave a node in F which are uncut. We are allowed at most d uncut edges to be a local minimum. So all of these $2d$ outgoing uncut edges need to generate exactly d edges. This remark forces $\mathbf{d}_{\text{max}}(\text{loc})$ to be smaller than 4, so suppose $\mathbf{d}_{\text{max}}(\text{loc}) = 4$. One node has only three possible neighbors for its $d/2$ uncut edges, so it is possible as long as $\frac{d}{2} \leq 3$, i.e., $d \leq 6$. For $d = 6$, linking all of these edges creates a triangle which makes the whole graph non-bipartite. $\square$

To finish the demonstration of Theorem 4.2, we still have to handle the cases for which $d \in \{2, 4\}$. We treat them separately and show that they fall in the Undefined regime so we directly use Theorem 4.1 to conclude.

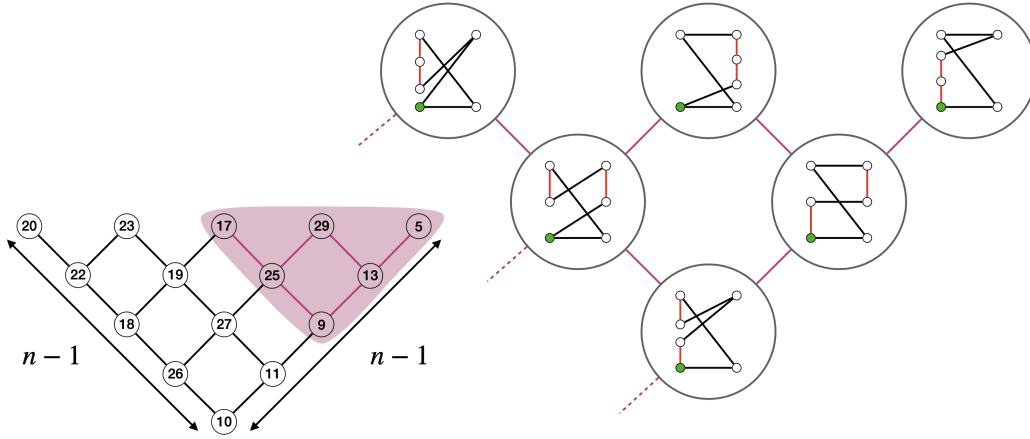


Figure 4.5: Left: G_{loc} of a cycle of size $n = 6$. Right: a precision of the first excited state configurations from the upper-right part of G_{loc} . Red edges are the uncut ones and the green node is the fixed node such that the top right node corresponding to node labeled 5 in G_{loc} is 000101. We easily see that nodes in $\{20, 23, 17, 29, 5\}$ are linked to the ground-state.

Case $d = 2$: As d must be even, we focus on even cycles. We see that it creates a large G_{loc} , see Fig. 4.5.

We can easily evaluate the average and maximum degree of this graph as :

$$\begin{aligned} \mathbf{d}_{\max}(\text{loc}) &= 4 \\ \mathbf{d}_{\text{avg}}(\text{loc}) &= 4 \frac{n-2}{n} = 4 \left(1 - \frac{2}{n}\right) \end{aligned}$$

These values bring the cycle in the Undefined regime. However, we can expect that **AQC** will easily work with a MaxCut on an even cycle because its G_{loc} is highly connected to the ground state. Fig. 4.1 shows how G_{loc} (which is the one in Fig. 4.5) is linked to the ground-state (blue edges). More precisely, there are $n - 1$ connections with the ground state in an $(n - 1)$ -regular graph. This means that there is no potential barrier to overcome going from states in G_{loc} to the $|\text{GS}\rangle$.

Another justification is to directly use the technical Theorem 4.1 which says that no AC occurs if $\lambda_0 < 4$, where λ_0 is the largest eigenvalue of G_{loc} . We know that $\lambda_0 = \mathbf{d}_{\max}(\text{loc})$ if and only if G_{loc} is $\mathbf{d}_{\max}(\text{loc})$ -regular, otherwise $\lambda_0 < \mathbf{d}_{\max}(\text{loc})$. So we are in the no-AC regime.

Case $d = 4$: By construction, in the case where $d = 4$, there is one possible configuration in a 4-regular graph that brings its G_{loc} in the Undefined regime. It can be artificially scaled up as shown in Fig. 4.6 :

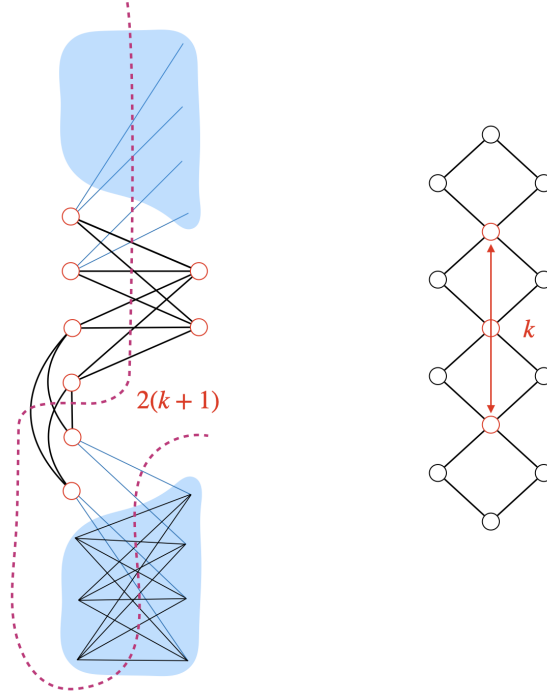


Figure 4.6: (left) 4-regular bipartite graph in one of its first excited state configurations and (right) G_{loc} where we disregarded the isolated node. Written in red, the number of red nodes ($k = 3$). In shaded blue, a part of the graph that completes the graph in a 4-regular one. The purple dashed line shows the bipartition of the graph.

We can easily derive the maximum and average degree of G_{loc} :

$$\begin{aligned} d_{\max}(\text{loc}) &= 4 \\ d_{\text{avg}}(\text{loc}) &= \frac{8(k+1)}{3k+4} = 2 + \frac{2k}{3k+4} \end{aligned}$$

where k is a parameter to construct the graph. G_{loc} is not connected to the ground state, so one can imagine that this will produce a potential barrier that creates an AC. But as one can see, the average degree of G_{loc} only tends to $2 + 2/3 < 4$, which is far from the AC appearance condition. A similar argument from the case $d = 2$ can be applied here when using directly the technical theorem with λ_0 . □

These above results allow us to conclude on the absence of anti-crossing during an adiabatic process to solve MaxCut over d -regular bipartite graphs and show theorem 4.2. One can deduce from this that there is no exponentially closing gap leading to a polynomial runtime to find the optimal cut in regular bipartite graphs via AQC. A natural question arises from this conclusion: Can we draw a similar conclusion for general bipartite graphs? We discuss this in the next section.

4.2.2 General bipartite graphs

In this section, we are interested in the behavior of the energies if we look at bipartite graphs in general. We construct a family of bipartite graphs that respect the condition of occurrence of an anti-crossing, meaning that exponentially closing gaps can arise even for MaxCut on bipartite graphs. Let $G(E, V)$ denote a bipartite graph. Similarly to the previous section, $\langle H_1 \rangle_0 = -\frac{|E|}{2}$, $E_{gs} = -|E|$ and $\Delta H_1 = l \in [1, \mathbf{d}_{\min}(G)]$. Claim 2 is still applicable with the minimum degree $\mathbf{d}_{\min}(G)$ of G . So $\Delta H_1 = \mathbf{d}_{\min}(G)$ and $n\alpha_{H_1}$ becomes $4\frac{\mathbf{d}_{\min}(G)}{\mathbf{d}_{\text{avg}}(G)}$. The condition for the different regimes can be written as follows:

- AC if $\mathbf{d}_{\text{avg}}(\text{loc}) > 4\frac{\mathbf{d}_{\min}(G)}{\mathbf{d}_{\text{avg}}(G)}$;
- No-AC if $\mathbf{d}_{\text{max}}(\text{loc}) < 4\frac{\mathbf{d}_{\min}(G)}{\mathbf{d}_{\text{avg}}(G)}$;
- Undefined if $\mathbf{d}_{\text{max}}(\text{loc}) > 4\frac{\mathbf{d}_{\min}(G)}{\mathbf{d}_{\text{avg}}(G)} > \mathbf{d}_{\text{avg}}(\text{loc})$.

We can see that the No-AC case is never verified by adapting the proof of Lemma 4.1. G_{loc} is generated by bit-flipping the nodes of G with degree $\mathbf{d}_{\min}(G)$. We know that there are $\mathbf{d}_{\min}(G)$ edges uncut in the first excited state, i.e. $2\mathbf{d}_{\min}(G)$ outgoing edges. By calling $\mathbf{d}_y(\text{loc})$ the degree of configuration y in G_{loc} , it means that there are $\mathbf{d}_y(\text{loc})$ nodes in G that can be flipped from y to give a configuration also in G_{loc} . Those $\mathbf{d}_y(\text{loc})$ nodes are necessarily of degree $\mathbf{d}_{\min}(G)$ and they count for $\frac{1}{2}\mathbf{d}_y(\text{loc})\mathbf{d}_{\min}(G)$ outgoing edges. So $\mathbf{d}_y(\text{loc})$ must verified the following inequality :

$$\begin{aligned} \frac{1}{2}\mathbf{d}_y(\text{loc})\mathbf{d}_{\min}(G) &\leq 2\mathbf{d}_{\min}(G) \\ \Rightarrow \mathbf{d}_y(\text{loc}) &\leq 4 \end{aligned} \tag{4.6}$$

The maximum of 4 is reached by the configuration where four nodes of G have degree $\mathbf{d}_{\min}(G)$ and can be flipped. This brings the No-AC condition to $\frac{\mathbf{d}_{\min}(G)}{\mathbf{d}_{\text{avg}}(G)} > 1$ which is never verified, with the limit case of regular graphs. So let us focus on the AC condition.

The first point gives us the condition for a graph G that produces an anti-crossing under an AQC evolution for the MaxCut problem. First, looking only at the right-hand side, the ratio $\frac{\mathbf{d}_{\min}(G)}{\mathbf{d}_{\text{avg}}(G)}$ is small for highly irregular graphs. From Eq. (4.6), the average degree for G_{loc} is certainly smaller than 4 so we need to play with the degree of G . Even though we remove the regularity hypothesis, we can still use some results from the above cases. Indeed, in that setting, G_{loc} arises from the bipartition of a $\mathbf{d}_{\min}(G)$ -regular induced subgraph of G . We look at graphs G with a large average degree but with also a small minimum degree and a large $\mathbf{d}_{\text{avg}}(\text{loc})$. The cycle produces the densest G_{loc} but it is highly connected to the ground-state and the average degree of the cycle is not quite large. The idea is to attach two complete bipartite graphs (K_{rr}, K_{ll}) that will increase the average degree of the graph by two parallel sequences of nodes of degree 2 (P_1, P_2) that will create the dense G_{loc} and small $\mathbf{d}_{\min}(G)$ of value 2. Fig. 4.7 provides an example of a such a

graph with $r = l = 3$ and where P_1 and P_2 are sequences of $k_1 = k_2 = 2$ adjacent nodes of degree 2. k_1 and k_2 need to be of the same parity to ensure bipartiteness of the whole graph. Three configurations of the same graph are shown, corresponding to the ground-state (left), and two configurations of the first excited subspace (middle, right), that create the different components in G_{loc} (Fig 4.8).

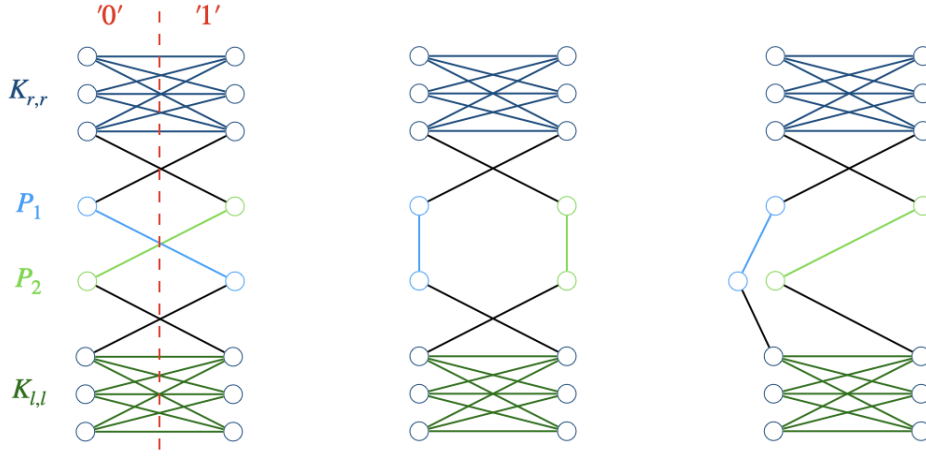


Figure 4.7: Configurations of G in its ground-state (left) and first excited state. (middle) a configuration far from GS, (right) a configuration neighboring GS.

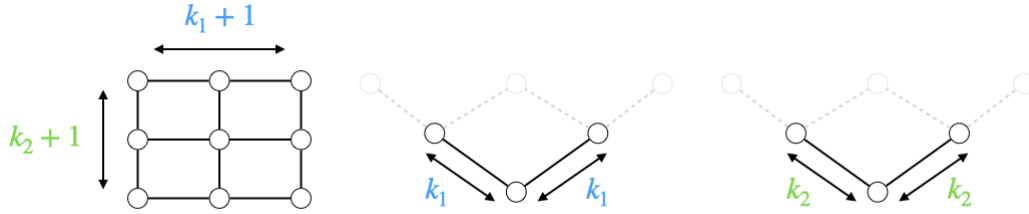


Figure 4.8: G_{loc} of graph G . Three components : (middle) and (right) similarly: components corresponding to states in a configuration close to the one on (Fig 4.7 - right) and (left) component corresponding to states in the configuration of (Fig 4.7 - middle). The light dashed grey edges and nodes show how these two components grow when $k_i > 2$.

The largest component of G_{loc} is a lattice of size $(k_1 + 1, k_2 + 1)$ if k_i represents the number of nodes in P_i . It is far away from the ground-state as we need to flip at least all the nodes of the $K_{r,r}$ part. The two other components can be viewed as subgraphs of the large component so they have smaller eigenvalues than the largest one of the lattice; they are also strongly connected to the ground-state. Fig. 4.9

shows the details of the relation between the nodes of G_{loc} and graph configurations in a left/right partition. The middle configuration of Fig. 4.7 corresponds to the middle node of the lattice in G_{loc} . Then moving each node in blue or green produce another configuration with the same edge penalty.

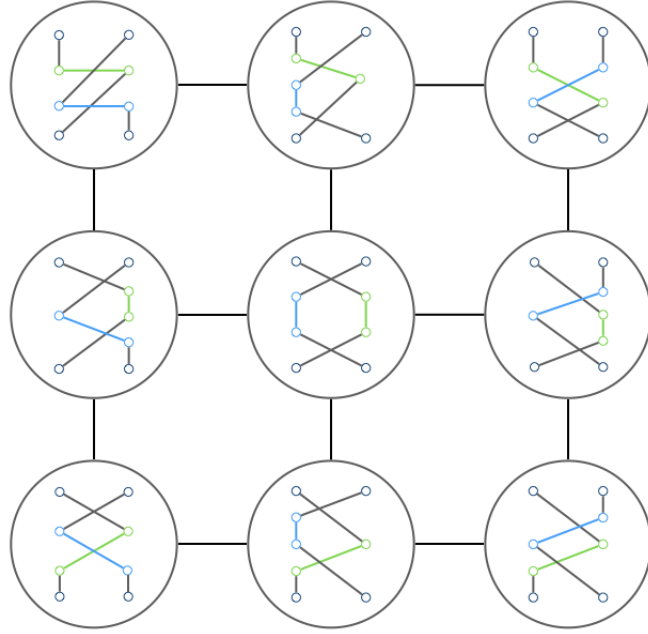


Figure 4.9: Details of the large component of G_{loc} and how each configuration is related by bit-flip. We intentionally omit the drawing of the $K_{r,r}$ and $K_{l,l}$ which do not play a role in G_{loc} .

We directly have that $\mathbf{d}_{\min}(G) = 2$. Now, we need to derive the average degree of G and of the largest component of G_{loc} (its maximum degree being 4).

$$\begin{aligned}
\mathbf{d}_{\text{avg}}(\text{loc}) &= \frac{4 * 2 + 2(k_1 - 1 + k_2 - 1) * 3 + (k_1 - 1)(k_2 - 1) * 4}{(k_1 + 1)(k_2 + 1)} \\
&= \frac{4k_1k_2 + 2(k_1 + k_2)}{(k_1 + 1)(k_2 + 1)} \\
&= 4 \left(1 - \frac{1 + \frac{1}{2}(k_1 + k_2)}{(k_1 + 1)(k_2 + 1)} \right) \\
&= 4 \left(1 - \frac{1}{k + 1} \right) \quad \text{for } k = k_1 = k_2
\end{aligned}$$

$$\begin{aligned}
\mathbf{d}_{\text{avg}}(G) &= \frac{(k_1 + k_2) * 2 + 2r * r + 2l * l + 4}{k_1 + k_2 + 2r + 2l} \\
&= \frac{2k + r^2 + l^2 + 2}{k + r + l} \quad \text{for } k = k_1 = k_2
\end{aligned}$$

Let us solve the equation $\mathbf{d}_{\text{avg}}(\text{loc}) > 4 \frac{\mathbf{d}_{\min}(G)}{\mathbf{d}_{\text{avg}}(G)}$ with $\mathbf{d}_{\min}(G) = 2$.

$$\begin{aligned}
\mathbf{d}_{\text{avg}}(\text{loc}) &> 4 \frac{\mathbf{d}_{\min}(G)}{\mathbf{d}_{\text{avg}}(G)} \\
\Rightarrow 1 - \frac{1}{k + 1} &> \frac{2(k + r + l)}{2k + r^2 + l^2 + 2} \\
\Rightarrow \frac{r^2 + l^2 + 2 - 2r - 2l}{2k + r^2 + l^2 + 2} &> \frac{1}{k + 1} \\
\Rightarrow (k + 1)(r^2 + l^2 + 2 - 2r - 2l) &> 2k + r^2 + l^2 + 2 \\
\Rightarrow k(r^2 + l^2 - 2r - 2l) &> 2r + 2l \\
\Rightarrow k &> \frac{2(r + l)}{r(r - 2) + l(l - 2)} \\
\Rightarrow k &> \frac{2(r + 3)}{r(r - 2) + 3} \quad \text{for } l = 3
\end{aligned}$$

We have a limit at $r = 3$ and $k = 2$ for a graph of size 16. Then the smallest graphs that satisfy the condition are for $r = 3$ and $k = 3$ or $r = 4, l = 3$ and $k = 2$ which bring the size of the smallest graphs satisfying the AC condition to 18 nodes.

This above construction shows that there exist bipartite graphs that exhibit an AC. The presence of an anti-crossing implies an exponentially closing gap bringing the provable runtime to find the optimal cut exponentially large in the size of the graph. This construction can be scaled up easily by growing the parameters k, r and l . In the next section, we numerically investigate the presence of AC on graphs of this family to support this theoretical result.

4.2.3 Numerical study: AC and other observations

In this section, we give some numerical evidence of the occurrence of the AC in the particular family we constructed in Sec. 4.2.2. The goal is to observe the behavior of the minimum gap and to confirm the exponentially closing gap. We then discuss whether or not these gaps lead to a computational inefficiency of AQC and moderate the term AC by looking at the more mathematical definition we introduced in Sec. 3.2.

Minimum gap study: Let us first show that the value of the minimum gap supports the theoretical results derived in Secs. 4.1 and 4.2.2. To compute this quantity for large graphs, we use the SciPy library [Jones *et al.* 01] and its optimized method, `scipy.sparse.linalg.eigs`, for matrices with a sparse representation. Our Hamiltonians have a sparse representation in the Pauli basis, enabling us to compute the minimum gap for graphs with up to 28 nodes.

To satisfy the conditions required for our application, we fix one node of the graph to lift the standard MaxCut symmetry. Specifically, we fix one node of the $K_{l,l}$ part on the left (L) side of the partition. We consider the family of graphs G_{rk} with the same structure as in the previous section, where we fix $l = 3$ and assume $k_1 = k_2 = k$. Therefore, we can vary two parameters (Fig. 4.10 shows the schematic energy landscape of H_1 for G_{rk}):

- Increasing r increases the distance between G_{loc} and the ground-state in the hypercube, as all the $K_{r,r}$ part needs to be flipped (fixing one node in the $K_{l,l}$ part blocks the possibility to flip this part entirely),
- Increasing k creates a larger G_{loc} , resulting in a larger local minimum that is not linked to the ground-state, but also increases the two other parts of G_{loc} connected to it.

We denote $\Delta_{rk}(s)$ as the difference between the two lowest instantaneous eigenvalues of $H(s)$ associated with G_{rk} , i.e., the spectral gap of the time-dependent Hamiltonian. We plot these gaps in Fig. 4.11 (a) by varying r and k . Specifically, we observe that increasing r by 1 divides the gap by 2. To illustrate this, we also plot Fig. 4.11 (b) the minimum gap of Δ_{rk} for $k = 2$ against r . We fit this curve with an exponentially decreasing function of r . When k is fixed, it is straightforward to see that $r \simeq \frac{n}{2}$.

Fig. 4.11 supports the main theorem in Sec. 4.1 and the construction in Sec. 4.2. The distance to the ground-state appears to play a major role in the minimum gap compared to the size of G_{loc} . Remember that G_{loc} has three components and two of them are linked to the ground-state while the other one is a lattice far from the ground-state. Increasing k also increases the width around the ground-state, making it easier to reach than if it were isolated while increasing r has no impact on G_{loc} .

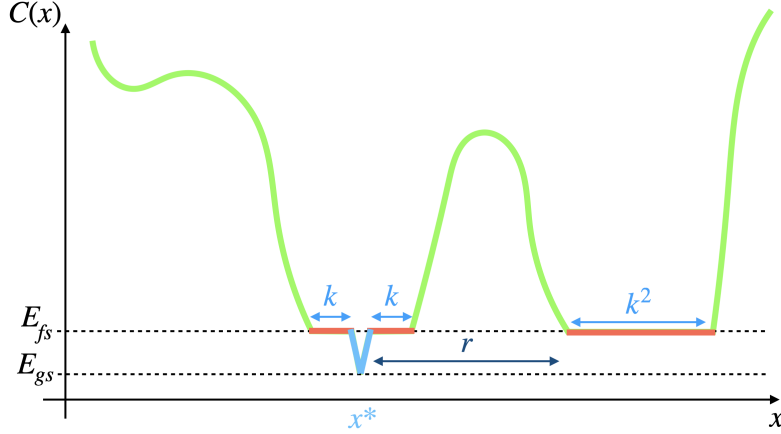


Figure 4.10: Schematic energy landscape of the MaxCut function on an instance G_{rk} and how r and k affect it.

Typically, it is assumed that an exponentially closing gap implies the failure of AQC [Altshuler *et al.* 2010]. In the next paragraph, we investigate the probability of measuring the ground-state at the end of a quantum evolution after a time $t_{\max}$ and discuss about the AC definition, which opens a new question on the computational efficiency of AQC.

Discussions about AC and AQC failure: Now that we have established the exponentially small gaps for the graph G_{rk} when r is increasing, we can wonder if it can be deduced that AQC is inefficient to solve those instances, as this is the usual deduction from small gaps. In Fig. 4.12, we observe the probability p_{rk} of measuring the ground-state at the end of a quantum annealing (QA) evolution for different instances of G_{rk} as a function of $t_{\max}$. This plot was obtained using the AnalogQPU of the Eviden quantum software. Surprisingly, the probability seems to reach the value around 0.5 faster than expected, meaning in a time $t_{\max}$ that does not appear to depend too much on the size of the graph. This observation is not a contradiction of the adiabatic theorem, as it will certainly converge to 1 in an exponentially long runtime. It could also be just a scale illusion: for much larger graphs, the probability might stay at zero for a longer time than observed here, but this is not what the point below suggests. However, it raises questions about the effectiveness of AQC in practical applications even when exponentially small gaps are present.

The observed gaps in Fig. 4.11 exhibit an exponentially closing behavior, which is a signature of the AC phenomenon we are looking at. However, the computational complexity does not seem to be affected, in the sense that a constant probability to obtain the optimal solution is reached in a time that does not seem to depend too much on the graph size. We can notice in Fig. 4.11 (a) that the trend of the gaps appears to be softer compared to other observed ACs [Bapst & Semerjian 2012],

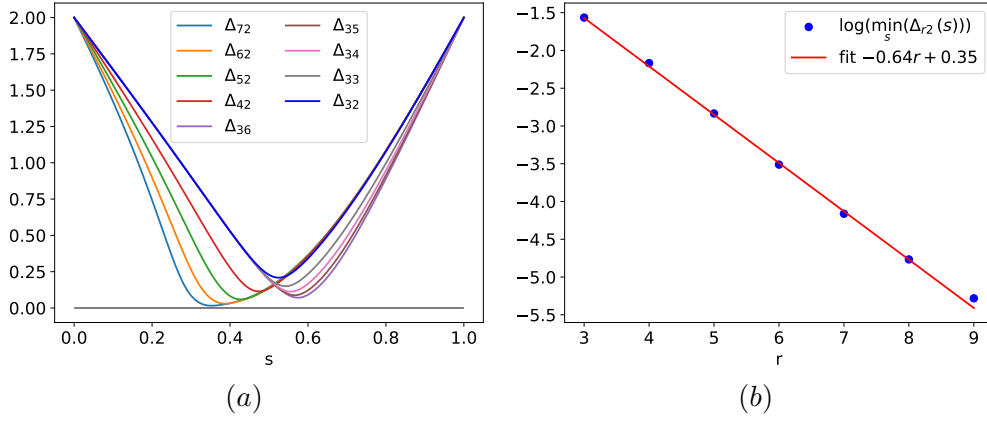


Figure 4.11: (a) Evolution of the spectral gap $\Delta_{rk}(s)$ and (b) minimum gap of Δ_{r2} for r going from 3 to 9 in logarithmic scale. It fits an exponentially decreasing tendency.

indicating a smoother transition. To address this observation, we use our work and intuition developed in Sec. 3.2 that proposed a more formal definition of anti-crossings that involves a new set of quantities. Given that only one AC is observed, Def. 3.1 should apply here. Let $g_0(s) = |\langle \phi_0(s) | \text{GS} \rangle|^2$ and $g_1(s) = |\langle \phi_1(s) | \text{GS} \rangle|^2$ be the overlap squared of the instantaneous eigenstates (zeroth and first respectively) of $H(s)$ with the ground-state $|\text{GS}\rangle$ of H_1 . Typically, at an anti-crossing point, these curves undergo a harsh exchange of position. If $g_0(s)$ smoothly increases toward 1, it is not an AC according to our definition. For the graph G_{rk} , the conditions given in this formal definition do not seem to be fully satisfied, as the plots in Fig. 4.13 show. On the left, an example of behavior of g_0 and g_1 when AC happens, the curves experience an almost discontinuity at AC point, changing the position of g_0 and g_1 (in the middle and right-hand plots, g_0 and g_1 for instances G_{32} and G_{72} respectively). In the G_{32} case, g_1 starts to become bigger than g_0 but it produces only a little bump and g_0 has a smooth growth toward 1. One could think that this phenomenon is due to the small size of the instance, and that by considering larger instances but with very small gaps, we would observe a “typical” AC behavior. However, on the G_{72} case, where the size increases and the gap decreases, this bump totally disappears and we can only attest to a smooth growth of g_0 . This observation indicates the opposite of an AC behavior leading to an efficient AQC evolution to solve these instances. This raises the question of whether every exponentially closing gap necessarily leads to a failure of AQC, or if AC is a particular event that creates an exponentially closing gap leading to a complete leak of the probability distribution into higher energy levels.

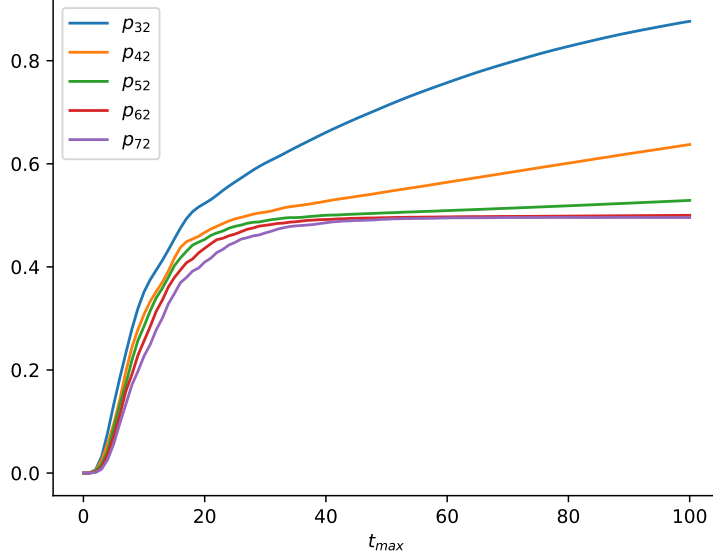


Figure 4.12: Probability of measuring the ground-state after a time $t_{\max}$ for instances with $k = 2$ and $r \in [3, 7]$.

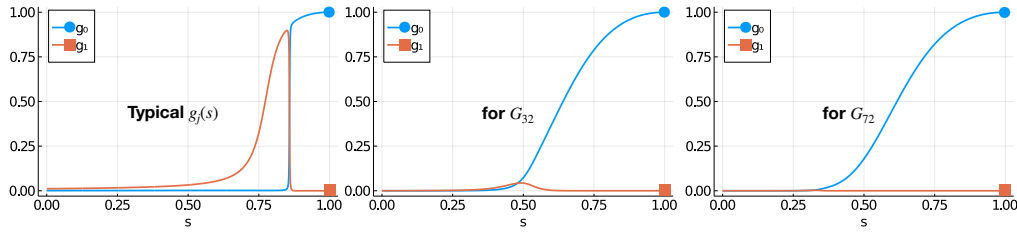


Figure 4.13: Evolution of $g_0(s)$ in blue and $g_1(s)$ in red for graph G_{72} (right), G_{32} (middle) and a typical behavior (left) during an AC like in Section 3.2.

4.2.4 Validation of perturbative expansion and limitations

We discuss here the validation of this expansion at first-order in the case of MaxCut over d -regular bipartite graphs. We need to look at the second order term and compare it to the first or zeroth order term. Eventually we give a little conclusion of the work developed in Sec. 4.1 and 4.2.

Delocalized state expansion: The eigen-basis of the initial Hamiltonian H_0 can be written as

$$|E_b\rangle = \frac{1}{\sqrt{2^n}} \sum_{x \in \{0,1\}^n} (-1)^{b \cdot x} |x\rangle,$$

where b is an n -bit-string and the centered dot stands for the scalar product over $\mathbb{F}_2^n$. There are $n + 1$ different eigen-levels where the k^{th} eigen-space has degeneracy

$\binom{n}{k}$ and correspond to eigen-states with bit-string b of Hamming weight $|b| = k$ and eigen-value $E_b^I = -n + 2|b|$ (see [Cvetkovic *et al.* 1980] for more details). With this notation, we can write $|\psi_0\rangle$ as $|E_{00\dots 00}\rangle$. We are interested in

$$E_0^{(2)} = \sum_{b \neq 00\dots 0} \frac{|\langle E_b | H_1 | \psi_0 \rangle|^2}{E_0^I - E_b^I}$$

For MaxCut problem on graph G , we know that $\langle E_b | H_1 | \psi_0 \rangle = -1/2$ if and only if G_b is exactly one edge (see Appendix B.1). G_b is the graph induced by the node i where $b_i = 1$. Therefore $E_0^{(2)} = -\frac{|E(G)|}{16}$. We have $E_0^{(1)} = \langle H_1 \rangle_0 \simeq -\frac{|E(G)|}{2}$ so $\frac{|E_0^{(2)}|}{|E_0^{(1)}|} = \frac{1}{8} < 1$.

Ground-state expansion: The eigen-basis of the final Hamiltonian H_1 is the canonical basis of the bit-string $|x\rangle$ with energy E_x , and we named $|\text{GS}\rangle$ the bit-string corresponding to the ground-state with energy E_{gs} . It follows that the second-order term is

$$E_{gs}^{(2)} = \sum_{x \in \{0,1\}^n} \frac{|\langle x | H_0 | \text{GS} \rangle|^2}{E_{gs} - E_x}$$

where $|\langle x | H_0 | \text{GS} \rangle| = 1$ if and only if the bit-string x is at exactly one bit-flip from the bit-string GS . We can rewrite it like

$$E_{gs}^{(2)} = \sum_{x \sim_{H_0} \text{GS}} \frac{1}{E_{gs} - E_x}$$

For the MaxCut problem on a d -regular bipartite graph, we can further simplify. Indeed, from the ground-state, flipping one bit gives an energy state $|x\rangle$ of exactly $E_x = E_{gs} + d$. So we end up with $E_{gs}^{(2)} = -\frac{n}{d}$. We have $E_{gs}^{(1)} = 0$ and $E_{gs}^{(0)} = E_{gs} = \frac{dn}{2}$ which leads to $\frac{|E_{gs}^{(2)}|}{|E_{gs}^{(0)}|} = \frac{2}{d^2} < 1$. For $d = 4$ we have the same value as for the delocalized state.

Local minima expansion: We work in the same basis as for the latter expansion and we look at

$$E_{fs}^{(2)} = \sum_{x \notin V(G_{loc})} \frac{|\langle x | H_0 | \text{FS}, 0 \rangle|^2}{E_{fs} - E_x} \leq \sum_{x \notin V(G_{loc})} \sum_{y \sim_{H_0} x} \frac{|\langle y | \text{FS}, 0 \rangle|^2}{E_{fs} - E_x}$$

The size of this double sum is the number of connections G_{loc} has with the whole hypercube, i.e. $|\partial G_{loc}|$. The term $|\langle y | \text{FS}, 0 \rangle|^2$ is large when the degree of node y in G_{loc} is large so with less occurrence in the above double sum. On average, when a graph is regular its vector coordinate value of the largest eigenvalue is $\frac{1}{\sqrt{|V(G_{loc})|}}$.

By introducing the conductance of the subgraph G_{loc} as $h(\text{loc}) = \frac{|\partial G_{loc}|}{|V(G_{loc})|}$, we can upper bound the second-order term with

$$|E_{fs}^{(2)}| \leq h(\text{loc}) \frac{1}{\min_x |E_{fs} - E_x|}$$

We know that $|E_{fs}^{(1)}| = \lambda_0(\text{loc}) \geq d_{\text{avg}}(\text{loc}) = n - h(\text{loc})$. So the ratio we need to check is $\frac{h(\text{loc})}{n-h(\text{loc})} \frac{1}{\min_x |E_{fs} - E_x|}$ which is smaller if G_{loc} is neighboring high energy states.

Conclusion

In this second technical chapter, we studied the occurrences of **AC** characterized only by exponentially closing gaps in a quantum evolution. We first provided a technical theorem that gives a condition to distinguish between the cases where **AC** can appear or not. This theorem assumes some conditions on H_1 and the validity of the perturbation expansion at first-order. We only suggest a validation for the MaxCut problem over d -regular bipartite graphs which allows us to prove the efficiency of **AQC** in this specific case. Although it is well known that this problem is trivially solvable classically, to the best of our knowledge there was no proof of the efficiency of **AQC** in this restrictive class of graphs. We also want to draw reader's attention on the validity we suggested as we only compared to next order term. A more rigorous approach would need to look at any order and show that under some condition on s_{lg} the expansion is valid. In the next section we give some hints on the higher order of the expansion.

More significantly, the tools we developed allowed us to construct a certain family of irregular bipartite graphs that satisfy the condition for **AC** occurrence. We did not rigorously validate the perturbative expansion for general bipartite graphs but the numerical computation of the minimum gap showed the exponential closure of the gap with the input size. This suggests that enough irregularity may lead to **AQC** inefficiency over this class of graphs. However, this is not what the numerical calculation of the probability of measuring the ground-state hints. It appears that a constant probability is reached in a time that does not strongly depend on the input size. Our insights about **AC** in Sec. 3.2 seems to indicate no localization of the instantaneous ground-state and a smooth evolution leads the quantum state to $|\text{GS}\rangle$ "easily".

The last section is dedicated to a current unfinished work where we develop higher order of the perturbative expansion for the MaxCut problem.

4.3 Higher order perturbative expansion for MaxCut

In this section we try to develop higher order of the perturbation theory by using the following proposition due to my supervisor Simon Martiel :

Proposition 4.1 (see Appendix B.1). *If H_1 is the Hamiltonian encoding the Max-Cut problem over a graph of n nodes and H_0 is the standard bit-flip operator with eigenpairs $(|E_b\rangle, E_b)$ for a bit-string b where $E_b = -n + 2|b|$, $|b|$ being the Hamming weight of b , then*

$$\langle E_b | H_1 | E_{b'} \rangle = \frac{1}{2}$$

if and only if $G_{b \oplus b'}$ is exactly an edge. Where $G_{b \oplus b'}$ is the subgraph induced by nodes i such that $(b \oplus b')_i = 1$.

The eigenvectors $|E_b\rangle$ of H_0 correspond to states where qubit i is in state $|+\rangle = \frac{|0\rangle + |1\rangle}{2}$ if $b_i = 0$ and $|-\rangle = \frac{|0\rangle - |1\rangle}{2}$ if $b_i = 1$. In the computational basis it is written as:

$$|E_b\rangle = \frac{1}{\sqrt{2^n}} \sum_{x \in \{0,1\}^n} (-1)^{b \cdot x} |x\rangle$$

The interesting result of Proposition 4.1 will allow us to take the perturbative expansion of the initial state $|\psi_0\rangle = |E_{0^n}\rangle$ further. Motivated by the recent result in [Dalzell et al. 2023] where the authors show an algorithm with a super Grover speed-up by essentially approximating the instantaneous ground-state close to the beginning $|\phi_0(s')\rangle$ for s' close to zero, we tackle the expansion of the initial state to see if anything relevant comes out. The super Grover speed-up comes as the overlap of the ground-state at s' with the final ground-state is slightly better than the usual $2^{-\frac{n}{2}}$ by a factor $c : 2^{-(\frac{n}{2}-c)}$.

We directly look at the time-dependent Hamiltonian $H(s) = (1-s)H_0 + sH_1 = H_0 + s(H_1 - H_0)$ where H_0 , the bit-flip operator and H_1 the MaxCut Hamiltonian (see Sec. 2.2.2). By doing this, the expansion at any time s is the instantaneous ground-state. So we suppose that there exist expansions of the ground-state and its energy like:

- $|\phi_0(s)\rangle = |E_{0^n}\rangle + s|E_{0^n}^{(1)}\rangle + s^2|E_{0^n}^{(2)}\rangle + \dots$
- $E_0(s) = E_0^{(0)} + sE_0^{(1)} + s^2E_0^{(2)} + \dots$

such that $H(s)|\phi_0(s)\rangle = E_0(s)|\phi_0(s)\rangle$ with the assumption that $\langle E_{0^n}^{(i)} | E_{0^n}^{(j)} \rangle = \delta_{ij}$. We identify the different terms in s^j :

$$\begin{aligned} s^0: & H_0|E_{0^n}\rangle = E_0^{(0)}|E_{0^n}\rangle \\ s^1: & (H_1 - H_0)|E_{0^n}\rangle + H_0|E_{0^n}^{(1)}\rangle = E_0^{(0)}|E_{0^n}^{(1)}\rangle + E_0^{(1)}|E_{0^n}\rangle \\ s^2: & (H_1 - H_0)|E_{0^n}^{(1)}\rangle + H_0|E_{0^n}^{(2)}\rangle = E_0^{(0)}|E_{0^n}^{(2)}\rangle + E_0^{(1)}|E_{0^n}^{(1)}\rangle + E_0^{(2)}|E_{0^n}\rangle \\ s^3: & (H_1 - H_0)|E_{0^n}^{(2)}\rangle + H_0|E_{0^n}^{(3)}\rangle = E_0^{(0)}|E_{0^n}^{(3)}\rangle + E_0^{(1)}|E_{0^n}^{(2)}\rangle + E_0^{(2)}|E_{0^n}^{(1)}\rangle + E_0^{(3)}|E_{0^n}\rangle \\ & \vdots \\ s^k: & (H_1 - H_0)|E_{0^n}^{(k-1)}\rangle + H_0|E_{0^n}^{(k)}\rangle = \sum_{l=0}^k E_0^{(k-l)}|E_{0^n}^{(l)}\rangle \end{aligned}$$

We know that $E_0^{(0)} = -n = E_0$, $E_b = -n + 2|b|$ and to get $E_0^{(k)}$ we project onto $\langle E_{0^n} |$ and then project onto another state $\langle E_b |$ to get $\langle E_b | E_{0^n}^{(k)} \rangle$. Using Proposition 4.1,

we can explicit the terms as :

$$\begin{aligned}
E_0^{(1)} &= \langle E_0^n | (H_1 - H_0) | E_0^n \rangle = \langle H_1 \rangle_0 + n \\
|E_0^{(1)}\rangle &= \sum_{b \neq 0^n} |E_b\rangle \frac{\langle E_b | H_1 | E_0^n \rangle}{E_0 - E_b} = \frac{1}{4 * 2} \sum_{b \text{ st } G_b \equiv \text{edge}} |E_b\rangle \\
E_0^{(2)} &= \langle E_0^n | (H_1 - H_0) | E_0^{(1)} \rangle = \sum_{b \neq 0^n} \frac{|\langle E_b | H_1 | E_0^n \rangle|^2}{E_0 - E_b} = -\frac{|E(G)|}{4 * 2^2} \\
|E_0^{(2)}\rangle &= \sum_{b \neq 0^n} \frac{|E_b\rangle}{(E_0 - E_b)} \left(\langle E_b | (H_1 - H_0) | E_0^{(1)} \rangle - E_0^{(1)} \langle E_b | E_0^{(1)} \rangle \right) \\
&= \sum_{b \neq 0^n} \frac{|E_b\rangle}{(E_0 - E_b)} \left(\sum_{b' \neq 0^n} \frac{\langle E_b | (H_1 - H_0) | E_{b'} \rangle \langle E_{b'} | H_1 | E_0^n \rangle}{(E_0 - E_{b'})} - E_0^{(1)} \frac{\langle E_b | H_1 | E_0^n \rangle}{(E_0 - E_b)} \right) \\
&= \sum_{b \neq 0^n} \frac{|E_b\rangle}{(E_0 - E_b)} \left(\sum_{b' \neq 0^n} \frac{\langle E_b | H_1 | E_{b'} \rangle \langle E_{b'} | H_1 | E_0^n \rangle}{(E_0 - E_{b'})} - (E_b + E_0^{(1)}) \frac{\langle E_b | H_1 | E_0^n \rangle}{(E_0 - E_b)} \right) \\
&= \frac{1}{4^2 * 2} \left(-(\langle H_1 \rangle_0 + 4) \sum_{G_b \equiv \text{edge}} |E_b\rangle + \frac{1}{2} \sum_{\substack{G_b \equiv P_2 \\ \text{extremities}}} |E_b\rangle + \frac{1}{2^2} \sum_{G_b \equiv 2 \text{ non-adjacent edges}} |E_b\rangle \right) \\
E_0^{(3)} &= \langle E_0^n | (H_1 - H_0) | E_0^{(2)} \rangle \\
&= \frac{1}{4^2 * 2^2} \left(-|E(G)| (\langle H_1 \rangle_0 + 4) + \frac{3}{2} |\{\text{triangle} \in G\}| \right)
\end{aligned}$$

where we used the fact that $\langle E_b | H_1 | E_{b'} \rangle \langle E_{b'} | H_1 | E_0^n \rangle = \frac{1}{4}$ if and only if $G_{b'}$ is exactly an edge and $G_{b \oplus b'}$ as well. This happens when G_b selects a pair of nodes that are the extremities of a path of length 2, $P_2 = \{G_{b'}, G_{b \oplus b'}\}$ or when G_b is two non-adjacent edges with one of them being $G_{b'}$. Then when looking at the energy it

further imposes that G_b is also an edge forming a triangle with P_2 .

$$\begin{aligned}
|E_{0^n}^{(3)}\rangle &= \sum_{b \neq 0^n} \frac{|E_b\rangle}{(E_0 - E_b)} \left(\langle E_b | (H_1 - H_0) | E_0^{(2)} \rangle - E_0^{(1)} \langle E_b | E_0^{(2)} \rangle - E_0^{(2)} \langle E_b | E_0^{(1)} \rangle \right) \\
&= \sum_{b \neq 0^n} \frac{|E_b\rangle}{(E_0 - E_b)} \left[\sum_{b', b'' \neq 0^n} \frac{\langle E_b | H_1 | E_{b'} \rangle \langle E_{b'} | H_1 | E_{b''} \rangle \langle E_{b''} | H_1 | E_{0^n} \rangle}{(E_0 - E_{b'}) (E_0 - E_{b''})} \right. \\
&\quad - \sum_{b' \neq 0^n} \frac{\langle E_b | H_1 | E_{b'} \rangle \langle E_{b'} | H_1 | E_{0^n} \rangle}{(E_0 - E_{b'})} \left(\frac{E_0^{(1)} + E_{b'}}{E_0 - E_{b'}} + \frac{E_0^{(1)} + E_b}{E_0 - E_b} \right) \\
&\quad \left. + \left(\frac{(E_0^{(1)} + E_b)^2}{E_0 - E_b} - E_0^{(2)} \right) \frac{\langle E_b | H_1 | E_{0^n} \rangle}{(E_0 - E_b)} \right] \\
&= \frac{-1}{2 * 4^2} \left(\frac{(\langle H_1 \rangle_0 + 4)^2}{4} + E_0^{(2)} \right) \sum_{G_b \equiv \text{edge}} |E_b\rangle + 2 \frac{\langle H_1 \rangle_0 + 4}{2^2 * 4^2} \sum_{\substack{G_b \equiv P_2 \\ \text{extremities}}} |E_b\rangle \\
&\quad + \frac{1}{2^2 * 4} \left(\frac{\langle H_1 \rangle_0 + 4}{4} + \frac{\langle H_1 \rangle_0 + 8}{8} \right) \sum_{\substack{G_b \equiv 2 \text{ non-} \\ \text{adjacent edges}}} |E_b\rangle \\
&\quad - \frac{1}{2^3 * 4^3} \sum_{\substack{G_b \equiv P_3 \\ \text{extremities}}} |E_b\rangle \\
&\quad - \frac{1}{2^3 * 4^2 * 8} \left(\sum_{\substack{G_b \equiv P_3 \\ \text{extremities}}} |E_b\rangle + \sum_{G_b \equiv \text{edge}} |E_b\rangle + \sum_{\substack{G_b \equiv \text{edge} \\ \text{under a hat}}} |E_b\rangle \right) \\
&\quad - \frac{1}{2^3 * 4 * 8 * 12} \sum_{\substack{G_b \equiv 3 \text{ non-} \\ \text{adjacent edges}}} |E_b\rangle \\
E_0^{(4)} &= -\frac{|E(G)|}{2^2 * 4^3} \left((\langle H_1 \rangle_0 + 4)^2 - \frac{|E(G)|}{2^2} \right) + 3|\{T \in G\}| \frac{\langle H_1 \rangle_0 + 4}{2^2 * 4^2} \\
&\quad - 2|\{S \in G\}| \left(\frac{1}{2^3 * 4^3} + \frac{1}{2^3 * 4^2 * 8} \right) - \frac{|E(G)|}{2^4 * 4^2 * 8} \\
&\quad \vdots \\
E_0^{(k)} &= \langle E_{0^n} | H_1 | E_0^{(k-1)} \rangle \\
\langle E_b | E_0^{(k)} \rangle &= \frac{1}{E_0 - E_b} \left(\langle E_b | H_1 | E_0^{(k-1)} \rangle - (E_b + E_0^{(1)}) \langle E_b | E_0^{(k-1)} \rangle \right. \\
&\quad \left. - \sum_{l=1}^{k-2} E_0^{(k-l)} \langle E_b | E_0^{(l)} \rangle \right)
\end{aligned}$$

where S stands for square and T for triangle and the following claim:

Claim 4. For bit-strings, b, b' and b'' , the product $\langle E_b | H_1 | E_{b'} \rangle \langle E_{b'} | H_1 | E_{b''} \rangle \langle E_{b''} | H_1 | E_{0^n} \rangle$ equals to $\frac{1}{2^3}$ if $G_{b''}$, $G_{b' \oplus b''}$ and $G_{b \oplus b'}$ are exactly an edge. These conditions are reunited in five different situation :

- (a) - $|b'| = 2$ sharing one bit with b'' and G_b selects a pair of nodes that are the extremities of a path of length three: $P_3 = G_{b''} \cap G_{b' \oplus b''} \cap G_{b \oplus b'}$.
- (b) - $|b'| = 2$ sharing one bit with b'' and G_b is a graph with four nodes and at least one edge non adjacent to $G_{b''}$, called *edge under the hat*. The hat being the path P_2 .
- (c) - $G_{b'}$ is two non-adjacent edges (which include $G_{b''}$) and G_b selects a pair of nodes that are the extremities of a path of length three: $P_3 = G_{b''} \cap G_{b' \oplus b''} \cap G_{b \oplus b'}$.
- (d) - $G_{b'}$ is two non-adjacent edges (which include $G_{b''}$) and $b = b''$ or $b = b'' \oplus b'$, i.e. G_b is exactly an edge.
- (e) - $G_{b'}$ is two non-adjacent edges (which include $G_{b''}$) and G_b is three non-adjacent edges, including $G_{b'}$.

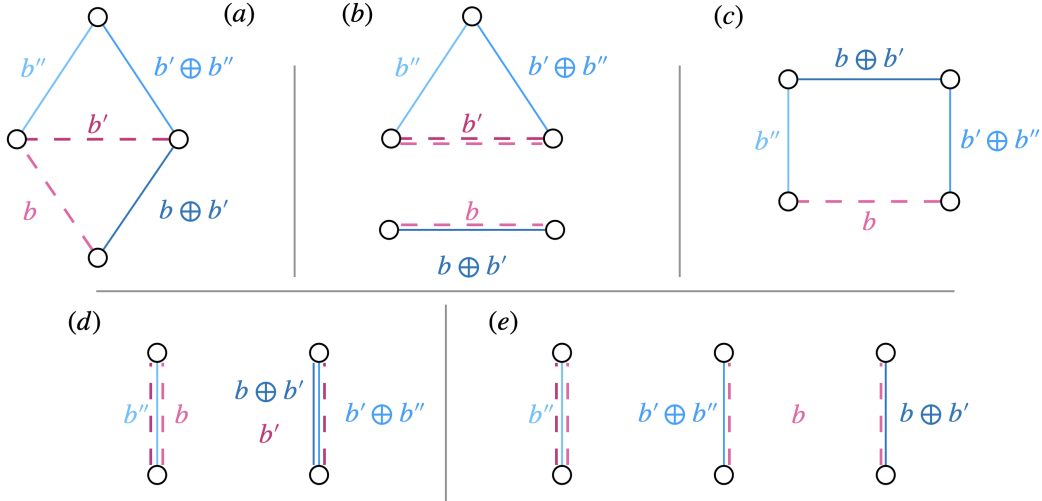


Figure 4.14: Proof of Claim 4. For clarity, we denoted G_s by s only. In blue the edges imposed by Proposition 4.1 and in pink and purple the possible candidates for a b and b' respectively. Dash lines mean that they are not necessarily corresponding to an edge in G .

From Fig. 4.14, it is easy to see which structure will appear in the computation of the energy coefficient $E_0^{(4)} = \langle E_0^n | H_1 | E_0^{(3)} \rangle$. Indeed, the composition by $\langle E_0^n | H_1 |$ forces G_b to be exactly an edge. So only situations (a), (c) and (d) are valid.

Discussion: Now that we have obtained the first three orders for the ground state and the first four orders for the energy, let's analyze the implications. While I would not put my finger on the accuracy of the prefactor of each term, the structural insights derived from the analysis remain interesting. Regarding the anti-crossing phenomenon, can we discern its presence in this derivation? At a certain time s^*

when an AC occurs, there is a sharp change in the amplitude of the instantaneous ground-state. Examining the first few coefficients, it becomes evident that distinct features of the input graph manifest. The abrupt change in amplitude might be linked to specific combinations of structures within the input graph. In the case of an AC, one can intuit that the information about the graph structure available to the quantum state up to s^* seems inadequate for guiding it toward the solution; instead, it appears to be misled and localized in an incorrect state. Is it worth exploring the idea that substantial structures within a graph could induce this effect? To address this, a more in-depth exploration of the perturbative coefficient derivation would be necessary.

Can we still infer some intriguing aspect of the quantum evolution with these first few coefficients? First, on the ground-state coefficients, let see how the first-order changes the initial state. Let us write it in the computational basis :

$$\begin{aligned}
\sum_{G_b \equiv \text{edge}} |E_b\rangle &= \sum_{G_b \equiv \text{edge}} \frac{1}{\sqrt{2^n}} \sum_{x \in \{0,1\}^n} (-1)^{b \cdot x} |x\rangle \\
&= \frac{1}{\sqrt{2^n}} \sum_{x \in \{0,1\}^n} |x\rangle \sum_{G_b \equiv \text{edge}} (-1)^{b \cdot x} \\
&= \frac{1}{\sqrt{2^n}} \sum_{x \in \{0,1\}^n} |x\rangle \sum_{\langle a,b \rangle \in E(G)} (-1)^{x_a + x_b} \\
&= \frac{1}{\sqrt{2^n}} \sum_{x \in \{0,1\}^n} (|\{\text{uncut edges in } x\}| - |\{\text{cut edges in } x\}|) |x\rangle \\
&= \frac{1}{\sqrt{2^n}} \sum_{x \in \{0,1\}^n} (|E(G)| + 2C(x)) |x\rangle
\end{aligned}$$

where $C(x)$ is the classical cost function of MaxCut (see Sec. 2.2.2) which counts (minus) the number edges across the bipartition of x . We observe that this term has little effect on a majority of states as $C(x)$ has the shape of a Gaussian around $\langle H_1 \rangle_0 = -\frac{|E(G)|}{2}$. It is acting non-trivially on states close to the best and the worst solution for MaxCut, but with opposite signs. It is less obvious how the other terms affect the different amplitudes but on the bad solutions (e.g. with lots of zeros) it is always a great influence with a plus sign.

Now, on the energy, we see that the k^{th} coefficient scales like $\mathcal{O}(|E(G)|^{k-1})$. Even though it is not exactly the same perturbative expansion as previously used in Sec. 4.1, this observation gives less credits to the validity of the first-order expansion in Sec. 4.2.4. However, it gives us really interesting features on how the quantum evolution works when solving the MaxCut problem. Apparently it treats the edges, then the triangles, then the squares and so on. Also, even cycles and odd cycles have opposite effect on the energy value which is rather expected since odd cycles are not bipartite. We can see that the different size of cycles will appear in the expansion and affecting differently the ground-state energy. It is as if the quantum state is more influenced by small cycles first. Intuitively, AQC is solving MaxCut locally

first, with short range structures and little by little starts to see larger structures. In the next part of this thesis, we actually investigate the short constant time Quantum Annealing regime to see if we can prove any analytical bound on the approximation ratio at such runtimes.

Part II

Approximation ratio in short-time Quantum Annealing

Direct Lieb-Robinson type approach

Contents

5.1 Locality in short-time quantum annealing	117
5.1.1 Locality and restriction to regular graph	118
5.1.2 Lieb-Robinson like bound	121
5.2 Two different problem applications	124
5.2.1 QA approximation of MaxCut	124
5.2.2 QA approximation of MIS	125
5.3 Toward a better bound for MaxCut?	127
5.3.1 (Non) Tightness of our LR bound	128
5.3.2 Hints for better approximation ratios	129

In this chapter, the goal is to prove a non-trivial approximation ratio, i.e. above random guess, using a quantum annealing process in constant time for MaxCut over cubic graphs. We first generalize the construction of [Farhi *et al.* 2014] where the authors proved the approximation ratio reached by single-layer QAOA. The generalization aims to encapsulate more combinatorial problems like the Maximum Independent Set problem, but still focusing on cubic graphs. Then, driven by the notion of the Lieb-Robinson velocity to recover a notion of locality in a continuous quantum evolution, we use a LR-type bound precise enough to prove that QA reaches a 0.5933-approximation of MaxCut on cubic graphs at constant time. The best available LR bounds in the literature are not suited for this purpose as they are usually asymptotic bounds, so they perform quite badly at really short radius. It will become clearer in the following sections how precise we need the bound to be. Eventually, we provide an analysis of the bound tightness and present arguments for the optimal approximation ratio with an alternative construction.

5.1 Locality in short-time quantum annealing

Recall that to evaluate the performance of a non-adiabatic quantum evolution like QA, we use the approximation ratio, which is the ratio of the expected energy of the final state to the value of the optimal cost. In the most general setting, we follow the notations of Sec. 2.2.1. We do not specify an initial Hamiltonian H_0 for the

moment, but assume that the initial state $|\psi_0\rangle$ is a uniform product state. Without loss of generality, given a graph $G(V, E)$, we seek to find the maximum of a cost function $C(x, G)$ defined on $x \in \{0, 1\}^n$. H_1^G encodes minus this cost function such that the ground-state encodes the optimal solution x^* . We restrict ourselves to cost functions decomposed into local cost functions on nodes and edges. In the final Hamiltonian H_1^G , it resolves into local observables on nodes and edges. In the most general framework, H_1 can be written as :

$$H_1^G = - \sum_{v \in V(G)} N^{(v)} - \sum_{e \in E(G)} M^{(e)}$$

Driving our quantum system according to the Hamiltonian $H(t, G) = (1 - \frac{t}{T})H_0 + \frac{t}{T}H_1^G$ over a graph G between times $t_1 < t$ induces a unitary evolution $U_{u,t}^G$. Using this notation, the state of the quantum system after a time t is given by $|\psi^G(t)\rangle = U_{0,t}^G |\psi_0\rangle$. This evolution operator is defined by $U_{0,0}^G = I$ and respects the Schrodinger equation :

$$i \frac{\partial}{\partial t} U_{0,t}^G = H(t, G) U_{0,t}^G \quad \text{for } t \in [0, T] \quad (5.1)$$

The approximation ratio for QA to solve C is then written as :

$$\rho_C = \max_T \min_G \frac{-\langle H_1^G \rangle_{G,T}}{C_{opt}(G)}$$

where $-\langle H_1^G \rangle_{G,T} = \langle \psi^G(T) | -H_1 | \psi^G(T) \rangle$ and $C_{opt}(G) = C(x^*, G)$ is the optimal cost value. The goal is to find a lower bound of this ratio. Taking the maximum on T may seem confusing, since in the infinite regime, the ratio converges to 1 by the adiabatic theorem. However, we can either force it to be smaller than a certain constant, or leave it as it is, and this maximum will be more relevant when using the local analysis detailed in the next section. By expectation linearity, we can write :

$$-\langle H_1^G \rangle_{G,T} = \sum_{v \in V} \langle N^{(v)} \rangle_{G,T} + \sum_{e \in E} \langle M^{(e)} \rangle_{G,T} \quad (5.2)$$

We are ultimately interested in producing lower bounds for each summand in the non-adiabatic regime where T is small. From now on, we will drop the dependence on T when explicit from context.

5.1.1 Locality and restriction to regular graph

We denote by $|\psi^G\rangle$ the state produced by the QA algorithm on the input graph G . This state depends on the whole input, in particular this entails that each term in Eq. 5.2 might depend on the structure of the whole graph G . For a positive integer p , we define the ball of radius p around a subgraph X of G (see Fig. 5.1)

Definition 5.1. We call $B_p(X)$ the ball of radius p around a subgraph X of G , i.e the subgraph of G containing X and all nodes and edges situated on a path of G of length at most p (i.e., with at most p edges), with an end-point in X .

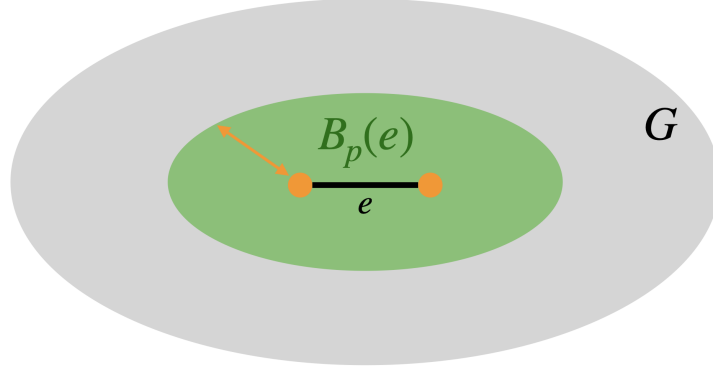


Figure 5.1: Schematic view of $B_p(e)$ (shaded green bag) in a graph G (shaded grey bag) for a subgraph X being only an edge e .

Let us assume for a moment that QA is “ p -local” for some constant p in the sense that, for each vertex and edge, it does not create correlations farther than a neighborhood at distance p . Formally, this entails that the expectation value of a local observable $N^{(v)}$ or $M^{(e)}$ only depends on the structure of the neighborhood at distance p , i.e. on $B_p(v)$ and $B_p(e)$ respectively. We point out that this notion of locality is the classical notion of local algorithm on graphs used in distributed computing, where “the output of a vertex in a local algorithm is a function of the input available within a constant-radius neighborhood of the vertex” [Suomela 2013].) In that particular setting, Eq. 5.2 can be written as:

$$-\langle H_1^G \rangle_{G,T} = \sum_{v \in V} \langle N^{(v)} \rangle_{B_p(v)} + \sum_{e \in E} \langle M^{(e)} \rangle_{B_p(e)} \quad (5.3)$$

where $|\psi^{B_p(v)}\rangle$ (resp. $|\psi^{B_p(e)}\rangle$) is the quantum state obtained by running the quantum process Hamiltonian $H(t, B_p(v))$ (resp. $H(t, B_p(e))$). With this locality assumption, we recover exactly the same argument as in the QAOA case [Farhi *et al.* 2014].

However, this locality condition for QA is of course too optimistic at this stage. We shall see in the next subsection that, thanks to Lieb-Robinson type inequalities, we can bound the difference between the actual values of energies on nodes and edges on the whole graph, and those obtained by only considering bounded radius balls around a node or an edge, hence obtaining a lower bound on the final expected value with a very similar expression to Eq. 5.3. Regular graphs have the particularity of having very few distinct small $B_p(X)$. For any node, its neighborhood at distance $p = 1$ is a star graph. As for edges, their neighborhoods at distance 1 also form a limited number of configurations, and the number of triangles in such neighborhoods will play a major role. For example, on 3-regular graphs, Fig. 5.2 shows the balls of radius 1 around a node v and an edge e .

Coming back to Eq. 5.3, all the terms in the first sum will be identical since all nodes have the same distance 1 neighborhood Ω_0 . The terms of the second sum

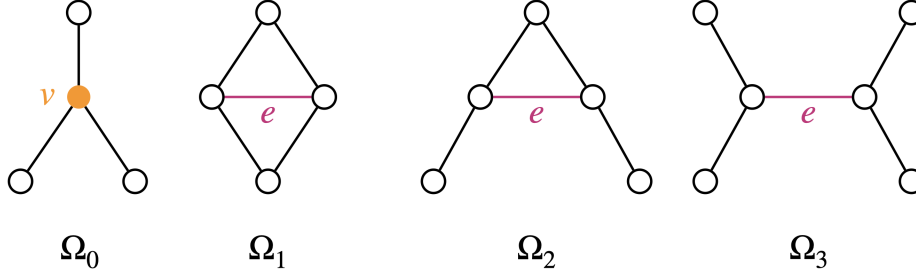


Figure 5.2: Subgraphs of 3-regular graphs for 1-qubit and 2-qubits neighborhoods at distance 1.

can be regrouped into 3 terms, corresponding to edges having Ω_1 , Ω_2 , and Ω_3 for distance 1 neighborhood. We point out the importance of starting from a product state, with the same state on every qubit otherwise, this latter gathering could not work. Thus, the expression of the final expected value becomes:

$$-\langle H_1^G \rangle_G = n \langle N^{(v)} \rangle_{\Omega_0} + g_1 \langle M^{(e)} \rangle_{\Omega_1} + g_2 \langle M^{(e)} \rangle_{\Omega_2} + g_3 \langle M^{(e)} \rangle_{\Omega_3} \quad (5.4)$$

where g_i is the number of edges e having Ω_i as $B_1(e)$, $\forall v, B_1(v) = \Omega_0$, and $\forall e, B_1(e)$ is one of the Ω_i . The indices v and e of the observables refer to the labels on the subgraphs in Fig. 5.2. We assume that, for configuration Ω_1 , the two nodes unlabeled with degree less than 3 are non-adjacent (if this happens then the four nodes form a connected component of the input graph, which is not critical for our analysis). Let's note n_i the number of configurations Ω_i in the input G . Hence there are $n_1 = g_1$ edges in configuration 1 and $4n_1 + 3n_2 = g_2$ in configuration 2 because there are four sides of Ω_1 and three edges of the triangle in Ω_2 , each of them being in a configuration of type Ω_2 . The number of edges in configuration Ω_3 corresponds to the remaining ones, i.e., $3n/2 - 5n_1 - 3n_2 = g_3$, where we used the fact that the total number of edges in a d -regular graph is $dn/2$. Therefore, the final expected energy can be written as

$$\begin{aligned} -\langle H_1^G \rangle_G &= n \langle N^{(v)} \rangle_{\Omega_0} + n_1 \langle M^{(e)} \rangle_{\Omega_1} + (4n_1 + 3n_2) \langle M^{(e)} \rangle_{\Omega_2} \\ &\quad + (3n/2 - 5n_1 - 3n_2) \langle M^{(e)} \rangle_{\Omega_3} \end{aligned} \quad (5.5)$$

This construction is the one used to analyze the performances of the QAOA for MaxCut on 3-regular graphs in [Farhi *et al.* 2014]. The authors used Eq. 5.5 together with an upper bound on the maximum cut in 3-regular graphs in order to derive a worst case lower bound of 0.6925 on the ratio achieved by single-layer QAOA. In our setting of QA, we cannot directly use such construction since the locality condition that allows us to write the final expected cost as in Eq. 5.3 does not hold. To allow a similar combinatorial argument, we thus need to recover some form of locality in the QA framework.

5.1.2 Lieb-Robinson like bound

The main tool we use in this work to regain some relaxed notion of locality in QA is a Lieb-Robinson like bound. Recall from Sec. 2.5, the LR bound is an upper limit on the velocity at which information travel in a quantum system. This bounded speed of information entails that, after evolving our system for a short amount of time, a local observation cannot depend too strongly on features lying far from the observed subsystem. In other words, considering an observable O_X localized on subsystem X , the quantity $\langle \psi^G | O_X | \psi^G \rangle$ will be close to $\langle \psi^\Omega | O_X | \psi^\Omega \rangle$ for Ω a $B_q(X)$ for a certain q (see Fig. 5.3).

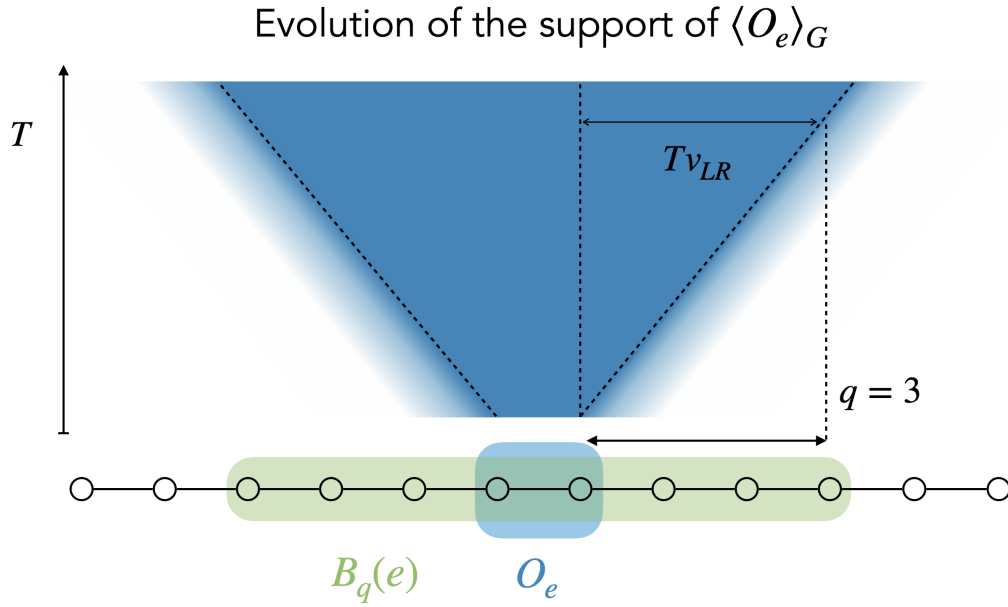


Figure 5.3: Schematic view of the evolution of the expectation value support of a local observable O_e on an edge e . v_{LR} indicates the Lieb-Robinson velocity.

More formally, we want to bound the following quantity:

$$\begin{aligned} |\langle \psi^G | O_X | \psi^G \rangle - \langle \psi^\Omega | O_X | \psi^\Omega \rangle| &= |\langle \psi_0 | (U_{0,t}^G)^\dagger O_X U_{0,t}^G | \psi_0 \rangle - \langle \psi_0 | (U_{0,t}^\Omega)^\dagger O_X U_{0,t}^\Omega | \psi_0 \rangle| \\ &= |\langle \psi_0 | [(U_{0,t}^G)^\dagger O_X U_{0,t}^G - (U_{0,t}^\Omega)^\dagger O_X U_{0,t}^\Omega] | \psi_0 \rangle| \end{aligned}$$

Since state-dependent LR bounds do not seem to exist to the best of our knowledge and are hard to derive, we first start by neglecting the initial state $|\psi_0\rangle$ of unit norm, and write:

$$|\langle \psi^G | O_X | \psi^G \rangle - \langle \psi^\Omega | O_X | \psi^\Omega \rangle| \leq \left\| (U_{0,t}^G)^\dagger O_X U_{0,t}^G - (U_{0,t}^\Omega)^\dagger O_X U_{0,t}^\Omega \right\| \quad (5.6)$$

where $\|\cdot\|$ denotes the operator norm. The two terms in the norm represent the time evolution of the observable O_X over the whole graph G and over the sub-graph Ω . Lieb-Robinson bounds are explicit, closed expressions upper bounding

this quantity [Haah *et al.* 2021], but they are too loose for our applications on approximation ratios. A numerically tractable expression (not necessarily in a closed form) is enough for our purpose. Although the expression used in this approach is not a LR bound in the strict sense of the concept, it bounds the same quantity, and for simplicity we refer to it as an LR bound. We will use the following result, adapted for time-dependent Hamiltonians from [Tran *et al.* 2019].

Proposition 5.1. *Let Ω be a subgraph of G , H_Ω be the terms of the total Hamiltonian H_G supported on the subgraph, and O_X an observable supported on X included in Ω . The total Hamiltonian H_G is a linear interpolation between H_0 and H_1 where only H_1 has interactions terms. For an evolution during T and $t \in [0, T]$, we have that:*

$$\left\| (U_{0,t}^G)^\dagger O_X U_{0,t}^G - (U_{0,t}^\Omega)^\dagger O_X U_{0,t}^\Omega \right\| \leq \int_0^t dt_1 \frac{t_1}{T} \left\| [(U_{t_1,t}^\Omega)^\dagger O_X U_{t_1,t}^\Omega, H_{1,\partial\Omega}] \right\| := LR_{O_X}^\Omega(t) \quad (5.7)$$

where $H_{1,\partial\Omega}$ is the final Hamiltonian reduced to the border of Ω . Notation $[\cdot, \cdot]$ corresponds to the commutator operation, i.e. for any operators A and B , $[A, B] = AB - BA$.

Proof. We note $U_I(t) = (U_{0,t}^\Omega)^\dagger U_{0,t}^G$, the evolution in the interaction picture of $V_I(t) = (U_{0,t}^\Omega)^\dagger V(t) U_{0,t}^\Omega$ where the perturbation V is $V(t) = H_G(t) - H_\Omega(t)$

$$\begin{aligned} \left\| (U_{0,t}^G)^\dagger O_X U_{0,t}^G - (U_{0,t}^\Omega)^\dagger O_X U_{0,t}^\Omega \right\| &= \left\| \int_0^t dt_1 \frac{d}{dt_1} \left(U_I(t_1)^\dagger (U_{0,t}^\Omega)^\dagger O_X U_{0,t}^\Omega U_I(t_1) \right) \right\| \\ &= \left\| \int_0^t dt_1 \left(U_I(t_1)^\dagger \left[(U_{0,t}^\Omega)^\dagger O_X U_{0,t}^\Omega, V_I(t_1) \right] U_I(t_1) \right) \right\| \\ &= \left\| \int_0^t dt_1 \left((U_{0,t_1}^G)^\dagger \left[(U_{t_1,t}^\Omega)^\dagger O_X U_{t_1,t}^\Omega, V(t_1) \right] U_{0,t_1}^G \right) \right\| \\ &= \left\| \int_0^t dt_1 \frac{t_1}{T} \left((U_{0,t_1}^G)^\dagger \left[(U_{t_1,t}^\Omega)^\dagger O_X U_{t_1,t}^\Omega, H_{1,\partial\Omega} \right] U_{0,t_1}^G \right) \right\| \end{aligned}$$

The last equality is true because in V we have every term in $\bar{\Omega}$, all the interactions terms between Ω and $\bar{\Omega}$ and the terms between two nodes of Ω that are not considered in Ω . The only interactions terms in $\partial\Omega$ lie in H_1 and we write it $H_{1,\partial\Omega}$. Thus, $V(t_1) = H_G(t_1) - H_\Omega(t_1) = H_{\bar{\Omega}} + \frac{t_1}{T} H_{1,\partial\Omega}$. Because $X \subset \Omega$, the left term in the commutator is strictly supported by Ω , so the term with $H_{\bar{\Omega}}$ cancels out and we are left with the last line. The factor $\frac{t_1}{T}$ comes from the linear interpolation we have assumed but we could adapt it to any other schedules. The final result uses the triangular inequality with the norm and integral and the fact that U_{0,t_1}^G is a unitary operator. $\square$

On the left-hand side of the inequality of Eq. (5.7), we have the norm of the difference between the time evolution of O_X over a graph G with the time evolution of the same observable over a subsystem Ω . This value corresponds to the largest

noticeable difference in energy one could possibly measure between the evolution on the full graph or restricted to Ω . Interestingly, the right-hand side of this inequality (i.e. the LR type bound) does not depend on the whole graph G . Potentially, only $H_{1,\partial\Omega}$ could depend on the graph but when working with regular graphs, there are only few possible choices for this term and they all depend on Ω . Therefore this LR bound only depends on the local shape Ω of the system around X . Thus, looking at the value of a local observable of an edge or a vertex on the whole graph or only locally on a subgraph is the same up to a small error given by this bound. This LR bound will quickly diverge when letting the evolution run for a large amount of time. In particular, this entails that this result can only be used to analyze the performances of very short time-continuous evolutions. Moreover, the time dependence of the bound is such that considering a larger subsystem Ω will yield a smaller bound for a fixed T . In other words, if one needs to study some evolution during a longer time (e.g. to get closer to the exact solution), one might need to increase the size of Ω . In this work, we choose to work with balls Ω of radius $p = 1$ around each vertex and edge, hence the subgraphs of Fig. 5.2, so we expect that the maximum over the running time to get the approximation ratio yields a very short T .

Now, with this result, we can lower bound the value of $\langle O_X \rangle_G$ that appears in the expression of the final expected energy in Eq. 5.2. By introducing $\langle O_X \rangle_\Omega^* = \langle O_X \rangle_\Omega - LR_{O_X}^\Omega$, we have that $\langle O_X \rangle_G \geq \langle O_X \rangle_\Omega - LR_{O_X}^\Omega$ for any observable O_X and subgraph Ω . So we can lower bound the final expected value with an expression similar to Eq. (5.5) :

$$-\langle H_1^G \rangle_{G,T} \geq n \langle N^{(v)} \rangle_{\Omega_0}^* + n_1 \langle M^{(e)} \rangle_{\Omega_1}^* + (4n_1 + 3n_2) \langle M^{(e)} \rangle_{\Omega_2}^* + (3n/2 - 5n_1 - 3n_2) \langle M^{(e)} \rangle_{\Omega_3}^* \quad (5.8)$$

The approximation ratio that we defined as

$$\rho_C = \max_T \min_G \frac{-\langle H_1^G \rangle_{G,T}}{C_{opt}(G)}$$

can be used to derive an approximation ratio of QA as a 1-local algorithm. The numerator is lower bounded by an expression that only depends on $B_1(X)$, for X being a node or an edge (see Eq. (5.8)). We also need to obtain an upper bound on the optimal cost value that can also depend on n_1 and n_2 . The minimization over all graphs G becomes a minimization over n_1 and n_2 . In a cubic graph, they must respect $4n_1 + 3n_2 \leq n$. The maximization task is ruled by the tightness of the LR-type bound we use.

In this section, we have worked on the construction developed in [Farhi *et al.* 2014] to derive an approximation ratio expression of QA as a 1-local algorithm. Unlike p -layer QAOA where $\langle O_X \rangle_G = \langle O_X \rangle_{B_p(X)}$, in QA it is only

approximately correct. The difficulty lies in the computation of the error made by this approximation. For $p = 1$, we suggest to use Proposition 5.1 as it is numerically tractable. In the next section, we apply this method to formally prove non-trivial bound on the approximation ratio reached by QA on MaxCut and MIS over cubic graphs.

5.2 Two different problem applications

In this section, we apply the method introduced above to derive approximation ratios for MaxCut and MIS on 3-regular graphs using QA metaheuristic. For each, we first recall the expression of the observable $N^{(v)}$ and $M^{(e)}$ that we introduced in Sec.2.2.1. Then we compute the corrected energy $\langle O_X \rangle_\Omega^*$ for appropriate O_X and Ω , for different values of T . Eventually, we classically solve the optimization problem $\min_{n_1, n_2} \max_T$ to derive the value of the approximation ratio.

5.2.1 QA approximation of MaxCut

In Sec. 2.2.2, we introduced the MaxCut problem as well as its encoding in a Hamiltonian. We recall that for an input graph $G(V, E)$, the final Hamiltonian is written like $H_1 = -\sum_{e \in E} O_e$ where for $e = \langle a, b \rangle$, $O_e = \frac{1}{2}(1 - \sigma_z^{(a)} \sigma_z^{(b)})$. To compute the approximation ratio with the previous developed method, we identify $N^{(v)} = 0$ and $M^{(e)} = O_e$. For the optimal cut, we use the same reasoning as in [Farhi *et al.* 2014]: the size of the maximum cut is at most the total number of edges $|E|$ and each configuration of type Ω_1 and Ω_2 (see Figure 5.2) has at least one edge that is not cut (they each contain a triangle that is disjoint from all other triangles). For cubic graph we know that $|E| = 3n/2$. Hence, we have that $C_{opt}(G) \leq 3n/2 - n_1 - n_2$. By normalizing with $n_1 \leftarrow n_1/n$ and $n_2 \leftarrow n_2/n$, we end up with a global approximation ratio ρ_{MC} for problem MaxCut that verifies:

$$\rho_{MC} \geq \max_T \min_{\substack{n_1, n_2 \text{ s.t.} \\ 4n_1 + 3n_2 \leq 1}} \frac{n_1 \langle O_e \rangle_{\Omega_1}^* + (4n_1 + 3n_2) \langle O_e \rangle_{\Omega_2}^* + (3/2 - 5n_1 - 3n_2) \langle O_e \rangle_{\Omega_3}^*}{3/2 - n_1 - n_2} \quad (5.9)$$

$$\geq \max_T \langle O_e \rangle_{\Omega_3}^* \quad (5.10)$$

where we used our technical result of Appendix C.1 which proves that under reasonable conditions, the minimization over n_1 and n_2 such that $4n_1 + 3n_2 \leq 1$ results in $n_1 = n_2 = 0$. We will see in the numerical computation that the necessary conditions are fulfilled to have this value of n_1 and n_2 . The worst case for QA is therefore configuration Ω_3 , i.e. QA struggles more on triangle-free graphs. Now, it is interesting to mention that the approximation ratio of QA for MaxCut is lower bounded by the edge energy in the middle of configuration Ω_3 (see Fig. 5.2) corrected by the LR-type bound. This configuration is bipartite, meaning that in the long time regime, $\langle O_e \rangle_{\Omega_3}$ converges toward 1. Indeed, we remind the reader that

for a classical state $|x\rangle$, $\langle x|O_e|x\rangle = 1$ if and only if edge e lies across the bipartition x and 0 otherwise. This also implies that $\rho_{MC} \geq 0.5$ because at $T = 0$, the state is a uniform superposition over all possible classical states. We know from Sec. 2.5 that the LR bound increases exponentially with T which is why $\langle O_e \rangle_{\Omega_3}^*$ reaches a maximum for a finite T .

We numerically evaluate the $\langle O_e \rangle_{\Omega_i}^*$ by computing the middle edge energy for each Ω_i and the corresponding LR-type bound of Proposition 5.7. We found that a maximum is reached at $T_{mc} = 1.62$ for which the numerical values for $\langle O_e \rangle_{\Omega_i}^*$ are presented in Table 5.1.

i	1	2	3
$\langle O_e \rangle_{\Omega_i}$	0.5951	0.6152	0.6350
$LR_{O_e}^{\Omega_i}$	0.0203	0.0310	0.0417
$\langle O_e \rangle_{\Omega_i}^*$	0.5748	0.5842	0.5933

Table 5.1: Numerical values to obtain $\langle O_e \rangle_{\Omega_i}^*$ at $T = T_{mc}$

We check that those values are compatible with the conditions of Appendix C.1, to have that $\rho_{MC} \geq \max_T \langle O_e \rangle_{\Omega_3}^* = 0.5933$.

Details on the computation: For $k \in \{1, 2, 3\}$, to compute the value of $\langle O_e \rangle_{\Omega_i}^*$ for $T \in [0, 5]$, we accessed to the value of $\langle O_e \rangle_{\Omega_i}$ and the LR-like bound $LR_{O_e}^{\Omega_i}(T)$ for appropriate O_X . $\langle O_e \rangle_{\Omega_i}$ can be computed directly from the quantum state $|\psi^{\Omega_i}(T)\rangle$. This state is computed by numerically solving the Schrödinger equation (2.1) with a relative tolerance of 10^{-9} on Ω_i . These computations were performed using the AnalogQPU simulator on Eviden Qaptiva using boost. To compute the LR bound, we have to compute the unitary evolution operator $U_{t_1, T}^{\Omega_i}$ for all $t_1 \in [0, T]$ and compute the integral on the right hand side of Eq. 5.7. We chose a time step of 10^{-3} to compute the integral. We compute the unitary operator for each time step by solving the Schrödinger equation for evolution operator (Eq. 5.1). These computations were carried using the Differential Equations Julia library. Finally, we chose the interaction terms $H_{1, \partial\Omega}$ to be as large as possible for each Ω_i . This is a sum over all interactions $M^{(e)}$ for edges e which are not in Ω but have at least one endpoint in Ω (see Fig. 5.4).

Here we proved that QA as a one local algorithm achieves a 0.5933 approximation of MaxCut over cubic graphs. The analysis further showed that triangles seems to help the quantum evolution to reach better ratio. In Chapter 7 we investigate more in depth this intuition to analyse the behaviors of larger double binary.

5.2.2 QA approximation of MIS

In Sec. 2.2.3, we introduced the Maximum Independent Set problem with its encoding in a Hamiltonian. We recall that for an input graph $G(V, E)$, the final Hamiltonian can be written like $H_1 = -\sum_{v \in V} R_v - \sum_{e \in E} Q_e$ where for a node v ,

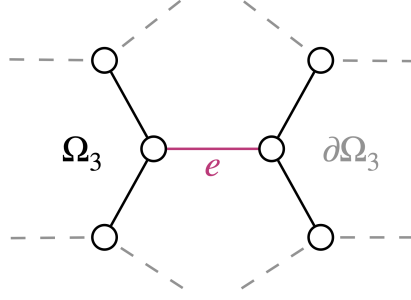


Figure 5.4: Example of $\partial\Omega$ for Ω_3 configuration maximizing the number of interactions with the rest of the graph. Note that an edge between two nodes would also belong to $\partial\Omega$.

$R_v = \frac{1}{2}(1 - \sigma_z^{(v)})$ and for an edge $e = \langle a, b \rangle$, $Q_e = -\frac{1}{4}(1 - \sigma_z^{(a)})(1 - \sigma_z^{(b)})$. Unlike Max-Cut, we see here that we have a component on the nodes and a component on the edges. We can identify the observables to apply the method of Sec. 5.1 $N^{(v)} = R_v$ and $M^{(e)} = Q_e$. These two components have opposite effects on the cost value of a solution x , R_v tends to increase the size and Q_e tends to minimize the number of edges in G_x . G_x is the graph induced by nodes v such that $x_v = 1$. Even if the measure of the final state of an annealing process can be a state for which G_x is not an IS, it is easy to correct it to have an IS of the same energy (see Sec. 2.2.3). The energy of x is the size of the IS produced from x . Now, we need to upper bound the size of an MIS in regular graph, we use the following claim :

Claim 5. For a d -regular graph, the size of the MIS is upper bounded by $n/2$ and this value is reached by bipartite graphs.

Proof. The proof is straightforward once we see that the total number of edges in a d -regular graph is $\frac{dn}{2}$, and any independent set I has $d|I|$ edges incident to it. $\square$

Along with Eq. (5.8), and the fact that $C_{opt}(G) \leq n/2$, the approximation ratio for MIS like the following, where we employed the same normalization as in the previous section with $n_1 \leftarrow n_1/n$ and $n_2 \leftarrow n_2/n$:

$$\rho_{MIS} \geq \max_T \min_{\substack{n_1, n_2 \text{ s.t.} \\ 4n_1 + 3n_2 \leq 1}} 2 \left[\langle R_v \rangle_{\Omega_0}^* + n_1 \langle Q_e \rangle_{\Omega_1}^* + (4n_1 + 3n_2) \langle Q_e \rangle_{\Omega_2}^* \right] \quad (5.11)$$

$$\begin{aligned} & + (3/2 - 5n_1 - 3n_2) \langle Q_e \rangle_{\Omega_3}^* \\ & \geq \max_T (2 \langle R_v \rangle_{\Omega_0}^* + 3 \langle Q_e \rangle_{\Omega_3}^*) \end{aligned} \quad (5.12)$$

where the last inequality comes from the optimization over n_1 and n_2 giving $n_1 = n_2 = 0$ (see Appendix C.1). Similarly to MaxCut case, triangle-free graphs seem to be worst cases for QA. At $T = 0$, we have the random guess. For a classical state x , we know that $\langle x | R_v | x \rangle = 1$ if and only if $x_v = 1$ and 0 otherwise. Also, for an edge $e = \langle a, b \rangle$, $\langle x | Q_e | x \rangle = -1$ if and only if $x_a = x_b = 1$ and 0 otherwise. So the random

guess has an approximation ratio of $2 \times \frac{1}{2} - 3 \times \frac{1}{4} = 0.25$. The maximization over T gives at time $T_{mis} = 1.32$, the numerical values for $\langle O_X \rangle_{\Omega_i}^*$ presented in Table 5.2.

i	0	1	2	3
$\langle O_X \rangle_{\Omega_i}$	0.4653	-0.1939	-0.1920	-0.1901
$LR_{O_X}^{\Omega_i}$	0.0072	0.0034	0.0065	0.0096
$\langle O_X \rangle_{\Omega_i}^*$	0.4581	-0.1972	-0.1985	-0.1997

Table 5.2: Numerical values to obtain $\langle O_X \rangle_{\Omega_i}^*$ at time $T = T_{mis}$ for the corresponding O_X .

From the numerical values, we can compute the approximation ratio reached by QA on MIS over cubic graphs, i.e. $\rho_{MIS} \geq 0.3171 > 0.25$. This proves that constant time QA is better than random guess for solving MIS. By random guess for MIS, we mean the algorithm that chooses uniformly at random in $\{0, 1\}^n$, which also corresponds to QA at $T = 0$. However, to put it into perspective, a naive greedy algorithm that runs in linear time achieves a 0.5 approximation. Let I be your candidate independent set. Then, for each node, if none of its neighbors is in I yet, put the node in I . On a d -regular graphs, it outputs an independent set of size n/d . Our purpose is to illustrate the generality of the approach based on Lieb-Robinson bounds, which can be used for estimating the energy both on nodes and on edges. We reach with a constant-time local algorithm a ratio that is better than random guess.

In this section, we applied the approach presented in Sec. 5.1 on two combinatorial problems, MaxCut and MIS. The computations showed that constant time QA in both cases is better than random guess. Although it was expected to beat random guess, these numbers provides the first guarantee for short time QA. The numerical values are still not of profound interest in comparison to already known algorithms (see Sec. 2.3). In the next section, motivated by a tightness analysis of the bound and numerical argument, we suggest an alternative approach, which is at this stage not practically tractable.

5.3 Toward a better bound for MaxCut?

We believe that the algorithm behaves significantly better than the approximation bounds that we were able to prove. In this section, we first argue this claim with a numerical analysis of our LR bound tightness on MaxCut suggesting some improvement direction. Then, in order to get the best of QA, we provide some hint for a new approach to find a tighter bound on the approximation ratio and conjecture that the actual approximation ratio for MaxCut in cubic graphs with a distance 1 analysis might be of 0.6963, while we only guarantee 0.59.

5.3.1 (Non) Tightness of our LR bound

The analysis of the LR bound tightness or non-tightness depends on the runtime T and on the radius p of the ball. The construction we followed to obtain our ratios is for $p = 1$. So, we study the dependency in T of the LR-type bound we used in the case of MaxCut. We will focus on the bound for Ω_3 as it is the most relevant one for deriving the ratios. For this purpose, we construct an example of a ball maximizing the $\partial\Omega_3$ as in the case when we compute the LR-type bound from Eq. (5.7). In Fig. 5.4, we see that Ω_3 can have up to eight interactions with the rest of the graph. We want to create the “worst” neighborhood W_3 for Ω_3 so that $\delta_1(W_3) = |\langle O_e \rangle_{\Omega_3} - \langle O_e \rangle_{W_3}|$ is maximal. A quick numerical analysis tells us that interactions between two nodes of Ω_3 increase the value of the final expected energy, while an interaction with a node outside Ω_3 decreases it. Here, to compare to the LR-type bound, as we said, we need eight interactions with the outside of Ω_3 . They will tend to decrease the value of the final energy. So in other word, this “worst” configuration we are looking for should push the energy value toward 0 as fast as possible as the edge energy in Ω_3 converges toward 1 progressively. Therefore, W_3 is a configuration in which the middle edge is necessarily uncut in the optimal state x^* . This happens when an edge belongs to two odd cycles. Base on the intuition that small cycles have effect on the edge energy faster than large ones, we suggest W_3 of Fig. 5.5 (left). In this configuration, all edges but e are cut in the optimal solution. In W_3 , $\langle O_e \rangle_{W_3}$ reaches a maximum at time $T^* = 3.15$.

From the derivation of the LR bound we used, i.e. Eq. (5.7), there are two potential steps where we are losing information. The first one is when we neglect the initial state $|\psi_0\rangle$ in Eq. 5.6, i.e. between $\delta_1(G) = |\langle \psi^G | O_e | \psi^G \rangle - \langle \psi^{\Omega_3} | O_e | \psi^{\Omega_3} \rangle|$ and $\delta_2(G) = \|(U_{0,t}^G)^\dagger O_X U_{0,t}^G - (U_{0,t}^{\Omega_3})^\dagger O_X U_{0,t}^{\Omega_3}\|$ where e is an edge in G . The second one is when proving Proposition 5.1 we use the triangular inequality with the operator norm and the integral, i.e., between $\delta_2(G)$ and $LR_{O_X}^{\Omega_3}$. To illustrate this, we plot in Fig. 5.5 these three quantities $\delta_1(G) \leq \delta_2(G) \leq LR_{O_X}^{\Omega_3}$ for the non-trivial ball W_3 we detailed above.

The LR bound used to derive the approximation ratio (i.e. the top red line in the graph) is interesting up to T_{mc} . It then diverges too quickly away from δ_1 (and so does the slightly tighter bound δ_2). This conveys the intuition that the bound becomes weak because the initial state is ignored. To the best of our knowledge, state-dependent Lieb-Robinson bounds do not exist in the literature as they are difficult to derive. In Chapter 7, we give more intuition on this improvement.

On proving better approximation ratio, this direct approach seems to be limited by the fact it is too complicated to derive state-dependent LR bounds. To circumvent this obstacle, in the next section we suggest another approach to get the best of the LR bound for the derivation of approximation ratios.

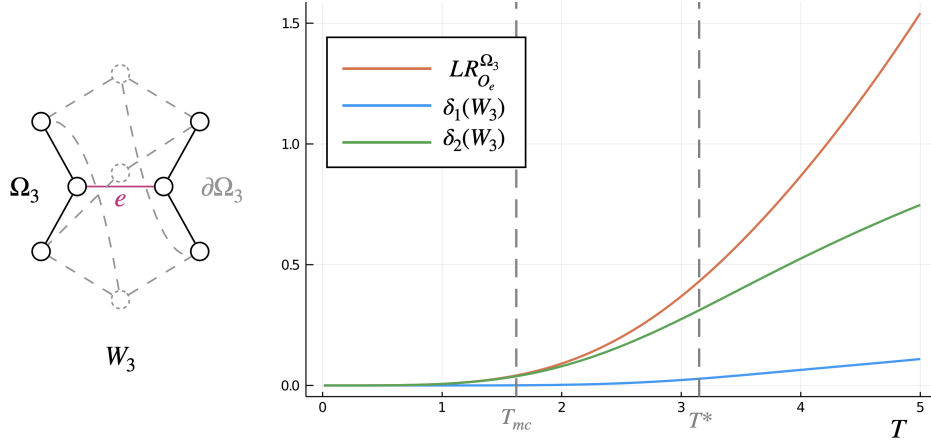


Figure 5.5: On the left, W_3 , worst edge configuration for LR bound centered on e . On the right, δ_1 and δ_2 for W_3 aside with $LR_{O_e}^{\Omega_3}$. The dashed lines indicate the time $T_{mc} = 1.62$ for which we obtain the value of 0.59 and $T^* = 3.15$.

5.3.2 Hints for better approximation ratios

To derive a better approximation ratio for a distance 1 analysis of QA applied to MaxCut, i.e. still using Eq. (5.8), we need to grasp the meaning of the star energies $\langle O_e \rangle_{\Omega}^*$. When we substitute the LR bound to the edge energy $\langle O_e \rangle_{\Omega}$, we are actually handling any “bad” trajectory of an edge energy in any possible configuration. “Bad” in the sense that the energy is evolving away from the trajectory of $\langle O_e \rangle_{\Omega}$ caused by the influence of graph structure at distance more than 1. So ideally we should use energies $\langle O_e \rangle_{\mathcal{G}}^{\Omega}$, corresponding to the minimum energy of the edge e among all possible cubic graphs $\mathcal{G}$ such that $B_1(e) = \Omega$:

$$\langle O_e \rangle_{\mathcal{G}}^{\Omega} = \min\{\langle O_e \rangle_G \mid G \in \mathcal{G}, e \in E(G) \text{ s.t. } B_1(e) = \Omega\}$$

That being said, this minimum is obviously intractable as the search space is infinite with infinitely large graphs. However, for short enough runtime, we can restrict ourselves to small configuration around the edge e , extending the Ω_i . Intuitively, the minimum is reached by edges that are necessarily uncut in the optimal solution. Thus, our strategy is to construct (small) balls W_i for each configuration Ω_i such that the energy of the observed edge quickly goes to 0 after its expected initial rise, similarly to the construction for the tightness analysis (see Sec. 5.3.1). In order to accelerate the convergence speed to 0, we enforced this condition via the adjunction of the smallest possible odd cycles. Indeed, as we already hinted in Sec. 2.5 and 4.3, small cycles converge faster to their target because they are considered first by the quantum evolution. The proposed balls W_i for each configuration Ω_i are shown in Fig. 5.6. We computed the energy of the edge e in each of these balls for $T \in [0, 5]$ and used Eq. 5.9, giving us a ratio of 0.6963 at $T^* = 3.15$. As mentioned above, this is just a numerical estimate and by no mean a formal derivation. However, unless

some very non-monotonous behavior arises when growing the radius of these balls, this is most likely an accurate estimate.

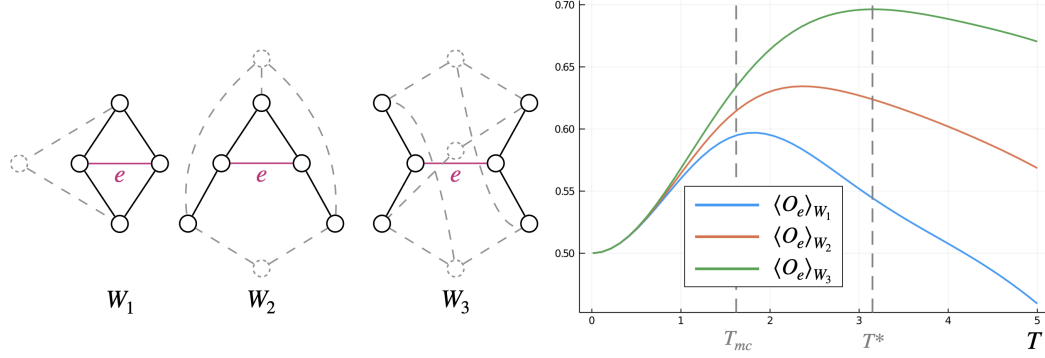


Figure 5.6: On the left, worst configurations W_i for an edge e that has $B_1(e) = \Omega_i$. On the right, evolution of the edge energy against T , $\langle O_e \rangle_{W_1}$ (blue, bottom), $\langle O_e \rangle_{W_2}$ (red, middle) and $\langle O_e \rangle_{W_3}$ (green, top).

This estimation is a truly local (distance 1) analysis of QA as the expression of the approximation we use to compute the ratio depends only on balls of radius 1. It thus seems reasonable to compare this ratio to one obtained for other truly local algorithms. The other known bounds in this setting are single layer QAOA (0.6925) [Farhi *et al.* 2014] and Hastings' (classical) local algorithm (0.6980) [Hastings 2019] (see Sec. 2.3 for more details). If our bound is verified, this would show that a QA in time $T^* = 3.15$ outperforms single layer QAOA.

As we will develop more formally this construction in Chapter 6, we do not give too much details, but based on already known LR bounds, it seems that one would need to look for any balls of radius up to $q \simeq 70$ in order to prove a relevant ratio. The best known LR bound for d -regular graphs is presented in [Chen *et al.* 2023]:

$$|\langle O_e \rangle_G - \langle O_e \rangle_{B_q(e)}| \leq \frac{(4(d-1)t)^q}{q!} \frac{1}{1-2/e}$$

which achieves a value of 0.014 for $t = T^* = 3.15$ and $q = 71$. Unfortunately, this radius is unreachable for the best known classical simulation algorithm of quantum annealing, and neither is it for the largest existing quantum computer. The larger ball $B_q(e)$ that can be completed in a cubic graph has $2(2^{q+1} - 1)$ nodes. To be able to use this approach one would need to reach a similar bound value for $q = 3$.

Discussion: The work presented in this chapter can be easily adapted to tackle other bounded degree graphs, e.g. MaxCut over 4 (resp 5)-regular graphs gives a ratio of 0.57 (resp. 0.56), with shorter annealing times. Secondly, we draw readers' attention to the fact that we only considered linear schedule of the form $s(t) = t/T$. Other types of schedules should be tested to get the best possible ratio. For example,

we tested with the optimal schedule of Grover [Roland & Cerf 2002], i.e. with

$$f(s) = \frac{1}{2} + \frac{2s - 1}{2\sqrt{2^n - (2^n - 1)(2s - 1)^2}}$$

and the total Hamiltonian $H(s) = (1 - f(s))H_0 + f(s)H_1$. It is also possible to add a perturbative Hamiltonian so the total Hamiltonian is $H(s) = (1 - s)H_0 + sH_1 + s(1 - s)H_{pert}$ where $H_{pert} = \frac{[H_0, H_1]}{2}$. Both initiatives output a ratio of 0.61, slightly improving on the bound presented in Sec. 5.2.1. In any case, the value of the edge energy as well as the LR-type bound value have to be computed again.

Conclusion

The general method described in this chapter provides tools for deriving approximation ratios for quantum annealing on combinatorial problems, with a very short runtime. The locality of quantum annealing given by the Lieb-Robinson bound allows combinatorial arguments on regular (and in particular cubic) graphs, of the same type as those used for QAOA. We emphasized that the theoretical arguments are valid for regular graphs of any fixed degree. We applied this technique to MaxCut and MIS to show how it can be used to access numerical values of approximation ratios for cubic graphs. We then discussed the non-tightness of the Lieb-Robinson bound using MaxCut and drew some “worst-case” graphs to argue that QA can probably achieve an approximation ratio for this problem significantly better than we can actually prove. Overall, we have developed tools to recover locality in the standard formulation of quantum annealing and use this relaxed locality to derive lower bounds on QA’s performance for the Maximum Cut and the Maximum Independent Set problem on cubic graphs.

There are two main directions for improving these results and the general issue of short-term QA as an approximation algorithm for graph optimization problems. The first direction would be to improve the algorithm itself by choosing a better schedule. In our analysis, we focused on a linear interpolation scheme, whereas QAOA, which can be considered a special case of QA, relies on a parameterized bang-bang scheme. There is, however, numerical evidence to suggest that some other schedules produce better energies (see [Brady *et al.* 2021, Guéry-Odelin *et al.* 2019]). We have proposed two types of schedule modification that give slightly better ratios. We believe that our tools would enable us to study this wider variety of QAs and obtain analytical guarantees of their performance.

The other direction would be to refine the analysis of constant-time QA using less restrictive locality arguments such as the one we introduced informally in Sec. 5.3.2. In the next chapter, we choose to formally apply this alternative method.

Tight Lieb-Robinson bound for approximation ratio

Contents

6.1	Local analysis of QA and Lieb-Robinson Bound	134
6.1.1	Parametrized QA	134
6.1.2	p -local analysis	135
6.1.3	Lieb-Robinson bound reduction	137
6.2	Tight LR bound on regular graphs for MaxCut approximation	139
6.2.1	Tight LR bound on regular graphs in QA	139
6.2.2	Global LR bound	144
6.2.3	Application to approximation ratio of MaxCut	146
6.3	Discussion	149
6.3.1	Tightness of the analysis method	149
6.3.2	Toward better ratios	151
6.3.3	Directions for generalizing the construction	152

In this chapter, we pursue the alternative construction suggested in the previous chapter (Sec. 5.3.2) to prove the following theorem.

Theorem. *With a 1-local analysis, the approximation ratio reached by QA to solve MaxCut over cubic graphs is above 0.7020.*

This result is to be compared with the previous one obtained in Chapter 5, where we proved a lower bound of 0.5933. We manage to increase the numerical value using a slightly different approach. First we formally define a p -local analysis of QA and introduce a new parameter, α , in the standard QA setting. Then we explain how a Lieb-Robinson bound helps to tract the minimization over all possible cubic graphs by reducing the total space search $\mathcal{G}$ to the family of radius q balls $\mathcal{B}_q$. To find the minimum among all those balls, we filter out most balls with a global LR bound ε and to get the best possible ratio, we employ a local bound ε_{loc} , adapted to the considered ball. Eventually, an optimization over T and α allows us to prove the theorem. An overview of the different steps of the proof are sum-up in Fig. 6.1.

We first go through the formal construction of the method that we introduced in Sec. 5.3.2 and we detail where the parameter α affect the annealing process. We

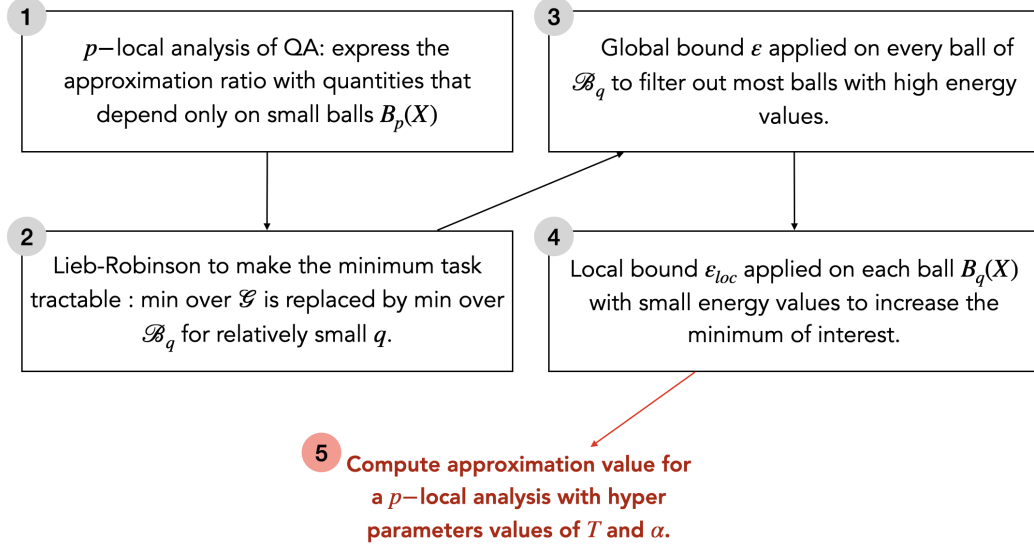


Figure 6.1: Overview of the different steps of the Theorem proof.

focus on the MaxCut problem only but as we developed in Sec. 5.1, it is possible to consider other combinatorial problems. Then, we derive our tight LR bound that allows us to compute the approximation ratio promised by the theorem.

6.1 Local analysis of QA and Lieb-Robinson Bound

In this section, we formally develop the construction hinted in Sec. 5.3.2. More exactly, we introduce a parametrized version of QA and write down the associated approximation ratio. Then we define properly the meaning of a p -local analysis of QA with example for $p = 0$ and $p = 1$. Eventually, we show how the Lieb-Robinson bound can be used to tract the desired approximation ratio.

6.1.1 Parametrized QA

Recall that for a given graph $G(V, E)$ in $\mathcal{G}$, a family of graphs, on which we want to solve MaxCut, the target Hamiltonian is $H_1^G = -\sum_{X \in E} O_X$ where $O_X = \frac{1-\sigma_z^{(a)}\sigma_z^{(b)}}{2}$ for the edge $X = \langle a, b \rangle$. The difference of this new setting is the introduction of a hyper-parameter α in the initial Hamiltonian as

$$H_0(\alpha) = -\sum_{v \in V} \frac{\sigma_x^{(v)}}{\alpha}$$

The initial state is still the uniform superposition $|\psi_0\rangle$, as for the final state, it is now depending on α , i.e. $|\psi^G(T, \alpha)\rangle = U_{T, \alpha}^G |\psi_0\rangle$. Here, $U_{T, \alpha}^G$ denotes the unitary evolution operator under the time-dependent Hamiltonian $\bar{H}(t, G) = (1 - \frac{t}{T})H_0(\alpha) + \frac{t}{T}H_1^G$ and T the total annealing time. This operator $U_{T, \alpha}^G$, corresponds

to a Schrödinger evolution, i.e. is a solution of:

$$\forall t \in [0, T], \quad i\hbar \frac{dU_{t,\alpha}^G}{dt} = H(t, G)U_{t,\alpha}^G$$

The expected value at the end of the annealing process is $\langle H_1^G \rangle_{G,\alpha} = \langle \psi^G(T, \alpha) | H_1^G | \psi^G(T, \alpha) \rangle$ and thus, our metric of interest is

$$\rho_{MC} = \max_{T,\alpha} \min_{G \in \mathcal{G}} \frac{-\langle H_1^G \rangle_{G,\alpha}}{C_{opt}(G)}.$$

Since we are interested in using local arguments to bound this quantity, as in Chapter 5, we will restrict the family of graphs $\mathcal{G}$ to the set of 3-regular graphs. The goal is to find a good lower bound for this ratio. This can be achieved by separately upper bounding the denominator and lower bounding the numerator. By linearity, the numerator can be written as a sum over the edges $\sum_{X \in E} \langle O_X \rangle_{G,\alpha}$.

Influence of α : To give a brief idea on the effect of the parameter on the quantum evolution, let us plot in Fig. 6.2, the edge energy value of the middle edge of W_3 that we introduced in Fig. 5.5 for different values of α . We know that in the standard QA, i.e. for $\alpha = 1$ (green curve), it starts at 0.5 at $T = 0$, it increases up to 0.6963 at time $T^* = 3.15$, and then decreases toward 0 its final value. Here, to some extent, α modulates the weight of H_0 compared to H_1 . As it will become clearer when deriving the LR bound in Sec. 6.2, increasing α slows down the evolution. Informally, the information travels at smaller speed in the quantum system. By consequence, for $\alpha > 1$, it takes more time to reach a similar value and it corrects its trajectory toward the target value later in time. In some sense, QA is “fooled” for longer times when α is large. This allows the edge energy to reach a higher maximum later.

From now on, we will drop the dependency in α in the edge energy notation, like for T , when it is clear from context. It will only appear when optimizing the approximation ratio.

6.1.2 p -local analysis

As we mention, due to its analog nature, QA is non local, so to compare it to other optimization solver like QAOA or to local classical algorithms, we need to define a way to analyze it as a local algorithm. In Chapter 5, we suggest an approach for a 1-local analysis. Here we define the notion for any p .

Definition 6.1 (p -local analysis of QA). *We call a p -local analysis of QA, an analysis that produces an approximation ratio that depends only on balls in $\mathcal{B}_p$. In other words, the expression used to compute the ratio only considered balls in $\mathcal{B}_p$.*

In Sec. 5.1, we expanded on one possibility to analyze QA at distance 1, but it was rather a restrictive version. Let us develop the construction of a 0-local and

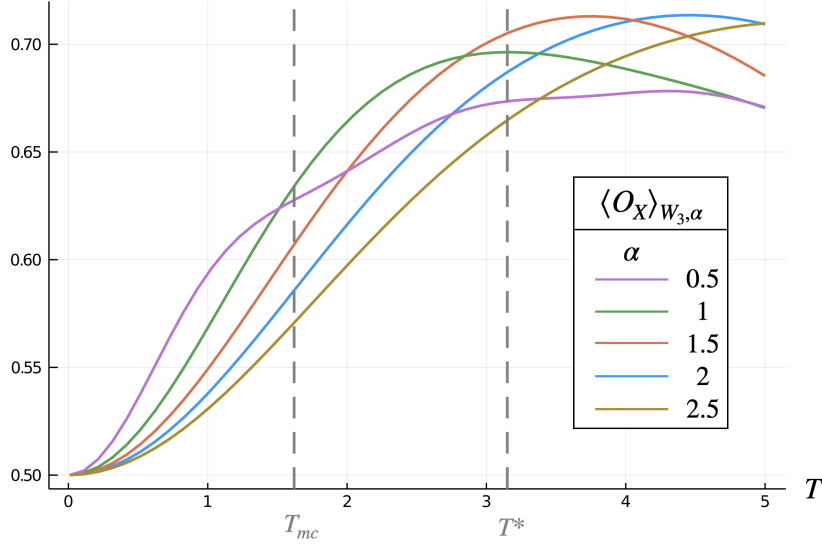


Figure 6.2: Edge energy in the middle of W_3 graph $\langle O_X \rangle_{W_3, \alpha}$ for $\alpha \in [0.5, 1, 1.5, 2, 2.5]$

1-local analysis in the more general case.

0-local: For any $X \in E$, a trivial lower bound of $\langle O_X \rangle_G$ is to assume that all edges are in the worst possible configuration. It means that it is lower bounded by $\min_{G \in \mathcal{G}} \langle O_X \rangle_G = \langle O_X \rangle_{\mathcal{G}}$, where we note $\mathcal{G}$ the set of all cubic graphs. With the trivial bound on $C_{opt}(G) < |E|$, it gives the following lower bound on the approximation ratio:

$$\rho_{MC} \geq \max_{T, \alpha} \langle O_X \rangle_{\mathcal{G}}.$$

Finding the latter value constitutes the approximation ratio of QA with 0-local analysis.

1-local: Now, we are allowed to use the knowledge at distance 1 from the edge X . Like in Sec. 5.1, for cubic graphs there are only three possible $B_1(X)$. We are referring to the Ω_i of Fig. 5.2. Therefore the edge energies $\langle O_X \rangle_G$ can be bounded by one of the following quantities $\langle O_X \rangle_{\mathcal{G}}^{\Omega_i} = \min\{\langle O_X \rangle_G \mid G \in \mathcal{G}, X \in E(G) \text{ s.t. } B_1(X) = \Omega_i\}$. And with a similar combinatorial argument, n_i being the number of Ω_i in G , the final expected energy can thus be lower bounded as:

$$-\langle H_1^G \rangle_G \geq n_1 \langle O_X \rangle_{\mathcal{G}}^{\Omega_1} + (4n_1 + 3n_2) \langle O_X \rangle_{\mathcal{G}}^{\Omega_2} + (|E| - 5n_1 - 3n_2) \langle O_X \rangle_{\mathcal{G}}^{\Omega_3}$$

We remind that this expression still depends on the input graph through the variables n_i , and thus, still needs to be minimized over the positive integers n_i 's with the constraint that $4n_1 + 3n_2 \leq |V|$. We also use the same upper bound on the optimal value, i.e. $C_{opt} \leq |E| - n_1 - n_2$, where $|E| = 3|V|/2$. We recover the lower

bound for the approximation ratio:

$$\rho_{MC} \geq \max_{T, \alpha} \min_{\substack{n_1, n_2 \text{ st} \\ 4n_1 + 3n_2 \leq |V|}} \frac{n_1 \langle O_X \rangle_{\mathcal{G}}^{\Omega_1} + (4n_1 + 3n_2) \langle O_X \rangle_{\mathcal{G}}^{\Omega_2} + (|E| - 5n_1 - 3n_2) \langle O_X \rangle_{\mathcal{G}}^{\Omega_3}}{|E| - n_1 - n_2} \quad (6.1)$$

The result from Appendix C.1 still applies and under reasonable conditions the minimization gives $n_1 = n_2 = 0$ so the approximation ratio becomes

$$\rho_{MC} \geq \max_{T, \alpha} \langle O_X \rangle_{\mathcal{G}}^{\Omega_3}$$

We see that this lower bound is very similar to the 0-local analysis case where we added the fact that edge X must have Ω_3 as its $B_1(X)$. We are really using more knowledge of the neighboring structure of X .

p-local: For a larger p , one would need to enumerate every possible $B_p(X)$ balls that can be completed in a cubic graphs and comes up with a linear combination of the edge energy in each of these $B_p(X)$. In [Wurtz & Love 2021], the authors undertook this approach to prove that $p = 2$ QAOA achieves a ratio of 0.7559, value conjectured originally in [Farhi et al. 2014]. It appears that the double binary tree T_p of depth p is also the worst case for $p = 2$ and it is widely believed in the community that T_p is a great candidate to be the ball $B_p(X)$ such that :

$$\rho_{MC} \geq \max_{T, \alpha} \langle O_X \rangle_{\mathcal{G}}^{T_p}$$

Motivated by this conjectured, the authors of [Basso et al. 2022] cleverly develop an iterative procedure to access the final energy of an edge in the middle of a T_p with a p layer QAOA. They were able to compute the exact value up to $p = 17$. No similar results exist for QA but in Chapter 7 we suggest an idea to derive this value.

With this in mind, to access to the approximation ratio reached by QA, we still need to compute the minimum over an infinite family of graphs. This minimum is obviously intractable. So to overcome this difficulty, as we mentioned at the end of Chapter 5, we will use the Lieb-Robinson bound presented in the next section.

6.1.3 Lieb-Robinson bound reduction

This idea of this section is to introduce two types of LR bounds that will help us in the computation of the minimization over all the cubic graphs. We already know that with a finite short runtime, the trajectory of an edge energy mostly depends on the neighboring structure. Consequently, when computing the minimum of interest, we can stop at a finite radius of the graph around the edge, with a penalty ε on the minimum. In other words, for a given time T and a desired error ε , there exists a $q(T, \varepsilon)$ such that $|\langle O_X \rangle_{\mathcal{G}}^{\Omega} - \langle O_X \rangle_{\mathcal{B}_q(T, \varepsilon)}^{\Omega}| \leq \varepsilon$ where $\mathcal{B}_q$ is the set of all possible $B_q(X)$ when X is an edge of a cubic graphs $G \in \mathcal{G}$. This is basically the main idea of the Lieb-Robinson velocity. Now, to get the best of the LR bound, we will describe a local bound, that is adapted to the considered ball $B_q(X)$ and a global one that is maximizing the local bounds.

Local bound. Suppose that for any graph G , and any edge X of G , there exists a $\varepsilon_{loc}(B_q(X), T, \alpha) > 0$ that upper bounds the absolute difference: $|\langle O_X \rangle_G - \langle O_X \rangle_{B_q(X)}| < \varepsilon_{loc}(B_q(X), T, \alpha)$. The local aspect lies in the fact that ε_{loc} depends on the ball $B_q(X)$. If such a bound exists, we have that $\langle O_X \rangle_G \geq \langle O_X \rangle_{B_q(X)} - \varepsilon_{loc}(B_q(X), T, \alpha)$. This lower bound is satisfied for all graphs $G \in \mathcal{G}$. The only dependence on the input graph now lies in $B_q(X)$. We can rewrite the minimization over $\mathcal{G}$ as:

$$\min_{G \in \mathcal{G}} \langle O_X \rangle_G \geq \min_{B_q(X) \in \mathcal{B}_q} (\langle O_X \rangle_{B_q(X)} - \varepsilon_{loc}(B_q(X), T, \alpha)) \quad (6.2)$$

$$\Rightarrow \langle O_X \rangle_{\mathcal{G}} \geq \langle O_X \rangle_{\mathcal{B}_q}^* \quad (6.3)$$

where $\langle O_X \rangle_{\mathcal{B}_q}^* = \min_{B_q(X) \in \mathcal{B}_q} (\langle O_X \rangle_{B_q(X)} - \varepsilon_{loc}(B_q(X), T, \alpha))$. Therefore, the approximation ratio becomes, for QA as a 0-local algorithm,

$$\rho_{MC} \geq \max_{T, \alpha} \langle O_X \rangle_{\mathcal{B}_q}^* \quad (6.4)$$

We can do the same for the 1-local analysis, when taking advantage of the neighborhood at radius 1 around the edge X . So we have that

$$\langle O_X \rangle_{\mathcal{G}}^{\Omega_i} \geq \min_{B_q(X) \in \mathcal{B}_q} (\langle O_X \rangle_{B_q(X)}^{\Omega_i} - \varepsilon_{loc}(B_q(X), T, \alpha)) = \langle O_X \rangle_{\mathcal{B}_{q,i}}^* \quad (6.5)$$

where $\mathcal{B}_{q,i}$ is the family of graphs $B_q(X) \in \mathcal{B}_q$ restricted to balls $B_q(X)$ for which $B_1(X) = \Omega_i$. Still assuming $n_1 = n_2 = 0$, the approximation ratio can be written as a new equation that depends only on these epsilons and worst edge energy among “small” ball of radius q :

$$\rho_{MC} \geq \max_{T, \alpha} \langle O_X \rangle_{\mathcal{B}_{q,3}}^* \quad (6.6)$$

It could be costly to compute the local bound for each ball $B_q(X)$. In particular it seems pointless to calculate them for those with high edge energy values. An upper bound would suffice to handle several cases at once.

Global bound. To avoid having to develop too many bounds ε_{loc} as the number of $B_q(X)$ explodes exponentially, it can be useful to apply a global bound for most of the balls in the minimization task. We define it as

$$\varepsilon(q, T, \alpha) = \max_{B_q(X) \in \mathcal{B}_q} \varepsilon_{loc}(B_q(X), T, \alpha) \quad (6.7)$$

We will see in Section 6.2 that this maximum is easy to derive from the analytical expression of the local bound. The global bound is used as follows:

$$\begin{aligned} (\langle O_X \rangle_{B_q(X)} - \varepsilon_{loc}(B_q(X), T, \alpha)) &\geq \langle O_X \rangle_{B_q(X)} - \max_{B_p(X) \in \mathcal{B}_q} \varepsilon_{loc}(B_q(X), T, \alpha) \\ &= \langle O_X \rangle_{B_q(X)} - \varepsilon(q, T, \alpha) \\ \Rightarrow \langle O_X \rangle_{\mathcal{B}_q}^* &\geq \langle O_X \rangle_{\mathcal{B}_q} - \varepsilon(q, T, \alpha) \end{aligned}$$

6.2. Tight LR bound on regular graphs for MaxCut approximation 139

Then for balls $B_q(X)$ for which $\langle O_X \rangle_{B_q(X)}$ is large enough, the global bound is sufficient in the minimization task over $\mathcal{B}_q$.

Now, we have reduced the problem of finding a minimum among an infinite number of graphs to a finite number of graphs. We point out that the finite size of $\mathcal{B}_q$ is only true if the maximum degree of the graphs we are looking at is fixed. Thus, for cubic graphs, the minimum is tractable. However, it might not be practically tractable if q is too large. As we mentioned in Sec. 5.3.2, the state-of-the-art LR bound achieves a value of 0.014 at $T = 3.15$ for $q = 71$. Recall that we need to solve the Schrödinger equation on each ball. Today's quantum computers are not precise enough to compute the edge energy, so this task is to be handled by a classical simulator. This highly limits the size of the balls that can be considered. The largest ball of $\mathcal{B}_q$ is the double binary tree T_q which has $2(2^{q+1} - 1)$ nodes. At already 30 nodes for $q = 3$, this defines the limit of classical resources we can use. Any known bound does better than the trivial bound of 2 (on the commutator) at this radius. Sec. 6.2 is dedicated to the derivation of a tighter LR bound tailored exactly for the purpose of having reasonable LR bound values for $q = 3$.

6.2 Tight LR bound on regular graphs for MaxCut approximation

The statement of our main result of the chapter is the following:

Theorem 6.1. *With a 1-local analysis, the approximation ratio reached by QA to solve MaxCut over cubic graphs is above 0.7020.*

In this section we prove Theorem 6.1 in two steps. First we develop a tight enough LR bound using the commutativity graph structure introduced in [Wang & Hazzard 2020] and by computing the exact values of the schedule's nested integrals. This determines the required value of q to achieve the best provable ratio. The second step is purely numerical and requires to enumerate each ball $B_q(X)$ and get the minimum of the final expected energy of the edge X inside these balls for each Ω_i , corrected by the LR bound. This new approach follows the idea presented in the last section of Chapter 5.

6.2.1 Tight LR bound on regular graphs in QA

As mentioned in section 6.1, the minimization to obtain the approximation ratio is intractable when performed over the entire graph family. However, the LR bound helps to reduce the size of the set on which we have to minimize to a finite subfamily of graphs, namely $\mathcal{B}_q$ (see Equation 6.3).

First we seek to develop a local bound $\varepsilon_{loc}(B_q(X), T, \alpha)$ such that $|\langle O_X \rangle_G - \langle O_X \rangle_{B_q(X)}| < \varepsilon_{loc}(B_q(X), T, \alpha)$ for all d -regular graphs G such that the ball at

distance q around edge X corresponds to $B_q(X)$. Let G be such a graph. Recall that it is rather difficult to manipulate this expression considering the initial state $|\psi_0\rangle$ so the first step (however costly in terms of tightness) is to get rid of this dependency by working directly with the evolution operator:

$$|\langle O_X \rangle_G - \langle O_X \rangle_{B_q(X)}| \leq \left\| O_X^G(T) - O_X^{B_q(X)}(T) \right\|$$

where we introduce the evolved observable under unitary operators U_T^G and $U_T^{B_q(X)}$ respectively, once again dropping the α from the notation. They are defined as:

$$O_X^G(T) = (U_T^G)^\dagger O_X U_T^G \quad (6.8)$$

$$O_X^{B_q(X)}(T) = (U_T^{B_q(X)})^\dagger O_X U_T^{B_q(X)} \quad (6.9)$$

In [Bravyi *et al.* 2006], the authors demonstrate that the evolution over a subset of nodes can also be expressed as :

$$O_X^{B_q(X)}(T) = \int d\mu(U) U O_X^G(T) U^\dagger$$

where $\mu(U)$ denotes the Haar measure over unitary operator U supported on $S = V \setminus V(B_p(X))$. Noticing that $U O_X^G(T) U^\dagger = O_X^G(T) + U[O_X^G(T), U^\dagger]$, we can bound the quantity of interest $\left\| O_X^G(T) - O_X^{B_q(X)}(T) \right\| \leq \int d\mu(U) \left\| [O_X^G(T), U^\dagger] \right\|$ for any unitary U supported on S . We are then left to bound the norm of this commutator.

Commutativity graph: Let us introduce a helpful tool presented in [Wang & Hazzard 2020] (see also [Chen & Lucas 2021]): the *commutativity graph* $\mathbf{G}(\mathbf{V}, \mathbf{E})$ associated to a Hamiltonian $H(t, G)$ having local interactions. In general, we can write $H(t) = \sum_j h_j(t) \gamma_j$ where γ_j are hermitian operators with norm less than unity and $h_j(t)$ are time-dependent scalars. The commutativity graph of $H(t)$ is constructed such that each operator γ_j is represented by a node j and two nodes j and k are connected if and only if $[\gamma_j, \gamma_k] \neq 0$. The structure of the graph captures the commutative and non-commutative relationships between the operators in the Hamiltonian.

In the case of MaxCut, the total time-dependent Hamiltonian writes

$$H(t, G) = \sum_{v \in V} \left(1 - \frac{t}{T}\right) \frac{\sigma_x^{(v)}}{\alpha} + \sum_{(a,b) \in E} \frac{t}{T} \frac{1 - \sigma_z^{(a)} \sigma_z^{(b)}}{2}$$

As described, the terms γ_j of the Hamiltonian are represented as nodes in the commutativity graph $\mathbf{G}$. We can distinguish two types of nodes in $\mathbf{V}$: those corresponding to interaction operators over the edges E of the original input graph, and those corresponding to local operators over nodes of V . This means that we have for $e = \langle a, b \rangle \in E$, $\gamma_e = \frac{1 - \sigma_z^{(a)} \sigma_z^{(b)}}{2}$ and for $v \in V$, $\gamma_v = \frac{\sigma_x^{(v)}}{\alpha}$. We can rewrite the total Hamiltonian as:

$$H(t, G) = \sum_{v \in V} h_v(t) \gamma_v + \sum_{e \in E} h_e(t) \gamma_e$$

Our notation fixes the time-dependent scalars as $h_e(t) = \frac{t}{T}$ for $e \in E$ and $h_v(t) = 1 - \frac{t}{T}$ for $v \in V$. Also, it is obvious to see that the commutativity graph is bipartite. The only pairs that do not commute are pairs $(\gamma_v, \gamma_e)_{v,e}$, where node v is incident to edge e in G . An example of commutativity graph is shown in Fig. 6.3.

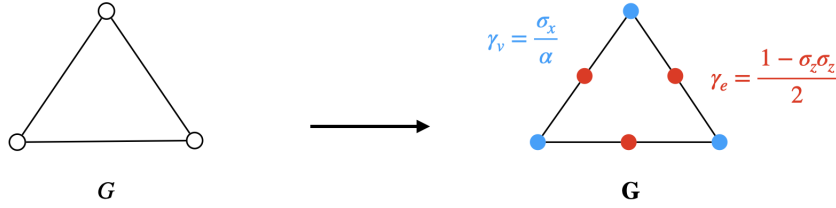


Figure 6.3: Example of a commutativity graph for the following Hamiltonian: $H(t, G) = \sum_{i=1}^3 (1 - \frac{t}{T}) \frac{\sigma_x^{(i)}}{\alpha} + \frac{t}{T} \frac{1 - \sigma_z^{(i)} \sigma_z^{(i+1)}}{2}$ where index i is taken modulo 3. Blue nodes represent 1-local operators and red nodes represent 2-interaction terms. In particular, blue nodes of $\mathbf{G}$ correspond to nodes of the original graph G , and red nodes correspond to its edges.

For a unitary A supported on S , we want to upper bound the quantity $\| [O_X^G(T), A] \|$. This edge X can be identified with a specific interaction term in the commutativity graph. Let us define X to be the node in $\mathbf{G}$ corresponding to the edge X and consider the operator $\gamma_X^A(T) = [\gamma_X(T), A]$ with $\gamma_X(t) = (U_T^G)^\dagger \gamma_X U_T^G$, dropping the dependency on G . Still following the first steps in [Wang & Hazzard 2020], we can arrive at a similar expression in the time-dependent regime (see Appendix C.2 for details):

$$\begin{aligned} \|\gamma_X^A(T)\| - \|\gamma_X^A(0)\| &\leq \sum_{v: \langle Xv \rangle \in \mathbf{G}} \int_0^T h_v(t) \|\gamma_X(t), \gamma_v^A(t)\| dt \\ &\leq \sum_{v: \langle Xv \rangle \in \mathbf{G}} \int_0^T (1 - \frac{t}{T}) \|\gamma_v^A(t)\| dt \end{aligned} \quad (6.10)$$

Now, we see on the right hand side of Eq. (6.10) that we have the norm of $\gamma_v(t)$ for some node v adjacent to X in $\mathbf{G}$. We can derive two update rules which we will use alternately depending on the considered node of $\mathbf{G}$, i.e. depending on whether it corresponds to an edge e of G or to a node v of G . These two rules are as follows:

$$\|\gamma_e^A(t)\| - \|\gamma_e^A(0)\| \leq \sum_{v: \langle ev \rangle \in \mathbf{G}} \int_0^t (1 - \frac{t'}{T}) \|\gamma_v^A(t')\| dt' \quad (6.11)$$

$$\|\gamma_v^A(t)\| - \|\gamma_v^A(0)\| \leq \frac{2}{\alpha} \sum_{e: \langle ve \rangle \in \mathbf{G}} \int_0^t \frac{t'}{T} \|\gamma_e^A(t')\| dt' \quad (6.12)$$

where we used two inequalities that for any t and any U :

$$\begin{aligned} \|\gamma_e^U(t)\| &= \|[\gamma_e(t), U]\| \leq \frac{1}{2} 2 \|\sigma_z^{(a)} \sigma_z^{(b)}\| \|U\| \leq \|U\| \\ \|\gamma_v^U(t)\| &= \|[\gamma_v(t), U]\| \leq \frac{1}{\alpha} 2 \|\sigma_x\| \|U\| \leq \frac{2}{\alpha} \|U\| \end{aligned}$$

where $\langle a, b \rangle$ is the edge on which γ_e is applied and we used the trivial commutation of the identity with any matrix. Note that $\|\gamma_j^A(0)\| = 0$ as long as j is inside $B_{q=k-1}(X)$, so we can iterate up to $2k$ steps as the first node outside $B_q(X)$ is a red node corresponding to an interaction term (see Fig. 6.4):

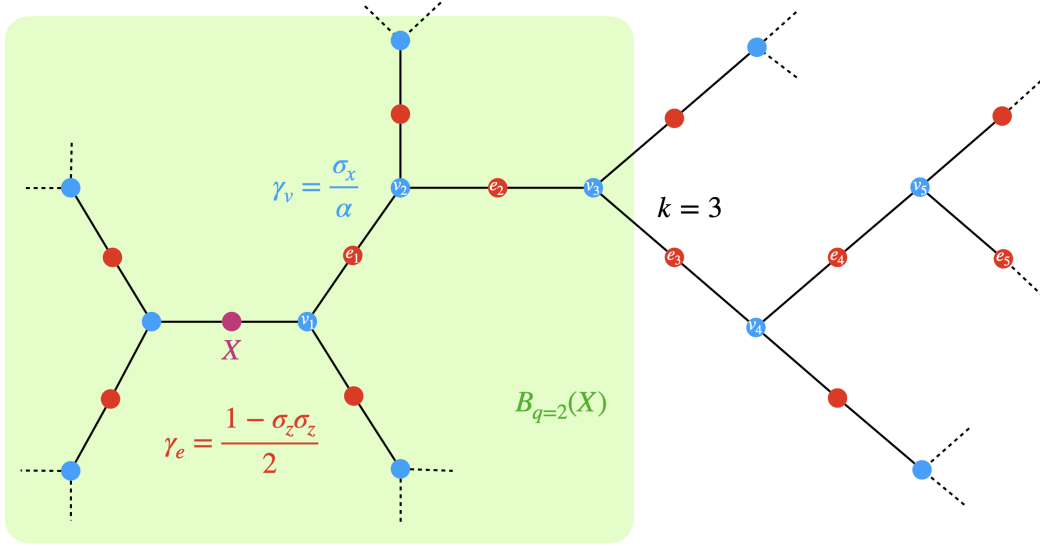


Figure 6.4: Commutativity graph of the cubic graph that maximizes the LR bound. The shaded area shows an example for $q = 2 = k - 1$.

$$\begin{aligned} \|\gamma_X^A(t)\| &\leq \left(\frac{2}{\alpha}\right)^k \sum_{v_1: \langle Xv_1 \rangle \in \mathbf{E}} \sum_{e_1: \langle v_1e_1 \rangle \in \mathbf{E}} \dots \sum_{e_k: \langle v_ke_k \rangle \in \mathbf{E}} \int_0^t h_{v_1}(t_1) \\ &\quad \times \int_0^{t_1} h_{e_1}(t_2) \dots \int_0^{t_{2k-1}} h_{e_k}(t_{2k}) \|\gamma_{e_k}^A(t_{2k})\| dt_{2k} \dots dt_2 dt_1 \end{aligned} \quad (6.13)$$

Now, let us introduce the following nested integral I_{2k} and I_{2k+1} that appears in Equation 6.13 where we replace each $h_j(t_i)$ by its expression and we pull out of the

6.2. Tight LR bound on regular graphs for MaxCut approximation 143

integral the factor T^{2k} and T^{2k+1} respectively so that the integrals depend only on k :

$$I_{2k} = \int_0^1 1 - u_1 \int_0^{u_1} u_2 \dots \int_0^{u_{2k-1}} u_{2k} du_{2k} \dots du_2 du_1$$

$$I_{2k+1} = \int_0^1 1 - u_1 \int_0^{u_1} u_2 \dots \int_0^{u_{2k}} 1 - u_{2k+1} du_{2k+1} \dots du_2 du_1$$

There is no known closed form for these integrals but we can easily have the exact numerical values for at least the first 100 points and we can upper bound it as we show in Appendix C.3. We can then write, following Eq. 6.13:

$$\|\gamma_X^A(T)\| \leq T^{2k} \left(\frac{2}{\alpha}\right)^k I_{2k} \sum_{v_1: \langle X v_1 \rangle \in \mathbf{E}} \sum_{e_1: \langle v_1 e_1 \rangle \in \mathbf{E}} \dots \sum_{e_k: \langle v_k e_k \rangle \in \mathbf{E}} \max_t \|\gamma_{e_k}^A(t)\|$$

where $\gamma_{e_k}^A(t)$ corresponds to an interaction node of the commutativity graph (i.e. red node) because we are at an even step. We used the fact that A is unitary and the dependence on $B_q(X)$ lies in computing the size of the nested sum. This bound can be improved by applying the update rule of Equation 6.10 and noticing that the first terms such that $\|\gamma_{e_k}^A(0)\| \neq 0$ only include all paths in $\mathcal{P}_{2k}(X, q)$. Where we define $\mathcal{P}_l(X, q)$ to be the set of paths of length l in $\mathbf{G}$ starting at X and ending in a node outside $B_{q=k-1}(X)$ (the green area in Fig. 6.4). After iterating several times, we get:

$$\begin{aligned} \|\gamma_X^A(T)\| &\leq T^{2k} \left(\frac{2}{\alpha}\right)^k I_{2k} \times |\mathcal{P}_{2k}(X, q)| \|\gamma_{e_k}^A(0)\| \\ &+ T^{2k+1} \left(\frac{2}{\alpha}\right)^k I_{2k+1} \times |\mathcal{P}_{2k+1}(X, q)| \|\gamma_{v_{k+1}}^A(0)\| \\ &+ T^{2k+2} \left(\frac{2}{\alpha}\right)^{k+1} I_{2k+2} \times |\mathcal{P}_{2k+2}(X, q)| \|\gamma_{e_{k+1}}^A(0)\| \\ &+ T^{2k+3} \left(\frac{2}{\alpha}\right)^{k+1} I_{2k+3} \times |\mathcal{P}_{2k+2}(X, q)| \|\gamma_{v_{k+2}}^A(0)\| \\ &+ T^{2k+4} \left(\frac{2}{\alpha}\right)^{k+2} I_{2k+4} \times |\mathcal{P}_{2k+3}(X, q)| \|\gamma_{e_{k+2}}^A(0)\| \\ &+ T^{2k+5} \left(\frac{2}{\alpha}\right)^{k+2} I_{2k+5} \times |\mathcal{P}_{2k+4}(X, q)| \|\gamma_{v_{k+3}}^A(0)\| \\ &+ T^{2k+6} \left(\frac{2}{\alpha}\right)^{k+3} I_{2k+6} \sum_{v_1: \langle X v_1 \rangle \in \mathbf{G}} \sum_{e_1: \langle v_1 e_1 \rangle \in \mathbf{G}} \dots \sum_{e_{k+2}: \langle v_{k+1} e_{k+2} \rangle \in \mathbf{G}} \max_t \|\gamma_{e_{k+3}}^A(t)\| \\ &= \varepsilon_{loc}(B_{q=k-1}(X), T, \alpha) \end{aligned} \tag{6.14}$$

We stop at the seventh iteration because numerically it appears that the bound reaches a minimum before increasing again. This is due to the fact that the num-

ber of paths in $|\mathcal{P}_l(X, q)|$ also increases exponentially with l . Each path considered above ends outside $B_q(X)$, because for the others we have $\|\gamma_j^A(0)\| \neq 0$ for $j \in [e_k, v_{k+1}, e_{k+1}, v_{k+2}, e_{k+2}, v_{k+3}]$. To define paths that go beyond $B_q(X)$, we implicitly extend the balls to maximize the number of paths in $\mathcal{P}_l(X, q)$ for $l \geq 2k$. Thanks to the upper bound of the integrals detailed in Appendix C.3, it is easy to see that the derived bound is decreasing with k and thus with q meaning that we have the following corollary:

Corollary 6.1. *For any $q > 0$ and any edge X inside a d -regular graph,*

$$\forall j > 0, \varepsilon_{loc}(B_{q+j}(X), T, \alpha) \leq \varepsilon_{loc}(B_q(X), T, \alpha)$$

the same goes for the global bound as taking the max preserves the inequality:

$$\forall j > 0, \varepsilon(q + j, T, \alpha) \leq \varepsilon(q, T, \alpha)$$

For the local bound, we need to compute for each ball $B_q(X)$ the number of paths. In practice, a subroutine counting the number of paths of a given size is used to compute the local bound. The last term with the multiple sums is counting for $(2d)^{k+3}$ as at each interaction term there are d possible choices of nodes and only 2 at the others. In this subsection we detailed the derivation of the LR local bound Eq. 6.14. In the next subsection, we pursue the derivation to get the global bound by taking the maximal number of paths.

6.2.2 Global LR bound

In this subsection, we use Eq. 6.14 to derive the global LR bound. As defined in Section 6.1, the global bound is obtained by considering the maximum of the local bound Eq. 6.14 over all balls in $\mathcal{B}_q$. To this end, we consider the worst-case scenario, i.e. the ball maximizing the possible number of paths. It's trivial to see that this corresponds to the cycle-free ball (Fig. 6.4).

In this cycle-free ball, we can count the number of paths corresponding to each term in LR bound's equation. In Fig. 6.5, we depict example paths for each of the necessary cases that we detail below.

For the first two terms, only direct simple paths reach the outside of $B_q(X)$, and there are $2(d-1)^k$ of them. The factor 2 comes from the initial choice at node X , you can go either left or right on Fig. 6.4. Once the side has been chosen, at each blue node (node v_r), there are $d-1$ possibilities, as the path can't go backwards by definition of direct single paths. In a path from X to the first node outside $B_{q=k-1}(X)$, i.e. of length $2k$, there are k blue nodes, bringing the total number of direct simple paths to $2(d-1)^k$ (path Fig. 6.5 (a)). The same number of paths is found for direct paths of length $2k+1$.

Then, for the third and fourth terms, we can distinguish simple direct paths that go one step further in $\mathbf{G}$, i.e. of length $2k+2$ and $2k+3$ respectively, from non-direct paths, i.e. passing several times through the same node or edge. For the third term counting paths of length $2k+2$, similarly to above, there are $2(d-1)^{k+1}$

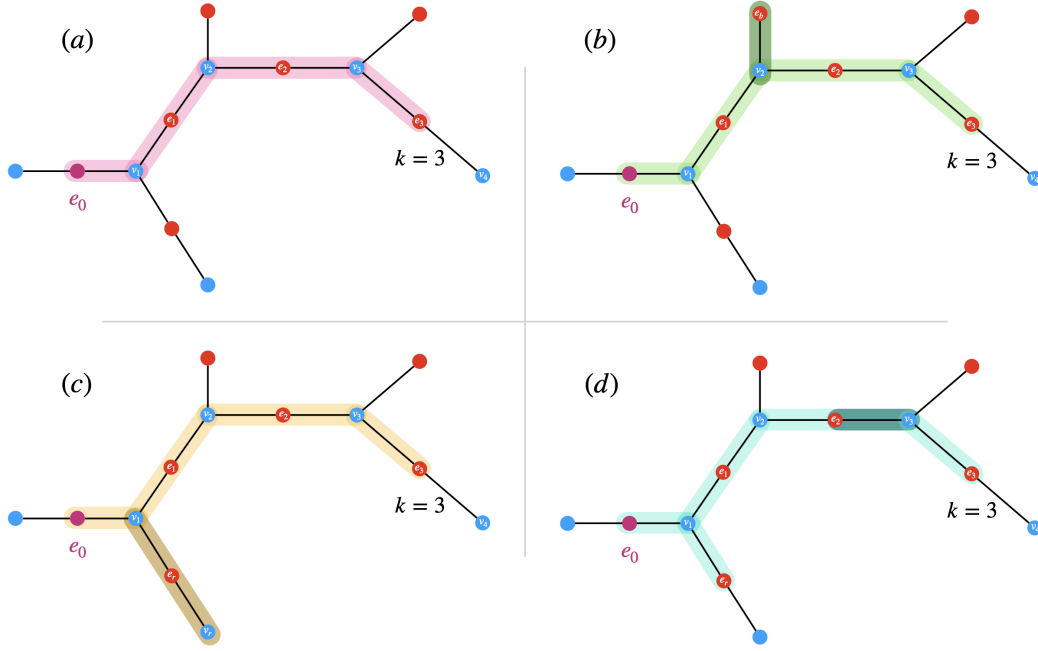


Figure 6.5: Example of different paths starting at X on the commutativity graph for the global bound with $k = 3$. (a) a simple path $(Xv_1e_1v_2e_2v_3e_3)$ in pink of length $2k$, (b) a green path $(Xv_1e_1v_2e_bv_2e_2v_3e_3)$ of length $2k + 2$ with a back-and-forth on one edge, (c) in orange a path $(Xv_1e_rv_re_rv_1e_1v_2e_2v_3e_3)$ of length $2k + 4$ with a back-and-forth on a branch of two edges and (d) in blue a path $(Xv_1e_rv_1e_1v_2e_2v_3e_2v_3e_3)$ of length $2k + 4$ with two back-and-forth on two edges at distance at most one from the simple path.

direct paths. For the non-direct ones, we need to count every edge that can be used at least twice in the path (path Fig. 6.5 (b)). At first, there are the two edges that start from X , then $(d - 1)$ at each blue node on the path and $+1$ at each red node, making a total of $2 + ((d - 1) + 1) * k$ edges that can be used twice in the $2(d - 1)^k$ possible paths. Therefore, the total number of paths in the third term is $2(d - 1)^{k+1} + (2 + dk) * 2(d - 1)^k$. For the fourth term, we use similar reasoning to arrive at $2(d - 1)^{k+1} + (dk + 2 + d - 1) * 2(d - 1)^k$ paths.

Let us see how the last two terms are derived. For the fifth term, we need to count all paths of length $2k + 4$ that lead to a red node outside the shaded area. There are three types of path: direct paths up to e_{k+2} counting for $2(d - 1)^{k+2}$, those with exactly one edge taking two or three times, i.e. reaching e_{k+1} counting for $(d(k + 1) + 2) * 2(d - 1)^{k+1}$ and those reaching e_k counting for $\left[(dk + 2) + \binom{dk+2}{2} + 2(d - 1) + k \right] * 2(d - 1)^k$. In the latter, a distinction is made: either an edge is used 4 or 5 times, or 2 different edges at a distance at most one from the direct path can be used 2 or 3 times (path Fig. 6.5 (d)), or finally choose a branch of length 2 far from the direct path (path Fig. 6.5 (c)). Similar reasoning is used for the sixth term.

We can then substitute the path counting in Eq. (6.14) to derive the following closed-form:

$$\begin{aligned}
\|\gamma_X^A(t)\| \leq & T^{2k} \left(\frac{2}{\alpha}\right)^k I_{2k} \times 2(d-1)^k + T^{2k+1} \left(\frac{2}{\alpha}\right)^{k+1} I_{2k+1} \times 2(d-1)^k \quad (6.15) \\
& + T^{2k+2} \left(\frac{2}{\alpha}\right)^{k+1} I_{2k+2} \times 2(d-1)^k [d(k+1) + 1] \\
& + T^{2k+3} \left(\frac{2}{\alpha}\right)^{k+2} I_{2k+3} \times 2(d-1)^k [d(k+2)] \\
& + T^{2k+4} \left(\frac{2}{\alpha}\right)^{k+2} I_{2k+4} \times 2(d-1)^k \left[d^2 \frac{(k+1)^2 + 3}{2} + d \frac{3k+2}{2} + k \right] \\
& + T^{2k+5} \left(\frac{2}{\alpha}\right)^{k+3} I_{2k+5} \times 2(d-1)^k \left[d^2 \frac{(k+2)^2 + 3}{2} + d \frac{k+1}{2} + k - 1 \right] \\
& + T^{2k+6} \left(\frac{2}{\alpha}\right)^{k+3} I_{2k+6} \times (2d)^{k+3} = \varepsilon(q = k-1, T, \alpha)
\end{aligned}$$

In this subsection, we developed the proof of our LR bound for any d -regular graph on which we want to solve MaxCut with a quantum annealing process. This bound achieves the best numerical value compared to the state-of-the-art of LR bounds. This is due to the fact that we have finely evaluated the nested integral with the standard schedule and used the commutativity graph of the Hamiltonian to tighten the bound. Here the free parameter α plays an important role: optimizing over its value will allow us to control the tightness of the bound (6.15). This point is further discussed in Section 6.3. In the next subsection, we apply the derived bounds (global and local) to obtain a numerical value of the approximation ratio for a 1-local analysis of QA.

6.2.3 Application to approximation ratio of MaxCut

In this subsection, we use the previously derived LR bounds to determine the approximation ratio of MaxCut over cubic graph with QA analyzed as a 1-local algorithm. The proof of the Theorem 6.1 proceeds with step 3, 4 and eventually 5, as illustrated in the overview of Fig. 6.1. We will use Eq. 6.6 to derive the approximation ratio with the 1-local analysis. For reproducibility, the code is available on GitHub¹.

For this purpose, after rigorous errors and trials, we set specific values $T = 3.33$, $\alpha = 1.53$ and $q = 3$, that establish the global bound $\varepsilon(q, T, \alpha) < 0.016$. In order to compute the required minimums of Eq. 6.5, $\langle O_X \rangle_{\mathcal{B}_{3,i}}^*$, we need to enumerate all balls in $\mathcal{B}_3$ and all cubic graphs in $\mathcal{B}_2$. We follow a smart hash-based iterative algorithm detailed in next paragraph 6.2.3. The algorithm generates 930449 balls. Employing the AnalogQPU simulator of Eviden Qaptiva (see next paragraph 6.2.3), we solve the Schrödinger equation to get the final state $|\psi^{B_3(X)}(T, \alpha)\rangle$ as described in the

¹<https://github.com/Arts-Braido/LR-bound-for-approximation>

Sec. 6.1.1. This allows to explicitly evaluate the value of $\langle O_X \rangle_{B_3(X)}$ for each ball in $\mathcal{B}_3$. We subtract the value of the global LR bound to the expected edge energy for each balls. To narrow down our selection, we retain only those balls for which $|\langle O_X \rangle_{B_3(X)} - \varepsilon(q, T, \alpha)| < 0.7020$. This initial step corresponds to step 3 of Fig. 6.1 and leads to the values for $\langle O_X \rangle_{\mathcal{B}_{3,1}}^* = 0.5502$ and $\langle O_X \rangle_{\mathcal{B}_{3,2}}^* = 0.6265$. These values satisfy the condition (see Appendix C.1) where the ratio reduces to $\rho_{MC} \geq \langle O_X \rangle_{\mathcal{B}_{3,3}}^*$. Consequently, our goal is to maximize this minimum for the remaining balls that have a corrected energy above 0.7020.

We are left with 7071 balls $B_3(X)$ with configuration Ω_3 at distance 1 around X , to which we apply the local bound. To compute the local bound we have access to a path counting algorithm as there is no closed form like the global bound. To find the maximum over parameters T and α , on Fig. 6.6, we plot the evolution of (a) $\max_T (\langle O_X \rangle_{B_3(X)} - \varepsilon(3, T, \alpha))$ and (b) $\max_T (\langle O_X \rangle_{B_3(X)} - \varepsilon_{loc}(B_3(X), T, \alpha))$ against α for the 18 worst balls $B_3(X)$ according to the global and local bounds respectively.

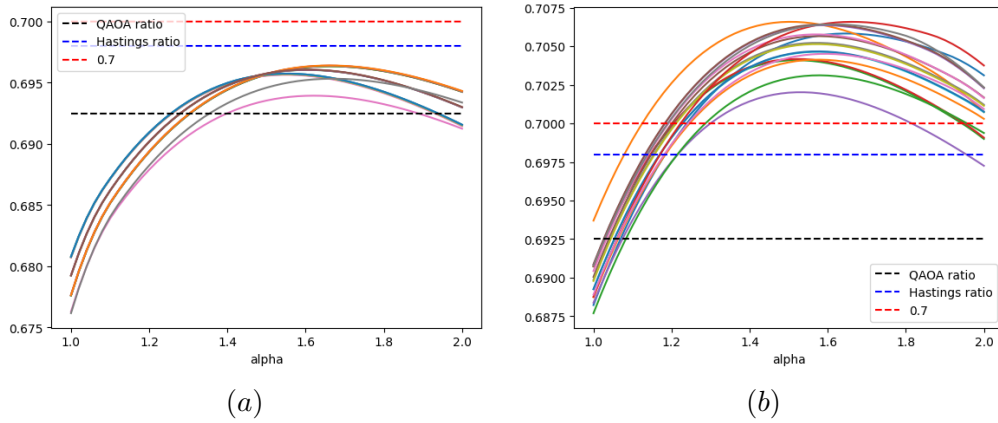


Figure 6.6: Evolution of (a) $\max_T (\langle O_X \rangle_{B_3(X)} - \varepsilon(3, T, \alpha))$ and (b) $\max_T (\langle O_X \rangle_{B_3(X)} - \varepsilon_{loc}(B_3(X), T, \alpha))$ against α for the 18 worst balls for which it goes under 0.7020 with the global bound.

The analysis reveals that around $\alpha = 1.5$ all these balls surpass the threshold of 0.7020, with the worst ball g depicted in Fig. 6.7. This plot finally fixes the value of $\langle O_X \rangle_{\mathcal{B}_{3,3}}^* = 0.70208\dots$ which proves Theorem 6.1.

To sum up, the constant-time analysis of Quantum Annealing (QA) for MaxCut over cubic graphs, analyzed as a 1-local algorithm, achieves an approximation ratio exceeding 0.7020. This result goes beyond any known ratio of 1-local algorithms, whether quantum or classical.

Let us take some lines to discuss the intriguing worst configuration we found. Note that the ball g in Fig. 6.7 is not the configuration we intuit in Sec. 5.3.1. Indeed this ball is bipartite and at first glance, the middle edge has no reason to converge toward 0. First we point out that W_3 (Fig. 5.6) is in fact the worst configuration

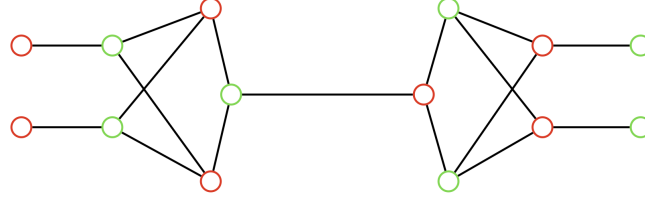


Figure 6.7: Ball $g = B_3(X)$ achieving the minimum $\langle O_X \rangle_{\mathcal{B}_{3,3}}^* = 0.70208\dots$

for $\alpha \in [1, 1.2]$ (green curve Fig. 6.6 (b)), then g has the smallest edge energy. This can be explained by the fact that on these plots we subtract the LR bound of the edge energy. It happens that W_3 has only one interaction with the rest of the graph while g has four which increase the local LR bound value. Secondly, the diameter of the ball seems to play a role here, W_3 has a diameter of 3 whereas the diameter of g is 5. Informally, left and right parts of g are still evolving independently for such short runtime pushing toward uncutting the middle edge.

Balls enumeration and numerical simulation: Here, we detail the specificity of our balls enumeration algorithm. We use the same idea as that developed in [Wurtz & Love 2021], i.e. an iterative algorithm. We first enumerate all the balls in $\mathcal{B}_1$ which are those on Fig. 5.2. Then, starting from $\mathcal{B}_{p-1}$, we enumerate $\mathcal{B}_p$ by completing the nodes on the boundary whose degree is less than d of the balls in $\mathcal{B}_{p-1}$. There are different ways of completing a node: either we link it to another existing node of degree less than d , or we create a new node and link it. By doing this on all nodes of degree less than d of a ball in $\mathcal{B}_{p-1}$, you get a ball in $\mathcal{B}_p$. You then apply an isomorphism test to check that the algorithm has not already generated isomorphic balls. The isomorphism test must also take into account the fact that there is a “marked” edge X in the balls $B_p(X)$. The `edge_match` argument in the Networkx [Hagberg *et al.* 2008], a Python library, isomorphism test takes this into consideration. On Fig. 6.8, we show the first iterations of the enumeration.

A naive approach could take more than one year to enumerate all the balls in $\mathcal{B}_3$. The isomorphism test is quite expensive as we do not have an efficient algorithm for it. Our idea is to hash the generated balls into a dictionary. The prerequisite for the hash function h to be useful is that for any two graphs G_1 and G_2 , $h(G_1) \neq h(G_2)$ implies that G_1 is not isomorphic to G_2 . If the hash is fast to generate and creates small enough bags of balls, then it greatly speeds up the enumeration task. For example a common hash function to quickly evaluate if two graphs might be isomorphic is the sorted list of degrees, if they are not equal it is certain that the two graphs are not isomorphic. This hash function however still evaluates equally for too many non isomorphic graphs, requiring too many calls to the isomorphic test. Instead, we used the tuple of the diameter and the sorted eigenvector centrality of the graph adjacency matrix to hash our balls. We manage to enumerate the 930449 balls of $\mathcal{B}_3$ and the regular ones in $\mathcal{B}_2$ in less than a day.

In order to compute each value $\langle O_X \rangle_B$, we relied on a full state vec-

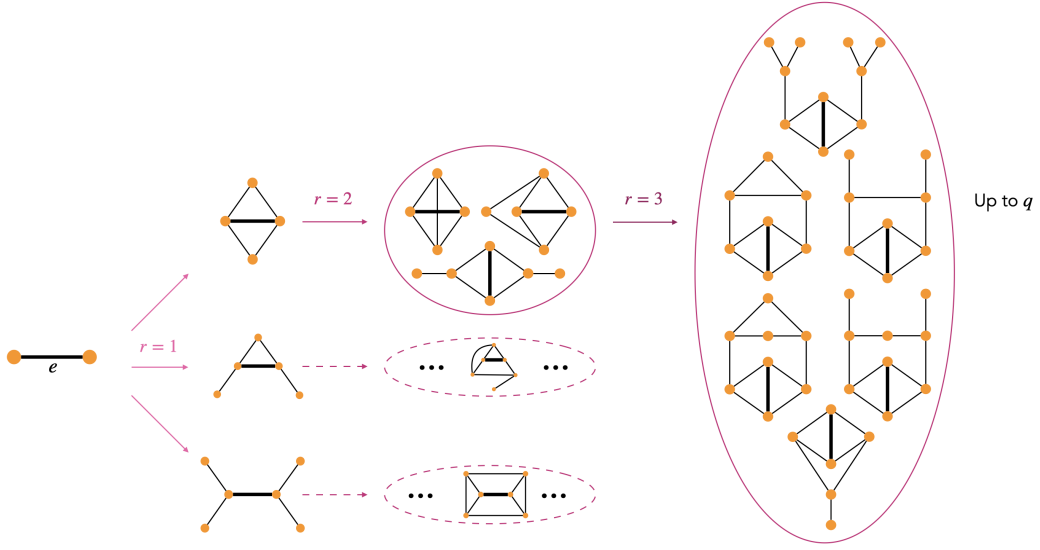


Figure 6.8: Iterative process to enumerate all balls of $\mathcal{B}_q$. There are 3 balls at $r = 1$, 123 balls at $r = 2$ and more than 900000 at $r = 3$.

tor representation using on a standard integrator (Boost’s ODEINT solver [Boost C++ Libraries 2022] used inside a proprietary code). The simulation were performed on a Eviden Qaptiva appliance using a mix of CPUs and GPUs. The simulations of the 930449 balls took approximately one month. The github code offers an alternative Qutip implementation of the same numerical simulations, for reproducibility.

6.3 Discussion

In this section, we discuss the result and the tightness of Theorem 6.1. In addition to the use of the commutativity graph, the exact value of the nested integrals alone would not have brought the ratio above the targeted numerical values of QAOA and Hastings local algorithm as we see on Fig. 6.6 (b) at $\alpha = 1$. Then we suggest other types of schedule to again make the most out of short time QA. Finally, we discuss about the generalization of the construction.

6.3.1 Tightness of the analysis method

The introduction of a new “hyperparameter” α significantly enhances the precision of the analysis. To attest this point, on Fig. 6.9, we plot the evolution of both the local $\varepsilon_{loc}(g, T, \alpha)$ and global $\varepsilon(q, T, \alpha)$ bounds against α for pairs (T, α) for which $\langle O_X \rangle_g = 0.7092$ and g denotes the ball of Fig. 6.7. The value of 0.7092 is totally arbitrary and similar plots are achieved with different values. So we clearly see on Fig. 6.9 that the LR bound is minimal around $\alpha = 1.5$ which means that the analysis of QA is tighter around this point. The trade-off between increasing α to

have a smaller LR bound also decrease the speed of the edge energy. We saw in Fig. 6.2 that increasing α slows down the evolution, needing more time to reach a similar value. This larger time increases in turn the LR bound.

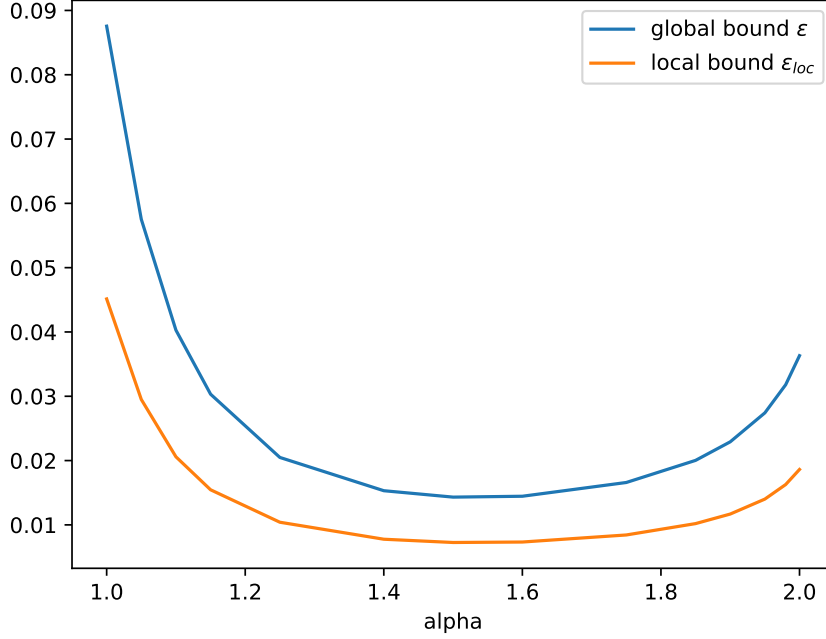


Figure 6.9: Evolution of $\varepsilon_{loc}(g, T, \alpha)$ and $\varepsilon(q = 3, T, \alpha)$ against α for pairs (T, α) for which $\langle O_X \rangle_g = 0.7092$ and g being the ball of Fig. 6.7.

We would like to draw readers' attention to the fact that both the Hastings algorithm and QAOA also include one or two hyperparameters for obtaining their best ratio value. In this sense, our parameterized QA analysis is nothing more complex. However, although these two other algorithms produce a tight ratio value, QA's is not tight simply because it is impossible to construct a graph such that every edge X has the g from Fig. 6.7 as its $B_3(X)$.

On the LR bound itself, we can see from Fig. 6.6 (a) that for many of the balls, the curves of the edge energy are indistinguishable. Indeed, at the pair (T, α) looked at, there's no visible difference in the value of the average edge energy. Even two balls in $\mathcal{B}_2$ can have the same trajectory. Informally, this suggests that LR bound is not yet tight, as for some ball the layer at distance 3 from the edge X does not impact the edge energy. As mentioned in Chapter 5, neglecting the initial state greatly affects the accuracy of the bound. In order to strongly improve the analysis it would be crucial to be able to take into consideration the initial state; unfortunately, at this stage we lack of mathematical tools for this purpose. Path counting is another lead suggested in [Chen *et al.* 2023]. In fact, it seems that only direct paths contribute to the LR bound. We explored this idea and the bound

would reach 0.7041. However the result of [Chen *et al.* 2023] cannot be adapted as is to our framework, so we cannot claim this approximation ratio at this stage.

6.3.2 Toward better ratios

An important field of research in the improvement of QA's performance is the optimization of the schedule. Indeed, all the construction above works for a linear interpolation with no specific optimization but one can look at the Hamiltonian

$$H(t, G) = (1 - f(t/T))H_0(\alpha) + f(t/T)H_1^G$$

with $f(t)$ such that $f(0) = 0$ and $f(1) = 1$. It is important to note that it also modifies the LR bound in the nested integrals. In commutativity graph notation, we have the $h_v(t) = 1 - f(t/T)$ and $h_e(t) = f(t/T)$. The challenge of this optimization is that one has to evaluate the energy on each of the 930449 balls for each new schedule to formally prove an improvement of the ratio. However, a good insight can be reached by looking only at the 123 balls in $\mathcal{B}_2$. In the linear case, we would obtain the same ratio if we only looked at $\mathcal{B}_2$ but using LR bounds of $q = 3$. For instance, we tried few cubic functions to already see that with $f(s) = 3.2s^3 - 4.8s^2 + 2.6s$, we should obtain a ratio above 0.7165 at $T = 2.77$ and $\alpha = 1.6$.

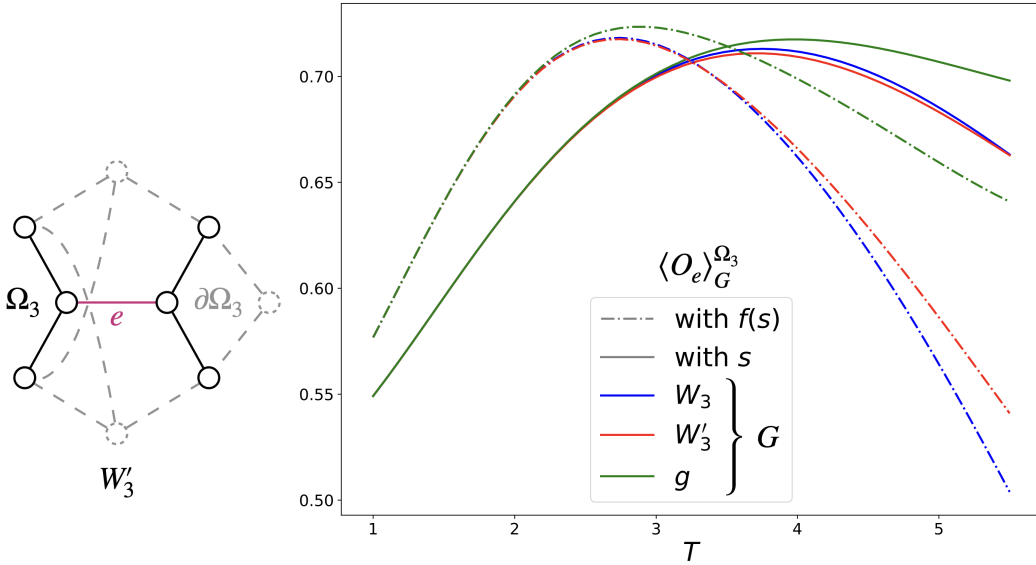


Figure 6.10: Effect of the schedule when computing the edge energy. Apart from balls g in Fig. 6.7 and W_3 in Fig. 5.5, W'_3 also have a low energy value. It has a similar structure to W_3 . In dash lines, the curve when using the cubic schedule $f(s)$. Those values are obtain without considering the LR bound. α is fixed to 1.5.

In Fig. 6.10, we plot the edge energy value inside three different balls : g , W_3 and W'_3 where W'_3 has a quite similar structure than W_3 , i.e. it is the other choice of $\partial\Omega_3$ that forces the middle edge to be uncut is the optimal state. Plain curves

are obtained with the linear interpolation and dashed ones with the schedule $f(s)$. It appears that this schedule speeds up the evolution, and the edge energies are reaching a slightly higher maximum at shorter times than the linear case. These two observations are aligned with improving the approximation ratio. A shorter runtime means a smaller LR bound. With the new schedule, W'_3 becomes the worst candidate and without LR correction its maximum over T and α is 0.7180. If the 0.7165 approximation ratio is correct, it would mean that we almost reached the tightest possible bound on the ratio with this approach.

6.3.3 Directions for generalizing the construction

We developed a LR bound for d -regular graphs applied to MaxCut on cubic graphs. We believe that our tools can be extended in several directions.

1. This work can be directly adapted to look for other d -regular graphs. A formal derivation of this bound for any d -regular graphs requires to enumerate all the balls in $\mathcal{B}_3$ that can be completed in a d -regular graphs. The code can be easily edited for this purpose. However large balls are certainly too big to solve Schrodinger equation for $d \geq 4$. It is still possible to get an intuition extrapolating the worst ball for $d = 3$ (Fig. 6.7). For example, the 1-local approximation ratio for MaxCut over 4-regular graphs may be close to 0.67.
2. For p -local analysis of QA with $p \geq 2$, the method developed here runs short as the time at which the best approximation ratio is achieved is certainly too large for the LR bound at $q = 3$. For a $p = 2$ -local analysis, our estimation gives that we would need to go up to $q = 5$ to achieve sufficiently small LR bound at time $T \simeq 6.1$, time for which the expected edge energy value seems to maximize. Nevertheless, by extrapolating at $p = 2$ the worst-case balls for $p = 1$ and by numerical experiments on these cases, we believe that the approximation ratio for MaxCut over cubic graphs is close to 0.77.
3. As discussed in the previous paragraph, the schedule can also be changed, the main work remains in the computation of the nested integrals of the schedule. Analytical bounds on these integrals are certainly too difficult to derive, but only numerical values are required to prove the bound. For any polynomial schedule, those integrals are easy to evaluate.
4. This construction can be applied to other combinatorial graph problems as we showed in Chapter 5 Sec. 5.1. With the approach taken in this chapter, more work is needed to adapt an ad hoc analytical formula for LR bound for the new problem Hamiltonian.

Conclusion

To conclude, in this work we developed a much tighter Lieb-Robinson bound compared to [Haah *et al.* 2021, Chen *et al.* 2023] by carefully manipulating the commu-

tativity graph and the nested integral of the QA schedule. Despite the continuous aspect of QA, we defined the notion of p -local analysis of the metaheuristic by approximating the full algorithm using its restriction to bounded radius subgraphs. Our 1-local analysis of QA allows us to analytically compare its performances with the performances of single-layer QAOA for MaxCut over cubic graphs. The tightness of the LR bound we have derived enables us to reduce the exhaustive numerical simulation to a tractable task that can be completed in a few weeks. Finally, we introduced a new parameter in the standard QA, enabling us to optimize the value of the ratio obtained and thus pass the 0.7 mark with a ratio going beyond 0.7020. This puts us ahead of single-layer QAOA and Hastings' 1-local algorithm for MaxCut over cubic graphs. The comparison has its limits, as the process we are studying is continuous and not intrinsically local, unlike the two algorithms mentioned. This work should be seen as a step forward in the study of quantum annealing, bringing more analytical tools to assess its algorithmic performances.

Also, as mentioned in Sec. 6.3.1, a slight improvement can be made on paths counting. Then, the remarkable small numerical value of our LR bound might be applied for a practical implementation of some Hamiltonian simulation schemes, e.g. [Haah *et al.* 2021].

As we hinted in Sec. 6.3.2, the bounds on the approximation ratio reached by this method are almost tight in the sense that even if we had considered that QA is strictly local, it would not significantly increase the proven ratio. However, the proven ratio is not tight for QA in the sense that no graph will reach this ratio because it would mean that every edge X of this graph has the worst candidate as their $B_3(X)$. For any of the “bad” configurations g , W_3 or W'_3 , it is impossible to construct such a graph. In the next chapter, we argue that the double binary tree is certainly a limiting case and we give an algorithm that efficiently computes the edge energy in the middle of the tree.

Effective approximation of MaxCut

Contents

7.1	Toward a state-dependent LR bound	155
7.1.1	Exact expansion of an edge energy	156
7.1.2	State-dependent LR bound	159
7.2	Double binary tree simulation and optimal approximation ratio	161
7.2.1	Symmetry in T_p for dimensional reduction	161
7.2.2	Optimal approximation ratio for QA?	165

In this chapter, we develop some incomplete work around the Lieb-Robinson bound and the effective approximation ratio for MaxCut on cubic graphs. We express the exact expansion of the energy of an edge within a graph, where each coefficient represents the influence of the graph structure at distance k from the edge. This expansion seems suitable for accessing the first state-dependent bound LR. Here, we bound the tail of the expansion by the bound developed in the previous chapter. The first terms are computable by evaluating the energy of the edges inside a double binary tree T_p of radius p . This approach for a state-dependent LR bound should compute the first terms for p up to six. Motivated by [Basso *et al.* 2022], we found a reduction of the Hilbert space thanks to the symmetry of the graph and were able to compute efficiently up to $p = 3$. With more advanced computational techniques, it is reasonable to assume that $p = 4$ is numerically accessible. Finally, we construct graphs that achieve the potential tight approximation ratio of QA.

7.1 Toward a state-dependent LR bound

In this section, we demonstrate an expansion of the energy of an edge X where the order coefficient k^{th} represents the influence of edges on the boundary of $B_k(X)$, i.e. edges at distance $k + 1$ from X . We then propose a way to use it with the bound developed in chapter 6 to derive a state-dependent LR bound.

First, we provide a technical lemma which is the main tool for the expansion. Let $G(V, E)$ be a graph on which we solve a combinatorial problem with the usual Hamiltonians of Sec. 2.2.1. The complete Hamiltonian is then written $H(t, G) =$

$(1 - \frac{t}{T})H_0 + \frac{t}{T}H_1^G$ for $t \in [0, T]$. H_0 is purely a 1-local Hamiltonian and H_1^G can have one- or two-local observables. The starting state is the uniform superposition $|\psi_0\rangle$. As in Chapter 5, for a subgraph Ω of G , we call $\partial\Omega$ the boundary of Ω , i.e. the edges outside Ω that have at least one end in Ω . We also denote by $U_{t_1, t}^G$ the solution of the Schrödinger equation between times t_1 and t :

$$i \frac{dU_{0, t}^G}{dt} = H(t, G)U_{0, t}^G$$

We write the final state $|\psi^G(t)\rangle = U_{0, t}^G|\psi_0\rangle$ and for an edge X the edge expected energy $\langle O_X \rangle_G = \langle \psi^G(t) | O_X | \psi^G(t) \rangle$. By adapting the proof of Proposition 5.1, we can have the following lemma:

Lemma 7.1. *For any edge X of G and any subgraph Ω such that $X \in E(\Omega)$, the following is true :*

$$\langle O_X \rangle_G - \langle O_X \rangle_\Omega = i \int_0^t dt_1 \frac{t_1}{T} \langle [H_{1, \partial\Omega}, (U_{t_1, t}^\Omega)^\dagger O_X U_{t_1, t}^\Omega] \rangle_G$$

where the expectation value inside the integral $\langle \dots \rangle_G = \langle \psi^G(t_1) | \dots | \psi^G(t_1) \rangle$ depends on the integrated variable t_1 .

Proof. We note $U_I(t) = (U_{0, t}^\Omega)^\dagger U_{0, t}^G$, the evolution in the interaction picture of $V_I(t) = (U_{0, t}^\Omega)^\dagger V(t) U_{0, t}^\Omega$ where the perturbation V is $V(t) = H_G(t) - H_\Omega(t)$. The state that evolves under $U_I(t)$ is written $|\psi_I(t)\rangle$.

$$\begin{aligned} \langle O_X \rangle_G - \langle O_X \rangle_\Omega &= \int_0^t dt_1 \frac{d}{dt_1} \left(\langle \psi_I(t_1) | (U_{0, t}^\Omega)^\dagger O_X U_{0, t}^\Omega | \psi_I(t_1) \rangle \right) \\ &= i \int_0^t dt_1 \left(\langle \psi_I(t_1) | [V_I(t_1), (U_{0, t}^\Omega)^\dagger O_X U_{0, t}^\Omega] | \psi_I(t_1) \rangle \right) \\ &= i \int_0^t dt_1 \left(\langle \psi^G(t_1) | [V(t_1), (U_{t_1, t}^\Omega)^\dagger O_X U_{t_1, t}^\Omega] | \psi^G(t_1) \rangle \right) \\ &= i \int_0^t dt_1 \frac{t_1}{T} \left\langle [H_{1, \partial\Omega}, (U_{t_1, t}^\Omega)^\dagger O_X U_{t_1, t}^\Omega] \right\rangle_G \end{aligned}$$

□

7.1.1 Exact expansion of an edge energy

By iteratively using Lemma 7.1, we shall prove the following theorem :

Theorem 7.1. *For any graph G , and any edge X in G , there exists r such that $B_{r+1}(X) = G$. For each edge, such r is bounded by the diameter of G . Let O_X be an observable supported on X , the quantity $\langle O_X \rangle_G$ can be expressed into the following expansions :*

- By introducing $O_Y^{(1)}(t_1) = i[H_1^{\partial B_1(X)}, (U_{t_1,t}^{B_1(X)})^\dagger O_X U_{t_1,t}^{B_1(X)}]$ and $O_Y^{(k)}(t_k) = i[H_1^{\partial B_k(X)}, (U_{t_k,t_{k-1}}^{B_k(X)})^\dagger O_Y^{(k-1)}(t_{k-1}) U_{t_k,t_{k-1}}^{B_k(X)}]$ we have:

$$\langle O_X \rangle_G = \langle O_X \rangle_{B_1(X)} + \sum_{k=1}^r \int_0^t dt_1 \frac{t_1}{T} \dots \int_0^{t_{k-1}} dt_k \frac{t_k}{T} \langle O_Y^{(k)} \rangle_{B_{k+1}(X)} \quad (7.1)$$

- Also, the expectation of O_X can be expanded like :

$$\langle O_X \rangle_G = \langle O_X \rangle_{B_1(X)} + i \sum_{k=1}^r \int_0^t dt_1 \frac{t_1}{T} \langle [H_1^{\partial B_k(X)}, (U_{t_1,t}^{B_k(X)})^\dagger O_X U_{t_1,t}^{B_k(X)}] \rangle_{B_{k+1}(X)} \quad (7.2)$$

Note that $V(B_r(X)) = V(G)$ and only some edges between two nodes of $B_r(X)$ are missing to have the full graph. In this sense, in some case we have $B_r(X) = G$ and the above equations adapt easily by noting that $H_1^{\partial B_r(X)} = 0$.

To prove Eq. 7.1, we apply Lemma 7.1 with $\Omega = B_1(X)$. Then we iterate with $\Omega = B_k(X)$ and the observable $O_Y^{(k)}$

$$\begin{aligned} \langle O_X \rangle_G &= \underbrace{\langle O_X \rangle_{B_1(X)} + \int_0^t \frac{t_1}{T} \langle O_Y^{(1)} \rangle_{B_2(X)} + \int_0^t dt_1 \frac{t_1}{T} \int_0^{t_1} dt_2 \frac{t_2}{T} \langle O_Y^{(2)} \rangle_{B_3(X)} + \dots}_{\langle O_X \rangle_{B_2(X)}} \\ &\quad + \int_0^t dt_1 \frac{t_1}{T} \dots \int_0^{t_{r-1}} dt_r \frac{t_r}{T} \langle O_Y^{(r)} \rangle_{B_{r+1}(X)} \\ &= \langle O_X \rangle_{B_1(X)} + \sum_{k=1}^r \int_0^t dt_1 \frac{t_1}{T} \dots \int_0^{t_{k-1}} dt_k \frac{t_k}{T} \langle O_Y^{(k)} \rangle_{B_{k+1}(X)} \end{aligned}$$

To prove Eq. 7.2, we first observe that :

$$\begin{aligned} \langle O_X \rangle_G &= \langle O_X \rangle_{B_1(X)} + (\langle O_X \rangle_{B_2(X)} - \langle O_X \rangle_{B_1(X)}) \\ &\quad + (\langle O_X \rangle_{B_3(X)} - \langle O_X \rangle_{B_2(X)}) \\ &\quad \vdots \\ &\quad + (\langle O_X \rangle_G - \langle O_X \rangle_{B_r(X)}) \end{aligned}$$

For each term $\langle O_X \rangle_{B_{k+1}(X)} - \langle O_X \rangle_{B_k(X)}$, we apply Lemma 7.1. It is valid because $B_k(X)$ is trivially a subgraph of $B_{k+1}(X)$ and so we have:

$$\langle O_X \rangle_{B_{k+1}(X)} - \langle O_X \rangle_{B_k(X)} = i \int_0^t dt_1 \frac{t_1}{T} \langle [H_1^{\partial B_k(X)}, (U_{t_1,t}^{B_k(X)})^\dagger O_X U_{t_1,t}^{B_k(X)}] \rangle_{B_{k+1}(X)}$$

□

To see the equivalence identity between Eq. (7.1) and (7.2), we observe that the k first terms of the expansions correspond to the final expectation $\langle O_X \rangle_{B_k(X)}$ and so for $t_1 \in [0, t]$, we have that :

$$\int_0^{t_1} dt_2 \frac{t_2}{T} \dots \int_0^{t_{k-1}} dt_k \frac{t_k}{T} \langle O_Y^{(k)} \rangle_{B_{k+1}(X)} = i \langle [H_1^{\partial B_k(X)}, (U_{t_1, t}^{B_k(X)})^\dagger O_X U_{t_1, t}^{B_k(X)}] \rangle_{B_{k+1}(X)}$$

These expansions reveal how an edge at distance k from X can affect the observed edge X . This theorem can be easily adapted for more complex Hamiltonian as long as H_0 is 1-local and there are interaction terms in H_1 only.

Now, we understand that the larger the border of $B_k(X)$, the greater the influence at distance k . Therefore, on a regular graph, edges of $\partial B_k(X)$ that link two nodes of $B_k(X)$ decrease the influence from the rest of the graph. This observation goes along with our intuition that small cycles are recognized first by the quantum annealing process. For the MaxCut problem, we recover that large girth graphs are the worst for the LR bound making the double binary tree the worst configuration for an edge. Thus it needs more time for QA to solve the problem on these graphs as far edges can greatly impact the expectation value of O_X .

To better understand the coefficients of the expansion, let us look closely at the quantity $\langle [H_1^{\partial B_k(X)}, (U_{t_1, t}^{B_k(X)})^\dagger O_X U_{t_1, t}^{B_k(X)}] \rangle_{B_{k+1}(X)}$. We see that we can apply our LR bound of Sec. 6.2 to see that these terms are exponentially small in k . Now, assume that $H_1^{\partial B_k(X)} = -\sum_{e \in \partial B_k(X)} O_e$ where each e has exactly one endpoint in $B_k(X)$. Let e_k be such edge labeled $\langle 0, 1 \rangle$ in the graph with node 0 being outside $B_k(X)$, the corresponding observable is given by :

$$O_{e_k} = \begin{pmatrix} \mathbf{0} & & (0) \\ & \mathbf{1} & \\ & & \mathbf{1} \\ (0) & & & \mathbf{0} \end{pmatrix}$$

where $\mathbf{0}$ and $\mathbf{1}$ are respectively the full zero square matrix and the identity matrix. Let A, B and D be the block matrices of $(U_{t_1, t}^{B_k(X)})^\dagger O_X U_{t_1, t}^{B_k(X)}$, i.e.

$$(U_{t_1, t}^{B_k(X)})^\dagger O_X U_{t_1, t}^{B_k(X)} = \begin{pmatrix} A & B & (0) \\ B^\dagger & D & \\ & & A & B \\ (0) & & B^\dagger & D \end{pmatrix}$$

where we added the dimension coming from the node 0 of the edge e_k . With these notations, the commutator becomes :

$$[O_{e_k}, (U_{t_1, t}^{B_k(X)})^\dagger O_X U_{t_1, t}^{B_k(X)}] = \begin{pmatrix} \mathbf{0} & -B & (0) \\ B^\dagger & \mathbf{0} & \\ & & \mathbf{0} & B \\ (0) & & -B^\dagger & \mathbf{0} \end{pmatrix}$$

So the value for $t_1 = 0$, $\langle \psi_0 | [O_{e_k}, (U_{t_1, t}^{B_k(X)})^\dagger O_X U_{t_1, t}^{B_k(X)}] | \psi_0 \rangle = 0$ and for $t_1 = t$, $[O_{e_k}, (U_{t_1, t}^{B_k(X)})^\dagger O_X U_{t_1, t}^{B_k(X)}] = [O_{e_k}, O_X] = 0$. In general, by decomposing the state

$$|\psi^{B_{k+1}(X)}(t_1)\rangle = \begin{pmatrix} \psi_{00} \\ \psi_{01} \\ \psi_{10} \\ \psi_{11} \end{pmatrix}$$

where $\psi_{x_0 x_1}$ is the reduced state to the subspace span by vectors $x = x_0 x_1 x_2 \dots$ with x_i the bit value of node i . Recall that we assume that edge $e_k = \langle 0, 1 \rangle$. Consequently,

$$\langle [O_{e_k}, (U_{t_1, t}^{B_k(X)})^\dagger O_X U_{t_1, t}^{B_k(X)}] \rangle_{B_{k+1}(X)} = 2i\mathcal{Im}(\langle \psi_{10} | B | \psi_{11} \rangle - \langle \psi_{00} | B | \psi_{01} \rangle)$$

where $\mathcal{Im}(z)$ calls the imaginary part of the scalar z . Given the $\mathbb{Z}_2$ symmetry of the MaxCut problem, we have that $|\psi_{11}\rangle = |\psi_{00}\rangle$ and $|\psi_{10}\rangle = |\psi_{01}\rangle$. We are left with :

$$2i\mathcal{Im}(\langle \psi_{01} | B | \psi_{00} \rangle - \langle \psi_{00} | B | \psi_{01} \rangle)$$

Further works need to be done on the understanding of the final distribution $|\psi^{B_{k+1}(X)}(t_1)\rangle$ and the matrix B to grasp fully the meaning of the coefficient terms in the expansion.

In this section, we prove two equivalent expansions of an observable expectation on X by adding information away from X in the graph. We argue that for the MaxCut problem, large girth graphs are more influenced by the outside. We started an analysis of the derived coefficients to understand the influence of an interaction at distance k of X . In the next section, we investigate how these expansions could help to derive a state-dependent LR bound.

7.1.2 State-dependent LR bound

In this section, we use the expansions of Theorem 7.1 to see the work we still need to do in order to derive a state-dependent LR bound.

With Eq. (7.1). Finding an upper bound $M_k(t)$ and a lower bound $m_k(t)$ on $\langle O_Y^{(k)} \rangle_{B_{k+1}(X)}$ gives at $t = T$,

$$\sum_{k=1}^r \frac{(T/2)^k}{k!} m_k(T) \leq \langle O_X \rangle_G - \langle O_X \rangle_{B_1(X)} \leq \sum_{k=1}^r \frac{(T/2)^k}{k!} M_k(T)$$

We see that the exponential time dependence of the LR bound is in $T/2$. By noticing that $\sum_{k=10}^{\infty} \frac{(T/2)^k}{k!} \simeq 3.10^{-5}$ and that it would be rather unexpected that $M_k(t)$ is not decreasing with k we have :

$$\langle O_X \rangle_G - \langle O_X \rangle_{B_1(X)} \leq \sum_{k=1}^9 \frac{(T/2)^k}{k!} M_k(T) + 3.10^{-5} M_{10}(T)$$

There is still some work to do on the first nine terms to derive a state dependent LR bound.

With Eq. (7.2). First, we apply Lemma 7.1 with $\Omega = B_p(X)$ for some finite p

$$\langle O_X \rangle_G = \langle O_X \rangle_{B_p(X)} + i \int_0^T dt_1 \frac{t_1}{T} \langle [H_1^{\partial B_p(X)}, (U_{t_1, T}^{B_p(X)})^\dagger O_X U_{t_1, T}^{B_p(X)}] \rangle_G$$

The last term can be bounded by the norm of the commutator to get rid of uncontrolled dependence in the input graph G to obtain :

$$\langle O_X \rangle_G \leq \langle O_X \rangle_{B_p(X)} + \int_0^T dt_1 \frac{t_1}{T} \left\| [H_1^{\partial B_p(X)}, (U_{t_1, T}^{B_p(X)})^\dagger O_X U_{t_1, T}^{B_p(X)}] \right\|$$

We know that $|\partial B_p(X)| \leq 2^{p+2}$ with equality for $B_p(X)$ the double binary tree. Let e_p denote an edge that belongs to the border $\partial B_p(X)$. From our LR bound $\varepsilon_{loc}(B_p(X), T, \alpha)$ developed in Chapter 6 Sec. 6.2, we have :

$$\langle O_X \rangle_G \leq \langle O_X \rangle_{B_p(X)} + \frac{T}{2} |\partial B_p(X)| \varepsilon_{loc}(B_p(X), T, \alpha)$$

For a runtime and parameter α that achieved the best approximation ratio in Sec. 6.2, i.e. $T = 3.33$ and $\alpha = 1.53$, it appears that $p = 6$ is enough to bound the tail of the expansion. We know that $\varepsilon_{loc}(B_p(X), T, \alpha) \leq \varepsilon(p, T, \alpha)$, the global bound is achieved for the double binary tree T_p of depth p . We have that $\frac{T}{2} 2^{p+2} \varepsilon(p, T, \alpha) \simeq 1.8 \times 10^{-4}$. This choice also fixes all $B_k(X)$ to be T_k :

$$\langle O_X \rangle_G - \langle O_X \rangle_{T_1} \leq i \sum_{k=1}^5 \int_0^t dt_1 \frac{t_1}{T} \langle [H_1^{\partial T_k}, (U_{t_1, t}^{T_k})^\dagger O_X U_{t_1, t}^{T_k}] \rangle_{T_{k+1}} + 1.8 \times 10^{-4}$$

where each term in the sum is a difference $\langle O_X \rangle_{T_{k+1}} - \langle O_X \rangle_{T_k}$ for $k \in [1, 5]$. In next Sec. 7.2, we provide an algorithm that allows us to compute $\langle O_X \rangle_{T_k}$ more efficiently than directly solving the Schrödinger equation. This method is still limited by the classical computational resources and runs easily up to $p = 3$, $p = 4$ seems accessible with more advanced programming code. We find that

$$\begin{aligned} |\langle O_X \rangle_{T_2} - \langle O_X \rangle_{T_1}| &\simeq 3.5 \times 10^{-3} \\ |\langle O_X \rangle_{T_3} - \langle O_X \rangle_{T_2}| &\simeq 0.1 \times 10^{-4} \end{aligned}$$

By assuming that $|\langle O_X \rangle_{T_{k+1}} - \langle O_X \rangle_{T_k}| \leq |\langle O_X \rangle_{T_3} - \langle O_X \rangle_{T_2}|$ for $k \in [3, 5]$, we finally obtain :

$$\langle O_X \rangle_G - \langle O_X \rangle_{T_1} \leq 3.72 \times 10^{-3}$$

Of course there might exist some ball $B_6(X) \neq T_6$ such that the bound is larger than for the binary tree but it is also most likely that we can bound the tail of the expansion for p less than 6. Indeed, these balls have at least one close loop that decreases the size of $\partial B_p(X)$ and $\varepsilon_{loc}(B_p(X), T, \alpha)$. Nevertheless, as highlighted in [Basso *et al.* 2022], an edge in a random regular graph has a high probability to have a T_k as a neighborhood at distance k , which motivates the work of next section.

7.2 Double binary tree simulation and optimal approximation ratio

In this section, we first provide an algorithm to decrease the dimension of the Hilbert space to compute more efficiently the edge energy in the middle of a T_p . This approach is motivated by the recent work of [Basso *et al.* 2022] where the authors managed to compute this latter energy with a QAOA circuit for p up to 17. The last section is dedicated to a discussion on the optimal approximation ratio QA reaches in short time by constructing cubic graphs where each edge X has a T_p as their $B_p(X)$.

7.2.1 Symmetry in T_p for dimensional reduction

The objective of this section is to leverage the symmetry present in T_p to reduce the dimensionality of the Hilbert space, which typically grows exponentially with the number of nodes n_p , where $n_p = 2(2^{p+1} - 1)$. The classical computational cost begins to increase noticeably around $n_3 = 30$. Initially, we label the binary tree in a manner such that a bipartition $x = x_L x_R$ encodes the bipartition x_L (or x_R) of the left (or right) part. Left and right parts are defined when putting the edge root at the top horizontally. This labeling process can be iteratively applied, where $x_L = x_L[1]x_{LL}x_{LR}$, and $x_L[i]$ corresponds to the i^{th} bit of x_L . An illustration for the case $p = 2$ is shown in Fig. 7.1. Furthermore, following this iterative process, we can represent $x = x_L x_R = x_L[1]x_{LL}x_{LR}x_R[1]x_{RL}x_{RR}$, and so forth. In addition to the usual $\mathbb{Z}_2$ symmetry, T_p exhibits a left/right symmetry at each “level” as it locally looks like a binary tree. This symmetry implies that the ordering $x \dots L x \dots R$ can always be rearranged as $x \dots R x \dots L$ without altering the resulting bipartition. We can then define equivalence classes among of the 2^{n_p} bit-strings.

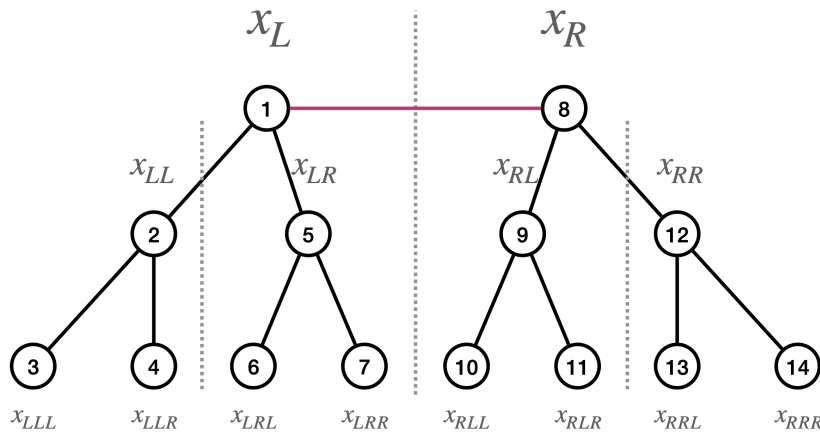


Figure 7.1: Example of a labeling for T_2 with the notation.

Hence, for every equivalence class of bipartition, we can designate a unique representative bit-string. We choose to select one of minimal Hamming weight among

all potential bit-strings within the equivalence class. Additionally, we ensure that the representative has its left part smaller than the right part at every level of the tree. Algorithm 1 outlines the routines needed to compute such a representative bipartition. The $*$ operation signifies string concatenation, and $|\cdot|$ denotes the computation of the Hamming weight. Furthermore, x_L and x_R represent the left and right parts, respectively, of a given x split in half.

Algorithm 1 Computation of the representative bit-string given any input string x . LR_norm deals with the left/right symmetry at each level and repr_bs deals with the $\mathbb{Z}_2$ symmetry above the LR_norm subroutine

<pre> LR_norm (x, at_root=false): if length(x) ≤ 1: return x end if at_root: root ← empty string else root ← x[1] x ← x[2:end] end left ← LR_norm(x_L, false) right ← LR_norm(x_R, false) if left < right: return root*left*right end return root*right*left end </pre>	<pre> repr_bs (x): x̄ ← bitflip(x) if x̄ < x : return LR_norm(x̄, true) end if x̄ = x : return min(LR_norm(x, true), LR_norm(x̄, true)) end return LR_norm(x, true) end </pre>
--	---

Now that we define a way to uniquely represents an equivalence class, we can solve the modified Schrödinger equation on the new Hilbert space composed only by the representatives. If N_{eff} is the new size of the Hilbert space, i.e. the total number of representative bit-strings, the initial state $|\tilde{\psi}_0\rangle$ is a vector of size N_{eff} with all entries equal to $2^{-\frac{n_p}{2}}$. The current state $|\tilde{\psi}(t)\rangle$ is ruled by the following equation :

$$i \frac{d}{ds} |\tilde{\psi}(s)\rangle = T \tilde{H}(s) |\tilde{\psi}(s)\rangle$$

where $\tilde{H}(s) = (1-s)\tilde{H}_0 + s\tilde{H}_1$. For two representatives x and y , $\tilde{H}_1$ is a diagonal matrix of size $N_{\text{eff}} \times N_{\text{eff}}$ with entries $\langle \tilde{x} | \tilde{H}_1 | \tilde{x} \rangle = \langle x | H_1^{T_p} | x \rangle$ and $\langle \tilde{y} | \tilde{H}_0 | \tilde{x} \rangle = \sum_{z, \text{repr_bs}(z)=x} \langle y | H_0 | z \rangle$. We denoted by $|\tilde{x}\rangle$ the vector in the reduced Hilbert space corresponding to bipartition x and by H_0 and $H_1^{T_p}$ the usual Hamiltonians for the MaxCut problem (see Sec. 2.2.2). $\tilde{H}_1$ is trivial to compute. Then, we follow Algorithm 2 to enumerate all representatives stored in the set **total** and to compute the values to complete $\tilde{H}_0$. The subroutine **flip(x,i)** returns a bit-string y equals to x with bit x_i flipped. **label** is an iterator from 1 to 2^{n_p} which attributes the new label to a representative, namely the $\tilde{x}$, so for a bipartition x , **label_dict[x]= $\tilde{x}$** . For two bipartitions x and y , the dictionary **count_neigh** is constructed such that

7.2. Double binary tree simulation and optimal approximation ratio163

$\text{count_neigh}[x][y] = -\langle \tilde{y} | \tilde{H}_0 | \tilde{x} \rangle$. We use two operators on sets, $\setminus$ for the difference and $+$ for the union.

Algorithm 2 Enumeration of all representative bit-strings at a given p stored in the set `total` and the relation between them in the dictionary `count_neigh`.

Initialization:

```

repr_set ← Set( $0^{n_p}$ )
total ← Set( $0^{n_p}$ )
count_neigh ← Dict( )
label_dict ← Dict( )
label ← iterator( $2^{n_p}$ )
while repr_set is not empty:
    new_repr_set ← Set( )
    for repr in repr_set:
        if repr not in label_dict:
            label_dict[repr] = next(label)
        end
        count_neigh[repr] = Dict( )
        for i in 1 ...  $n_p$ :
            new_x ← flip(repr, i)
            new_repr ← repr_bs(new_x)
            if new_repr not in label_dict:
                label_dict[new_repr] ← next(label)
            end
            if new_repr not in count_neigh[repr]:
                count_neigh[repr][new_repr] ← 0
            end
            count_neigh[repr][new_repr] ← +1
            if |new_x| ≥ |repr|:
                append(new_repr, new_repr_set)
            end
        end
    end
    new_repr_set ← new_repr_set \ total
    total ← total + new_repr_set
    repr_set ← new_repr_set
end

```

This approach cleverly uses all symmetries in a double binary tree to reduce the Hilbert space as much as possible. We still guarantee an exact computation of the final quantum state. Indeed, it is rather direct to reconstruct the state $|\psi(s)\rangle$ given $|\tilde{\psi}(s)\rangle$, each amplitude $\langle x | \psi(s) \rangle = \langle \tilde{x} | \tilde{\psi}(s) \rangle$. The final step is to compute the value of the edge X energy in the middle of the double binary tree $\langle O_X \rangle_{T_p}$. With the same labeling as Fig. 7.1, for a bipartition x , edge $X = \langle a, b \rangle$ bit value is given by $x_a = x_L[1]$ and $x_b = x_R[1]$. Consequently, by calling $|\psi_T\rangle$ (resp. $|\tilde{\psi}_T\rangle$) the state $|\psi(s)\rangle$ (resp. $|\tilde{\psi}(s)\rangle$) at the end of the annealing and N_x the number of bipartitions

that have x as a representative, we have :

$$\langle O_X \rangle_{T_p} = \sum_{x \in \{0,1\}^{n_p}, x_a \neq x_b} |\langle x | \psi_T \rangle|^2 \quad (7.3)$$

$$= \sum_{x \in \text{total}, x_a \neq x_b} N_x |\langle \tilde{x} | \tilde{\psi}_T \rangle|^2 \quad (7.4)$$

We ran the different procedures for $p \in \{1, 2, 3\}$ and we obtained Table 7.1 and Fig. 7.2. $p = 4$ seems doable with a very powerful Ordinary Differential Equation (ODE) solver.

p	n_p	N_{eff}
1	6	$12 \simeq 2^{3.6}$
2	14	$462 \simeq 2^{8.85}$
3	30	$816312 \simeq 2^{19.64}$
4	62	$\sim 2^{41}$

Table 7.1: Results from the reduction of the Hilbert space on T_p . The lightgrey row is a guess for N_{eff} inferred from the first three where the power is approximately just below $\frac{2}{3}n_p$.

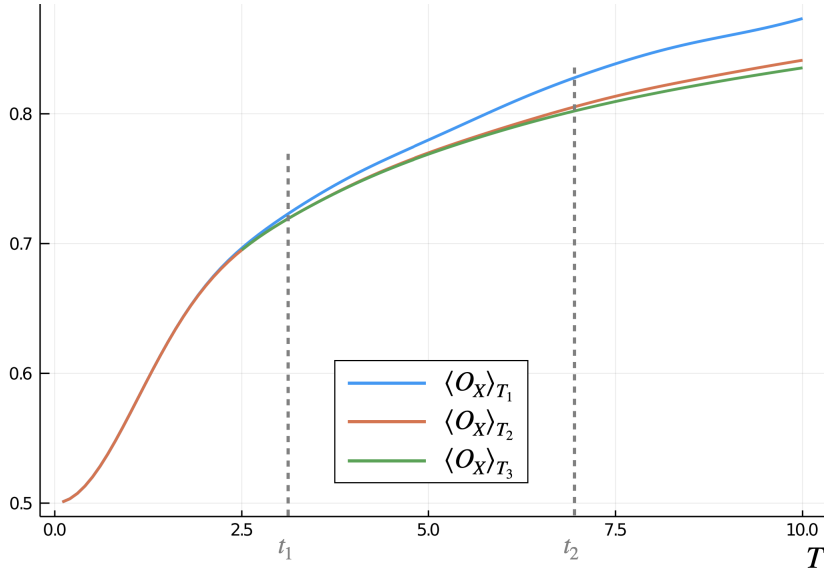


Figure 7.2: Edge energy in a double binary tree T_p for p ranging from 1 to 3 and $\alpha = 1$. $t_1 = 3$ points out the runtime for which $\langle O_X \rangle_{T_1}$ differs by 0.0035 from $\langle O_X \rangle_{T_2} \simeq 0.7151$, similarly $t_2 = 7$ between $\langle O_X \rangle_{T_2}$ and $\langle O_X \rangle_{T_3} \simeq 0.8028$.

This method is still limited by the size of the Hilbert space N_{eff} which seems to grow exponentially fast with n_p . For the standard QA regime ($\alpha = 1$ and linear schedule), Fig. 7.2 points out the arbitrary chosen runtimes t_p that are playing

the role of the number p of layers in QAOA. As mentioned in [Basso *et al.* 2022], random regular graphs generally have large girth, so they locally look like tree T_p . This observation would bring the effective optimal approximation ratio of QA with a 1-local analysis to 0.7151 and with a 2-local analysis above 0.8. To assess the tightness of those values, in the next section we construct such cubic graphs.

7.2.2 Optimal approximation ratio for QA?

In this section, we construct 3-regular graphs that achieve the value of the double binary tree middle edge energy as their approximation ratio. The property of these graphs is that each edge has exactly the same neighborhood at a given distance. They are denoted G_6 , G_{14} and G_{30} , the index corresponding to the number of their vertices (Fig. 7.3).

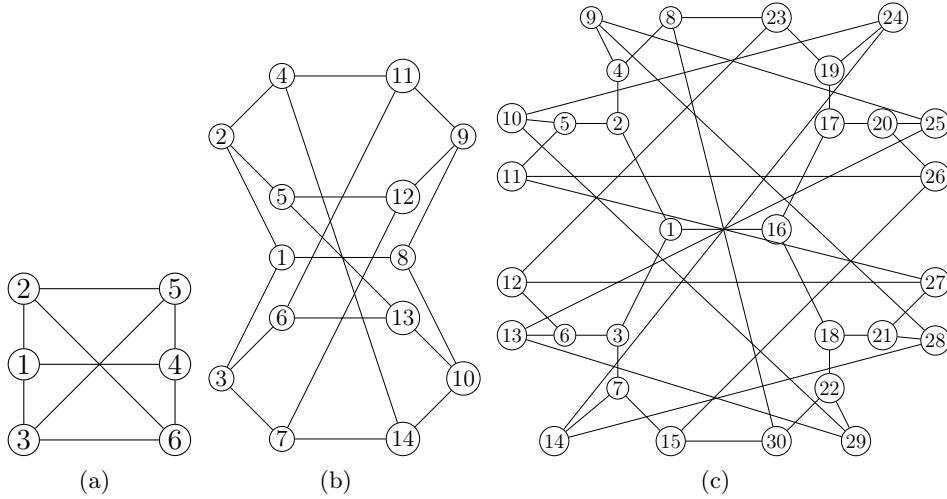


Figure 7.3: Worst graphs from a QA point of view in short running time. (a) G_6 with minimum cycle size = 4. (b) G_{14} with minimum cycle size = 6. (c) G_{30} with minimum cycle size = 8. Each edge has exactly the same neighborhood.

In Fig. 7.4, we plot the approximation ratio reached by the graphs of Fig. 7.3 as a function of T . For G_{30} , each data points took more than 15 hours to compute. We observe that for an edge X , $\langle O_X \rangle_{T_1} \simeq \frac{-\langle H_1^{G_{14}} \rangle_{G_{14}}}{C_{opt}(G_{14})}$ and $\langle O_X \rangle_{T_2} \simeq \frac{-\langle H_1^{G_{30}} \rangle_{G_{30}}}{C_{opt}(G_{30})}$. These graphs do reach the announced value as their approximation ratio. In graph theory, these graphs are known as a cage. A cage is a regular graph that has as few nodes as possible for its girth, the smallest induced cycle of the graph. A cage is parametrized by two parameter (d, g) , d being the regularity of the graph and g the size of the girth. G_6 is a $(3, 4)$ -cage, G_{14} a $(3, 6)$ -cage and G_{30} a $(3, 8)$ -cage. For $d = 3$, the largest cage is a $(3, 12)$ graph with 126 nodes. These graphs have an even greater number of symmetry than the double binary tree, and it would be quite interesting to take advantage of it to compute their approximation ratio for large cages.

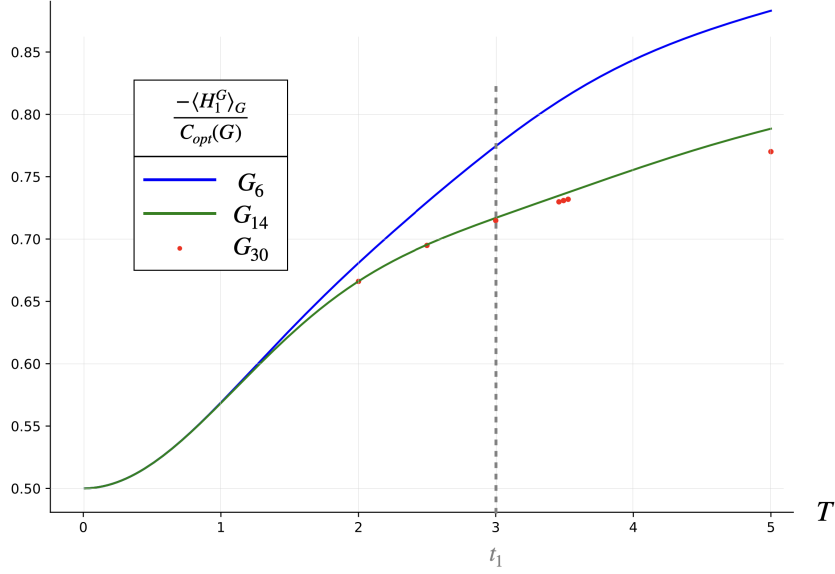


Figure 7.4: Approximation ratio reached by G_6 (blue line), G_{14} (orange line) and G_{30} (red dots) against running time T . The grey dotted line points out time $t_1 = 3$.

To complete this analysis, we also perform a benchmark over thousands cubic graphs and compute their approximation ratio. All of the tested graphs have a higher ratio than G_{30} . Non-bipartite graphs start at $T = 0$ with an offset because the random guess already does better than 0.5. Bipartite graphs have a greater ratios if there are small closed loops. Small cycles are considered faster by the quantum process.

Conclusion

In conclusion to this exploratory chapter, we initiated the development of a state-dependent [LR](#) bound and highlighted areas requiring further investigation to achieve a significant improvement over usual [LR](#) bounds. The aim behind this approach is to establish a tighter approximation ratio of [QA](#) on MaxCut problems, particularly on regular graphs. We believe that the worst-case scenario occurs on the double binary tree. Consequently, we devised a more efficient procedure for computing edge energies within a binary tree's midpoint, surpassing the direct solving of the Schrödinger equation. Our approach leverages every symmetry present in T_p , resulting in a reduction of the Hilbert space by a factor of $2^{n_p/3}$. This allows us to speculate that the optimal approximation ratio achievable by QA, analyzed as a local algorithm on MaxCut over cubic graphs, is 0.715 for $\alpha = 1$. To validate the tightness of this value, we constructed regular graphs, known as cages, that attain this optimal ratio.

Conclusion & Outlooks

Conclusion

In this thesis, we mainly tackled the analysis of the computational complexity of analog quantum computing by borrowing some well-known analytical results from theoretical physics and by introducing and developing new theoretical tools.

In the first part, we focused on the analysis of exponentially closing gaps, which are the signature of avoided level crossings. These phenomena are of important interest in the complexity analysis of Adiabatic Quantum Computing, as the main theoretical foundation is the adiabatic theorem, that guarantees the optimal final solution if the running time is inversely proportional to the minimum gap squared. This latter result puts the analysis of the minimum gap scaling as the principal candidate to prove any efficiency or inefficiency of the AQC framework. In Chapter 3, we first explored the limitation of the most recent characterization by [Choi 2020] of an AC (Def. 2.6) by explaining in which situations the definition fails to classify an AC. Supported by theoretical results we proved, **we introduced a new definition of an AC to capture more possible cases (Def. 3.1)** that could be labeled as AC. We illustrated the intuition that guided us to our definition with a toy model of the k -clique problem. This example exhibits an AC that is not captured by Def. 2.6, but is by ours (Def. 3.1). The main difference between this two definitions is the point of view adopted, Def. 2.6 crossings focus on examining “the direction of the ground-state” whereas our definition of AC examines “the provenance of the ground-state”. Our main observation is that at crossing point, the derivative of a certain spectral quantity is inversely proportional to the minimum gap, making the derivative almost undefined as the gap reduces at an exponential rate. Our characterization also has its own limitations, which we have highlighted.

After developing our intuition on the AC phenomenon, in Chapter 4, we employed **perturbation theory** at the beginning and end of the quantum evolution **to derive a condition on the occurrence of an exponentially closing gap** via first-order quantum phase transition. This condition is valid under certain assumptions concerning local minima as well as the validity of the first-order perturbative expansion. We apply our result on the MaxCut problem. In particular, we succeeded in proving the hypotheses in the case of **regular bipartite graphs** to show that the gap does not shrink exponentially fast with the input size, making **AQC efficient in these cases**. We further investigated arbitrary bipartite graphs and constructed a family of graphs satisfying the occurrence condition of **exponentially decreasing gap**. **High irregularity**, which is a property of the graphs constructed, seems to make efficient MaxCut solving difficult for AQC. To complete our work, we provided a numerical gap analysis of the graphs in question, as well as for the performance of AQC on this class of graphs. The gap study validated the theory, however, it appears that the quantum evolution still achieves a constant

probability of measuring the optimal state without depending too much on the input size. The application of our AC definition (Chapter 3) also pointed out that the evolution does not undergo an anti-crossing; on the contrary, increasing the size smoothes then evolution. Eventually, in a last section, we developed higher-order coefficients of the perturbation theory for the MaxCut problem. We observe in the first orders that small cycles seem to affect the quantum evolution faster than large ones, and even and odd cycles have opposite effects on the energy. This leads us to believe that a quantum evolution is initially more affected by small local structures.

In the second part, we turn to the study of quantum annealing complexity. In QA, the running time is an arbitrary parameter, making it a good candidate for an approximation metaheuristic. In fact, one of the most famous variational quantum circuit for approximate optimization, QAOA, is inspired by analog evolution. These metaheuristics have recently received increasing attention as they are suitable for NISQ computers thanks to their finite runtime. In the original work of QAOA, the authors used the MaxCut problem on cubic graphs to assess the performance of single-layer QAOA and proved that it reaches an approximation ratio of 0.6925. The number of layers is the analog of the running time in QA. A question arises: How does QA behave at finite short time compared to QAOA? Prior to this thesis, it had not even been proven that its performance was strictly superior to that of the random guess. That is why, in Chapter 5, we developed new theoretical tools to answer this question. To use combinatorial arguments similar to the proof of QAOA, we needed a relaxed notion of locality in the continuous setting. To this purpose, we **developed a Lieb-Robinson like bound that allows us to study QA locally**. Our method can be applied to a wide range of combinatorial graph problems as long as the input is regular. We numerically evaluated the LR-type bound for the MaxCut and Maximum Independent Set problems over 3-regular graphs. It resulted that **QA as a local algorithm achieves a 0.5933 approximation ratio for MaxCut** and 0.31 for MIS. Both performances exceed the random guess value of 0.5 and 0.25 respectively. To put this into perspective, we analyzed the tightness of our LR bound to conclude that QA probably performs even better and hinted a new approach that could improve approximation ratios.

In Chapter 6, we formalized the idea suggested at the end of the previous chapter and developed it in the case of the MaxCut problem over cubic graphs. We introduced a parametrized version of QA to give us a degree of freedom that allowed us to tighten the analysis. The previously known LR bounds reach a sufficiently low numerical value for the approximation ratio if the local structure considered has a radius of around 70. With the available classical computational power, only a radius of 3 seems reasonable in practice. To meet this constraint and make the most of the new approach developed, **we have derived a new LR bound tailored to the problem, which achieves unprecedented small numerical values**. A careful enumeration of all balls of radius 3 around an edge of a cubic graph combined with an optimization of the introduced parameter yielded a **proven new ratio of 0.7020 with a 1-local analysis of QA**. This ratio outperforms all known local

algorithms of MaxCut over cubic graphs. In addition, we also argued that this value is almost tight for this approach. Eventually, we suggested some directions for improving the ratio by optimizing the schedule and some directions for generalizing the approach.

Lastly, we ended our work with the exploratory Chapter 7, which gathers **attempts at a state-dependent LR bound** that could improved the scope of our method. It also provides arguments on the worst-case neighborhood for an edge, namely the double binary tree. Indeed, this configuration maximizes the number of connections with the rest of the graph. Moreover, in random regular graph, all edges have a local neighborhood that is tree-like. Therefore, evaluating the edge energy inside a double binary tree would give a good insight into the optimal performance of QA for MaxCut on random regular graphs. **We developed an algorithm to reduce the size of the Hilbert space thanks to the numerous symmetries of the double binary tree.** This allows us to efficiently access edge energy values for tree of depth 1,2 and 3. To finally confirm that the values reached by the binary tree is tight for QA, we constructed 3-regular graphs that reach this value as an approximation ratio.

Further work

The work we have undertaken in this thesis opens new doors for further results. As mentioned, direct applications of the tools developed to other combinatorial problems are potential direct directions.

With regard to the LR bound, to improve numerical results, we have suggested that the main difficulty is now to derive a state-dependent LR bound. Neglecting the initial state is costly. In Chapter 7, we developed our first ideas in this direction, but some ingredients are still missing to finalize the construction. Furthermore, with the exception of an attempt with a cubic scheduling function, the approximation ratio in this thesis are developed with a linear schedule. It would be interesting to investigate this further to see the potential of short time QA. For example, work on optimal control of [Brady *et al.* 2021] could help derive more advanced schedules. A reverse annealing approach also seems promising and would deserve more analytical study to support it [Crosson & Lidar 2021]. Regarding the application of the LR limit we derived, we could consider a practical implementation of the quantum circuit for Hamiltonian simulation proposed in [Haah *et al.* 2021].

As far as avoided level crossings are concerned, there are still some misunderstandings about the definition. We define it by a large derivative of the instantaneous eigenstates at stake at the crossing point. However, it seems that there are Hamiltonians for which the gap reduces exponentially fast with the size of the input without degrading too much the performance of adiabatic quantum evolution, in the sense that the probability of obtaining the optimum remains reasonable. This striking observation leads us to consider an adapted version of the adiabatic theorem for combinatorial optimization that relaxes the dependence on the minimum gap for

other spectral quantities. The idea would be to design a more restrictive condition on execution time to achieve a constant probability of measuring the optimal solution.

Finally, the ideal result would be to prove a certain guarantee on the quality of the solution derived from a continuous quantum evolution in polynomial time. By allowing an execution time that scales polynomially with the size of the input, we expect a fraction of the evolved state to transition to higher-energy states. Can we quantify which fraction jumps to which energy level? We could certainly start by studying the probability of a non-adiabatic Landau-Zener transition in polynomial time.

Appendices

Some interesting features of the k -clique problem

Contents

A.1 Influence of the driver graph	173
A.2 Some property of H_0	175
A.3 Lower bound on the minimum gap	176

In this appendix, we provide results on the maximum-weight k -clique problem. Some of them were derived during the internship preceding the thesis.

A.1 Influence of the driver graph

In this section, we want to observe the influence of the driver graph. Recall from Sec. 2.2.4 that we are restricting the Hilbert space to bit-strings of Hamming weight k . The mixing Hamiltonian $H_0 = -\sum_{\langle a,b \rangle \in G_{\text{driver}}} \text{SWAP}^{(a,b)}$ stabilizes this reduced space by performing only swaps between qubits. For a bit-string x , we call G_x the graph induced by nodes i such that $x_i = 1$. $H_1|x\rangle = C(x)|x\rangle$, where $C(x)$ counts the number of missing edge in G_x . This encoding is the one suggested in [Childs *et al.* 2002]. The authors provided a numerical analysis of the median running time needed to achieve a ground-state probability above $\frac{1}{8}$. They ran the analysis over 500 random graphs for size ranging from 7 to 18. They observed a quadratic scaling of the median runtime when using a complete driver graph K_n .

We tackle here the influence of the driver graph on the median runtime. For example, we look at the star graph S_n and the path graph P_n . In Fig. A.1, we plot the result obtain in each case. It is rather impressive how the driver graph seems to play a significant role in the median efficiency of the continuous evolution. We recover the result from the original work in (a). For the star graph (b), we observe a linear tendency with the input size and for the path graph we need a logarithmic scale to plot the data. The main observation is the spectacular efficiency of the star graph as a driver over the complete graph. One could expect that a more dense mixing Hamiltonian would help the evolution and here we observe the contrary. The path graph however seems to be a quite poor choice as a driver graph.

To run the evolution, we need to start from the ground-state of H_0 . For the complete driver graph and the star graph they have the following expressions re-

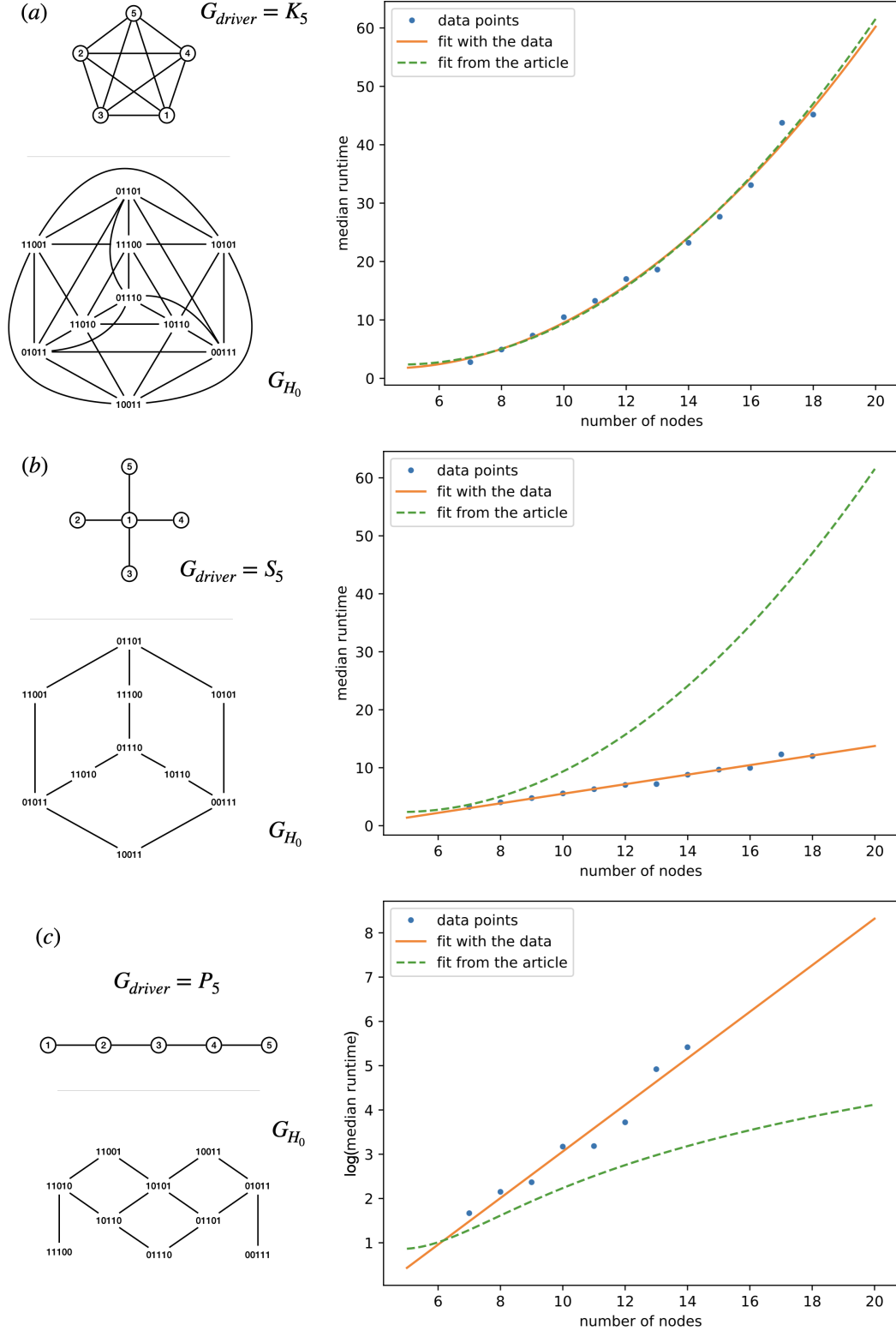


Figure A.1: On the left, G_{driver} with the corresponding graph G_{H_0} with adjacency matrix $-H_0$ with $n = 5$ and $k = 3$ and on the right the median time to reach a ground-state probability of $\frac{1}{8}$ over 500 random graphs with size from 7 to 18. (a) for the complete graph, (b) for the star graph and (c) the path graph in log scale. Green dotted curve is the fit from [Childs *et al.* 2002], in orange our data fit.

spectively :

$$|\psi_0^K\rangle = \binom{n}{k}^{-\frac{1}{2}} \sum_{x, |x|=k} |x\rangle$$

$$|\psi_0^S\rangle = \frac{1}{\sqrt{2}} \begin{pmatrix} \dots \\ \frac{1}{\sqrt{\binom{n-1}{k}}} \\ \dots \\ \frac{1}{\sqrt{\binom{n-1}{k-1}}} \\ \dots \end{pmatrix}$$

For the path graph, we do not have a closed form and we need to compute it at the beginning of the algorithm. This is why the data in Fig. A.1 for the path graph stops at $n = 14$.

To explain these main differences in the median runtime scaling, in the next section we derive some of the spectral quantity that can play a role in the efficiency of AQC.

A.2 Some property of H_0

We call G_{H_0} the graph induced by the adjacency matrix $-H_0$, i.e. the one in Fig. A.1 on the left below the driver graph. G_{H_0} represents the graph of the search space and the relation between the possible solution. At first glance, it would be natural to prefer the complete driver graph as it generates the most dense G_{H_0} so it should be easier to amplify the amplitude of the optimal solution, i.e. the clique. However, it numerically appears that the star driver generates a better G_{H_0} for solving the problem. Let us introduce two spectral quantities of a graph :

- λ_{H_0} is the principal eigenvector of G_{H_0} ;
- γ_{H_0} is the principal ratio of G_{H_0} , it is the ratio of the largest component of $|\psi_0\rangle$ with the smallest. It reflects the regularity of the graph ($=1$ if G_{H_0} is regular to $=e^n$ if highly irregular like the kite graph [Zhang 2021]);

We suppose also that $k < n/2$, otherwise in the rest of this appendix, many k must be replaced by $n - k$. In Table A.1, we sum up some usual characterization about the graph G_{H_0} along with the two previously introduced spectral properties.

Conjecture : The median running time scales like the product $\lambda_{H_0}\gamma_{H_0}$. H_0 needs a high regularity but not too dense. The star seems to have the better scaling, namely linear while the path is the worst with an exponential tendency. For the complete graph, it scales like kn , which for small instances is quadratic in n since $k \simeq n/2$ at small n .

G_{driver}	Complete	Star	Path
λ_{H_0}	$k(n-k)$	$\sqrt{k(n-k)}$	$2 \cos(\frac{k+1}{n+1} \frac{\pi}{2}) \frac{\sin(\frac{k\pi}{2(n+1)})}{\sin(\frac{\pi}{2(n+1)})} \underset{n \rightarrow \infty}{\sim} 2k$
γ_{H_0}	1	$\sqrt{\frac{n-k}{k}}$	$\simeq 10^{-2^{k-1}} \binom{n+k}{k}^k$
diameter	k	$2k$	$k(n-k)$
$\frac{ E(G_{H_0}) }{ V(G_{H_0}) }$	$\frac{k(n-k)}{2}$	$\frac{k(n-k)}{n}$	$\frac{k(n-k)}{n}$
degree	$k(n-k)$	k and $n-k$	from 1 to $2k$

Table A.1: Quantities about graph G_{H_0} . For the path driver, the value of γ_{H_0} is only an estimation.

A.3 Lower bound on the minimum gap

Let $\mathbf{d}_{\max}(H_0)$ and d be respectively the maximal degree and the diameter of G_{H_0} . Inspired by the Appendix D in [Altshuler *et al.* 2010], one can derive a lower bound of the gap that depends on $\mathbf{d}_{\max}(H_0)$ and d .

Lemma A.1. *For a given H_0 with its associated $\mathbf{d}_{\max}(H_0)$ and d as defined above, a unique solution such that $E_0(1) = 0$ and $1 \leq E_i(1) \leq \binom{k}{2}$ for all $i > 0$ on average, the minimum gap is lower bounded as: $\Delta_{\min} \geq O((\mathbf{d}_{\max}(H_0)k^2)^{-d})$*

Proof. Let us first consider the case $s \geq \frac{2\mathbf{d}_{\max}(H_0)+1}{2\mathbf{d}_{\max}(H_0)+2}$. The diagonal elements of $H(s)$ are given by $H_{ii} = sE_i$ and the non-diagonal elements satisfy $\sum_{j \neq i} H_{ij} \leq -(1-s)\mathbf{d}_{\max}(H_0)$. Therefore, the Gershgorin circles have a radius of at most $(1-s)\mathbf{d}_{\max}(H_0)$ and the circle around the solution centered in 0, because $E_0 = 0$, is disjoint from the other circles around $sE_i \geq s$ if $s > 2(1-s)\mathbf{d}_{\max}(H_0)$. Thus, the eigenvalue gap between the two lowest is lower bounded as: $\Delta(s) > s - 2(1-s)\mathbf{d}_{\max}(H_0) \geq \frac{1}{2\mathbf{d}_{\max}(H_0)+2} \geq \Omega(\frac{1}{\mathbf{d}_{\max}(H_0)})$.

Let us now consider the case $s \leq \frac{2\mathbf{d}_{\max}(H_0)+1}{2\mathbf{d}_{\max}(H_0)+2}$. Let $Q = \frac{-H(s) + s\binom{k}{2}\mathbb{I}}{1-s} = A + \lambda(\binom{k}{2}\mathbb{I} - H_1)$, where $A = -H_0$ is the adjacency matrix of G_{H_0} and $\lambda = \frac{s}{1-s} \leq 2\mathbf{d}_{\max}(H_0) + 1$. Since every $E_i \leq \binom{k}{2}$, all elements of Q are positives. By the mixing properties of G_{H_0} , A^d has all elements above 1, thus this is also true for Q . Therefore, $\mu_0^d - \mu_1^d \geq 1$, where μ_0 and μ_1 are the 2 largest eigenvalues of Q . The eigenvalues of Q are upper bounded by the spectral radius which is upper bounded by the norm $\|Q\|_1 = \max_i \sum_j |Q_{ij}|$ so that for all i , $\mu_i \leq \lambda\binom{k}{2} + \mathbf{d}_{\max}(H_0)$. Furthermore we have $\mu_0^d - \mu_1^d \leq (\mu_0 - \mu_1)(\mu_0 + \mu_1)^{d-1}$ so that

$$\begin{aligned}
 \mu_0 - \mu_1 &\geq \frac{1}{(\mu_0 + \mu_1)^{d-1}} \\
 &\geq \frac{1}{(\lambda k(k-1) + 2\mathbf{d}_{\max}(H_0))^{d-1}} \\
 &\geq \frac{1}{(2\mathbf{d}_{\max}(H_0) \times k(k-1) + k(k-1) + 2\mathbf{d}_{\max}(H_0))^{d-1}}
 \end{aligned}$$

Finally, we have

$$\begin{aligned}\Delta(s) &= (1-s)(\mu_0 - \mu_1) \\ &\geq \frac{1}{(2\mathbf{d}_{\max}(H_0) + 2)(2\mathbf{d}_{\max}(H_0) \times k(k-1) + k(k-1) + 2\mathbf{d}_{\max}(H_0))^{d-1}} \\ \text{i.e. } \Delta(s) &\geq O((\mathbf{d}_{\max}(H_0)k^2)^{-d})\end{aligned}$$

□

From Lemma 3, we can only conclude that the algorithm has a run time upper bounded by a superpolynomial in the size of the problem for $k = O(\text{poly}(\log(n)))$ which is no improvement compare to classical algorithms that find k -clique. Also given Table A.1, we see that the path graph clearly has the worst scaling with a diameter that grows with $\mathcal{O}(n)$, while the complete and star driver have a diameter in $\mathcal{O}(k)$ only

MaxCut H_1 on H_0 eigenvectors

Contents

B.1 Proof of Proposition 4.1	179
-------------------------------------	------------

B.1 Proof of Proposition 4.1

In this appendix, we prove the proposition due to my supervisor Simon Martiel. Let restate it.

Proposition B.1. *If H_1 is the Hamiltonian encoding the MaxCut problem over a graph $G(V, E)$ of n nodes and H_0 is the standard bit-flip operator with eigenpairs $(|E_b\rangle, E_b)$ for a bit-string b where $E_b = -n + 2|b|$, $|b|$ being the Hamming weight of b , then*

$$\langle E_b | H_1 | E_{b'} \rangle = \frac{1}{2}$$

if and only if $G_{b \oplus b'}$ is exactly an edge. Where $G_{b \oplus b'}$ is the subgraph induced by nodes i such that $(b \oplus b')_i = 1$.

The eigenvectors $|E_b\rangle$ of H_0 correspond to states where qubit i is in state $|+\rangle = \frac{|0\rangle + |1\rangle}{2}$ if $b_i = 0$ and $|-\rangle = \frac{|0\rangle - |1\rangle}{2}$ if $b_i = 1$. In the computational basis it is written as:

$$|E_b\rangle = \frac{1}{\sqrt{2^n}} \sum_{x \in \{0,1\}^n} (-1)^{b \cdot x} |x\rangle$$

Let b and b' be two bit-strings, we have that $H_1|x\rangle = C(x)|x\rangle$ for any classical state $|x\rangle$ where $C(x)$ is the classical cost function for the MaxCut problem, counting minus the number of edges across the bipartition x . Recall that we are in the minimization setting so there is a minus absorbed by the cost value, i.e. $C(x) \in [-|E(G)|, 0]$. We can then write :

$$\begin{aligned} \langle E_b | H_1 | E_{b'} \rangle &= \frac{1}{2^n} \sum_{x \in \{0,1\}^n} \sum_{y \in \{0,1\}^n} (-1)^{b \cdot x} (-1)^{b' \cdot y} \langle x | H_1 | y \rangle \\ &= \frac{1}{2^n} \sum_{x \in \{0,1\}^n} (-1)^{(b \oplus b') \cdot x} C(x) \\ &= \langle E_{b \oplus b'} | H_1 | E_{0^n} \rangle \end{aligned}$$

where we recognize the ground-state $|E_{0^n}\rangle$ of H_0 . Therefore we can focus on proving that $\langle E_b|H_1|E_{0^n}\rangle = \frac{1}{2}$ if and only if G_b is exactly an edge and 0 otherwise. Let consider a bit-string b of Hamming weight k , we can write :

$$\begin{aligned}\langle E_b|H_1|E_{0^n}\rangle &= \frac{1}{2^n} \sum_{x \in \{0,1\}^n} (-1)^{b \cdot x} C(x) \\ &= \frac{1}{2^n} \sum_{y \in \{0,1\}^{n-k}} \sum_{x \in \{0,1\}^k} (-1)^{|x|} \left(C_{G_{\bar{b}}}(y) + C_{G_b}(x) + D_{G,b}(x, y) \right)\end{aligned}$$

where $\bar{b}$ is the bit-string obtained by bit-flipping every bits in b , C_G is the MaxCut cost function associated to graph G and $D_{G,b}(x, y)$ is (minus) the number of edges cut across G_b and $G_{\bar{b}}$ given bipartition x of G_b and y of $G_{\bar{b}}$. In the two sums above, the sum over y ranges over all possible bipartitions of $G_{\bar{b}}$, while the sum over x ranges over all possible bipartitions of G_b .

Since $\sum_{x \in \{0,1\}^k} (-1)^{|x|} = 0$, the term $C_{G_{\bar{b}}}(y)$ vanishes. Then for each edge $\langle i, j \rangle$ between G_b and $G_{\bar{b}}$ cut by bipartition x and y , i.e. without loss of generality, if i belongs to G_b and j to $G_{\bar{b}}$, it means that $x_i \neq y_j$, it exists another bipartition x' and y' that also cuts edge $\langle i, j \rangle$ with $|x'| = |x| \pm 1$. For example, x' is x with $x'_i = \bar{x}_i$ and y' is y with $y'_j = \bar{y}_j$. So the term $D_{G,b}(x, y)$ also vanishes. We are left with :

$$\begin{aligned}\langle E_b|H_1|E_{0^n}\rangle &= \frac{1}{2^n} \sum_{y \in \{0,1\}^{n-k}} \sum_{x \in \{0,1\}^k} (-1)^{|x|} C_{G_b}(x) \\ &= \frac{1}{2^k} \sum_{x \in \{0,1\}^k} (-1)^{|x|} C_{G_b}(x)\end{aligned}$$

Now if $k = 0$ or $k = 1$, $C_{G_b}(x) = 0$. Assume that $k \geq 3$. For all edges $\langle i, j \rangle$ in G_b , only half of the bit-strings $x \in \{0,1\}^k$ will contribute to the sum over x , in fact every time $x_i \neq x_j$. Among them, half has a positive contribution and the other half a negative contribution, bringing the total sum to zero (see Table B.1)

$x = x_1 x_i x_j$	$(-1)^{ x }$	$C_{G_b}(x)$
000	+1	0
001	-1	-1
010	-1	-1
011	+1	0
100	-1	0
101	+1	-1
110	+1	-1
111	-1	0

Table B.1: Example of the values in the sum over x for $k = 3$. Only rows shaded in light grey have a contribution in the sum.

For the case $k = 2$, the above argument does not work anymore. G_b is either two isolated nodes or exactly an edge of G . In the first case $C_{G_b}(x) = 0$ for any x .

In the second one, as the sum in x is over the set $\{00, 01, 10, 11\}$, only terms 01 and 10 has a non-zero value of $C_{G_b}(x) = -1$, and they contribute positively to the sum because the Hamming weight is odd. Thus, we obtain :

$$\begin{aligned}\langle E_b | H_1 | E_{0^n} \rangle &= \frac{1}{2^k} \sum_{x \in \{0,1\}^k} (-1)^{|x|} C_{G_b}(x) \\ &= \frac{1}{2^k} \times (-2) \times (-1) \\ &= \frac{1}{2}\end{aligned}$$

if and only if G_b is exactly an edge. If not, the quantity is zero. □

By adapting the proof, a similar result can be proved for the asymmetric H_1 :

Claim 6. If H_1 is the Hamiltonian encoding the MaxCut problem over a graph $G(V, E)$ of n nodes where node labeled 1 is fixed to remove the symmetry and H_0 is the standard bit-flip operator over $n - 1$ qubits with eigenpairs $(|E_b\rangle, E_b)$, then

$$\langle E_b | H_1 | E_{0^n} \rangle = \frac{1}{2}$$

if and only if $(G - 1)_b$ is exactly an edge or exactly one of node 1 neighbors. Where $(G - 1)_b$ is the subgraph of G induced by nodes i such that $b_i = 1$ after deleting node 1.

We can see that there are exactly the same amount of bit-string b for which the quantity equals $1/2$, which is the total number of edges in the input graph $|E(G)|$.

Technical results for the LR bound based ratio

Contents

C.1 Minimization of Equation 6.6	183
C.2 Proof of Equation 6.10	184
C.3 Nested integrals	184

C.1 Minimization of Equation 6.6

Let us study the following function with two variables:

$$\forall x \geq 0, y \geq 0 | 4x + 3y \leq 1, f(x, y) = \frac{ax + b(4x + 3y) + c(\frac{3}{2} - 5x - 3y)}{\frac{3}{2} - x - y}$$

where $a = \langle O_X \rangle_{\mathcal{G}}^{\Omega_1}$, $b = \langle O_X \rangle_{\mathcal{G}}^{\Omega_2}$ and $c = \langle O_X \rangle_{\mathcal{G}}^{\Omega_3}$. Empirically we suppose that $a \leq b \leq c$ and we will see later that this assumption is verified. Let x, y be positive number such that $4x + 3y \leq 1$. Then we have that $x + y \leq \frac{1}{3}$ and $x \leq \frac{1}{4}$. The function f can be rewritten in three parts:

$$f(x, y) = \frac{\frac{3}{2}c}{\frac{3}{2} - (x + y)} - \frac{3(c - b)(x + y)}{\frac{3}{2} - (x + y)} - \frac{(2c - a - b)x}{\frac{3}{2} - (x + y)}$$

If $x = y = 0$, then $f(0, 0) = c$. Let us see the condition on which f can only increase if $x + y > 0$. The first term can increase at most by $c - c\frac{\frac{3}{2}}{\frac{3}{2}-1} = \frac{2}{7}c$. The second term can decrease at most by $\frac{c-b}{\frac{3}{2}-\frac{1}{3}} = (c-b)\frac{6}{7}$. The last term can decrease by at most $\frac{\frac{1}{4}(2c-a-b)}{\frac{3}{2}-\frac{1}{3}} = (2c-a-b)\frac{3}{14}$. Thus $f(0, 0)$ is the minimum if $\frac{2}{7}c \geq (c-b)\frac{6}{7} + (2c-a-b)\frac{3}{14}$. Therefore, we can derive the following condition to satisfy to have $f(0, 0)$ as the minimum:

$$\begin{aligned} c &\geq 3(c-b) + \frac{3}{4}(2c-a-b) \\ \Rightarrow 3.5c &\leq 3.75b + 0.75a \end{aligned}$$

For example with $a \geq 0.5$, $b \geq 0.57$ and $c \leq 0.71$ the assumption and the condition are satisfied.

C.2 Proof of Equation 6.10

We are working with a Hamiltonian of general form $H(t) = \sum_{j \in V(\mathbf{G})} h_j(t) \gamma_j$ and for any unitary A supported on S (outside a certain region around one node X), we want to show Equation 6.10 in the time-dependent regime. We follow exactly the same steps of [Wang & Hazzard 2020] but with function h_j that depends on the time. First let us look at the derivative of $\gamma_X^A(t)$:

$$\frac{d[\gamma_X(t), A]}{dt} = -i \sum_{v: \langle Xv \rangle \in \mathbf{G}} h_v(t) [(U_T^G)^\dagger [\gamma_X, \gamma_v] U_T^G, A]$$

Then we define $\tau_A(t) = \hat{U}^\dagger \gamma_X^A(t) \hat{U}$ where $\hat{U}$ is the unitary that is solution of $i \frac{d\hat{U}}{dt} = -\sum_{v: \langle Xv \rangle \in \mathbf{G}} h_v(t) \gamma_v(t) \hat{U}$. That way, we have $\|\tau_A(T)\| = \|\gamma_X^A(T)\|$ and the derivative of τ_A is given by:

$$\dot{\tau}_A(t) = -i \sum_{v: \langle Xv \rangle \in \mathbf{G}} h_v(t) \hat{U}^\dagger [\gamma_X(t), [\gamma_v(t), A]] \hat{U}$$

Now we can proceed as follow:

$$\begin{aligned} \|\gamma_X^A(T)\| - \|\gamma_X^A(0)\| &= \|\tau_A(T)\| - \|\tau_A(0)\| \\ &\leq \int_0^T \|\dot{\tau}_A(t')\| dt' \\ &\leq \sum_{v: \langle Xv \rangle \in \mathbf{G}} \int_0^T h_v(t) \|[\gamma_X(t), [\gamma_v(t), A]]\| dt \end{aligned}$$

□

C.3 Nested integrals

In this appendix, we detail the computation of the nested integrals that play an important role in the LR-bound. To tackle this derivation, we introduce for all $k \in \mathbb{N}^*$:

$$I_{2k}(x) = \int_0^x 1 - u_1 \int_0^{u_1} u_2 \dots \int_0^{u_{2k-1}} u_{2k} du_{2k} \dots du_2 du_1$$

and

$$I_{2k+1}(x) = \int_0^x 1 - u_1 \int_0^{u_1} u_2 \dots \int_0^{u_{2k}} 1 - u_{2k+1} du_{2k+1} \dots du_2 du_1$$

polynomials in x define in $[-1, 1]$. The goal is to compute these polynomials at $x = 1$, which is nothing else than the sum of the polynomial coefficients.

Even case $I_{2k}(x)$: The highest degree of this polynomial is $4k$ as there are $2k$ integrals and there is always a way to choose a u_j to integrate so the $2k$ u_j bring the total degree to $4k$. The highest order coefficient is straightforward $\frac{(-1)^k}{2^{2k}(2k)!}$. The

least order term is the one where we choose the minimum number of u_j to integrate which is k . This observation brings the least order term to be of degree $3k$ with coefficient $\prod_{l=2}^k \frac{1}{3l(3l-1)}$. In general, there exist positive coefficients $a_j(k)$ to express $I_{2k}(x)$ as a polynomial:

$$I_{2k}(x) = \sum_{j=0}^k a_j(k)(-1)^j x^{3k+j} \quad (\text{C.1})$$

$$= x^{3k} \sum_{j=0}^k a_j(k)(-x)^j \quad (\text{C.2})$$

and we define $a_0(0) = 1$. With these notations, the quantity of interest is $I_{2k}(1) = \sum_{j=0}^k (-1)^j a_j(k)$. To find a recurrence relation first notice that $I_{2k}(x) = \int_0^x 1 - u_1 \int_0^{u_1} u_2 I_{2k-2}(u_2) du_2 du_1$ and develop:

$$\begin{aligned} I_{2k}(x) &= \int_0^x 1 - u_1 \int_0^{u_1} u_2 * I_{2k-2}(u_2) du_2 du_1 \\ &= \sum_{j=0}^{k-1} a_j(k-1)(-1)^j \int_0^x 1 - u_1 \int_0^{u_1} u_2^{3k+j-2} du_2 du_1 \\ &= \sum_{j=0}^{k-1} a_j(k-1)(-1)^j \int_0^x (1 - u_1) \frac{u_1^{3k+j-1}}{3k+j-1} du_1 \\ &= \sum_{j=0}^{k-1} a_j(k-1)(-1)^j \left[\frac{x^{3k+j}}{(3k+j-1)(3k+j)} - \frac{x^{3k+j+1}}{(3k+j-1)(3k+j+1)} \right] \\ &= x^{3k} \sum_{j=0}^{k-1} a_j(k-1)(-1)^j \left[\frac{x^j}{(3k+j-1)(3k+j)} - \frac{x^{j+1}}{(3k+j-1)(3k+j+1)} \right] \\ &= x^{3k} \sum_{j=0}^{k-1} a_j(k-1)(-1)^j \frac{x^j}{(3k+j-1)(3k+j)} \\ &\quad - x^{3k} \sum_{j=1}^k a_{j-1}(k-1)(-1)^{j-1} \frac{x^j}{(3k+j-2)(3k+j)} \\ &= \frac{a_0(k-1)}{3k(3k-1)} x^{3k} + \frac{(-1)^k a_{k-1}(k-1)}{4k(4k-2)} x^{4k} \\ &\quad + x^{3k} \sum_{j=1}^{k-1} \frac{(3k+j-2)a_j(k-1) + (3k+j-1)a_{j-1}(k-1)}{(3k+j-2)(3k+j-1)(3k+j)} (-x)^j \end{aligned} \quad (\text{C.3})$$

In the last line we can identify the $a_j(k)$ coefficients:

$$a_0(k) = \frac{a_0(k-1)}{3k(3k-1)} \quad (\text{C.4})$$

$$a_j(k) = \frac{a_j(k-1)}{(3k+j-1)(3k+j)} + \frac{a_{j-1}(k-1)}{(3k+j-2)(3k+j)} \quad \text{for } j \in [1, \dots, k-1] \quad (\text{C.5})$$

$$a_k(k) = \frac{a_{k-1}(k-1)}{4k(4k-2)} \quad (\text{C.6})$$

We recover the higher and least order coefficients mentioned above. So we can try to compute $I_{2k}(1)$:

$$\begin{aligned} I_{2k}(1) &= \sum_{j=0}^k (-1)^j a_j(k) \\ &= \sum_{j=1}^{k-1} (-1)^j \left[\frac{a_j(k-1)}{(3k+j-1)(3k+j)} + \frac{a_{j-1}(k-1)}{(3k+j-2)(3k+j)} \right] \\ &\quad + \frac{a_0(k-1)}{3k(3k-1)} + (-1)^k \frac{a_{k-1}(k-1)}{4k(4k-2)} \\ &= \sum_{j=1}^{k-1} (-1)^j \frac{a_j(k-1)}{(3k+j-1)(3k+j)} + \sum_{j=0}^{k-2} (-1)^{j+1} \frac{a_j(k-1)}{(3k+j-1)(3k+j+1)} \\ &\quad + \frac{a_0(k-1)}{3k(3k-1)} + (-1)^k \frac{a_{k-1}(k-1)}{4k(4k-2)} \\ &= \sum_{j=0}^{k-1} \frac{(-1)^j a_j(k-1)}{(3k+j-1)(3k+j)(3k+j+1)} \end{aligned} \quad (\text{C.7})$$

Computing recursively the coefficients $(a_j(k))_j$ allows us to evaluate the nested integrals of interest. At the end of this appendix, we develop the numerical analysis of it.

We can still have a loose upper bound (sufficient for Corollary 6.1) because the

$a_j(k)$ are positives, by looking at $I_{2k}(-1)$:

$$\begin{aligned}
 I_{2k}(-1) &= \sum_{j=0}^k a_j(k) \\
 &= \frac{a_0(k-1)}{3k(3k-1)} + \sum_{j=1}^{k-1} \left[\frac{a_j(k-1)}{(3k+j-1)(3k+j)} + \frac{a_{j-1}(k-1)}{(3k+j-2)(3k+j)} \right] \\
 &\quad + \frac{a_{k-1}(k-1)}{4k(4k-2)} \\
 &= \frac{a_0(k-1)}{3k(3k-1)} + \sum_{j=1}^{k-1} \frac{a_j(k-1)}{(3k+j-1)(3k+j)} + \sum_{j=0}^{k-2} \frac{a_j(k-1)}{(3k+j-1)(3k+j+1)} \\
 &\quad + \frac{a_{k-1}(k-1)}{4k(4k-2)} \\
 &= \sum_{j=0}^{k-1} \frac{6k+2j+1}{(3k+j-1)(3k+j)(3k+j+1)} a_j(k-1) \tag{C.8}
 \end{aligned}$$

$$\begin{aligned}
 &\leq \frac{6k+1}{(3k-1)3k(3k+1)} I_{2k-2}(-1) \\
 &\leq I_2(-1) \prod_{l=2}^k \frac{6l+1}{(3l-1)3l(3l+1)} \\
 &\leq \frac{6^{k+1}k!}{(3k+1)!} \tag{C.9}
 \end{aligned}$$

Odd case $I_{2k+1}(x)$: The higher degree of this polynomial is $4k+2$ as there are $2k+1$ integrals and there is a way to always choose a u_j to integrate so the $2k+1$ u_j bring the total degree to $4k+2$. The least order term is the one where we choose the minimum number of u_j to integrate which is k . This observation brings the least order term to be of degree $3k+1$. In general, there exist positive coefficients $b_j(k+1)$ to express $I_{2k+1}(x)$ as a polynomial:

$$I_{2k+1}(x) = \sum_{j=0}^{k+1} b_j(k+1)(-1)^j x^{3k+j+1} \tag{C.10}$$

$$= x^{3k+1} \sum_{j=0}^{k+1} b_j(k+1)(-x)^j \tag{C.11}$$

$$\begin{aligned}
I_{2k+1}(x) &= \int_0^x 1 - u_1 \int_0^{u_1} u_2 * I_{2k-1}(u_2) du_2 du_1 \\
&= \dots \\
&= \frac{b_0(k)}{3k(3k+1)} x^{3k+1} + x^{3k+1} \sum_{j=1}^k \frac{(3k+j-1)b_j(k) + (3k+j)b_{j-1}(k)}{(3k+j+1)(3k+j-1)(3k+j)} (-x)^j \\
&\quad + \frac{(-1)^{k+1}b_k(k)}{4k(4k+2)} x^{4k+2} \quad (C.12)
\end{aligned}$$

We can identify the $b_j(k)$ coefficients:

$$b_0(k+1) = \frac{b_0(k)}{3k(3k+1)} \quad (C.13)$$

$$b_j(k+1) = \frac{b_j(k)}{(3k+j+1)(3k+j)} + \frac{b_{j-1}(k)}{(3k+j+1)(3k+j-1)} \quad \text{for } j \in [1, \dots, k] \quad (C.14)$$

$$b_{k+1}(k+1) = \frac{b_k(k)}{4k(4k+2)} \quad (C.15)$$

Those coefficients allows us to compute the exact value of the nested integral. Similarly to the even case, we can upper bound the integral of interest like this:

$$\begin{aligned}
I_{2k+1}(1) &= \sum_{j=0}^{k+1} (-1)^j b_j(k+1) \\
&= \sum_{j=0}^k \frac{(-1)^j b_j(k)}{(3k+j)(3k+j+1)(3k+j+2)} \quad (C.16)
\end{aligned}$$

$$I_{2k+1}(1) \leq I_{2k+1}(-1) \leq \frac{6^{k+2}(k+1)!}{(3k+2)!} \quad (C.17)$$

In practice, a numerical study is enough as we are interested in the numerical value. Using the derived coefficients above, it appears that it follows something like $I_{2k} \simeq \frac{\text{num}(I_{2k})}{(4k)!}$ and $I_{2k+1} \simeq \frac{\text{num}(I_{2k+1})}{(4k+2)!}$ where both $\text{num}(I_{2k+1})$ and $\text{num}(I_{2k})$ grow like $e^{\mathcal{O}(k^2)}$. A numerical fit gives a tendency function with an almost 1 correlation factor, see Fig. C.1.

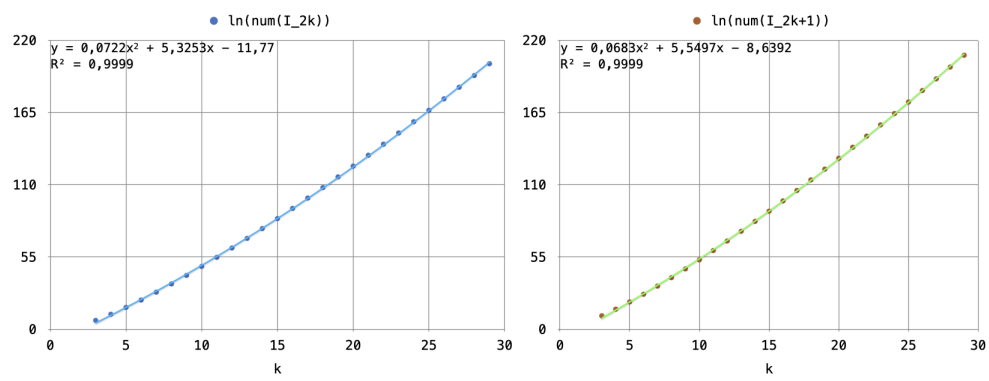


Figure C.1: Fit curves for the numerator logarithm of the nested integral for even (left) and odd (right) iteration step.

Bibliography

- [Aharonov *et al.* 2004] D. Aharonov, W. van Dam, J. Kempe, Z. Landau, S. Lloyd and O. Regev. *Adiabatic quantum computation is equivalent to standard quantum computation*. In 45th Annual IEEE Symposium on Foundations of Computer Science, pages 42–51, 2004. (Cited on pages [22](#) and [35](#).)
- [Al-Jazari 1974] Ibn al-Razzaz Al-Jazari. The book of knowledge of ingenious mechanical devices. D. Reidel Publishing Company, 1974. (Cited on pages [1](#) and [14](#).)
- [Albash & Lidar 2018] Tameem Albash and Daniel A. Lidar. *Adiabatic quantum computation*. Rev. Mod. Phys., vol. 90, page 015002, Jan 2018. (Cited on pages [23](#), [31](#), [32](#) and [69](#).)
- [Alimonti & Kann 2000] Paola Alimonti and Viggo Kann. *Some APX-completeness results for cubic graphs*. Theoretical Computer Science, vol. 237, no. 1, pages 123–134, 2000. (Cited on page [50](#).)
- [Altshuler *et al.* 2010] Boris Altshuler, Hari Krovi and Jérémie Roland. *Anderson localization makes adiabatic quantum optimization fail*. Proceedings of the National Academy of Sciences, vol. 107, no. 28, pages 12446–12450, 2010. (Cited on pages [23](#), [52](#), [57](#), [86](#), [104](#) and [176](#).)
- [Amin & Choi 2009] M. H. S. Amin and V. Choi. *First-order quantum phase transition in adiabatic quantum computation*. Phys. Rev. A, vol. 80, page 062326, Dec 2009. (Cited on pages [23](#), [52](#), [58](#), [86](#) and [92](#).)
- [Anderson 1958] P. W. Anderson. *Absence of Diffusion in Certain Random Lattices*. Phys. Rev., vol. 109, pages 1492–1505, Mar 1958. (Cited on pages [5](#), [17](#), [36](#) and [51](#).)
- [Andrist *et al.* 2023] Ruben S. Andrist, Martin J. A. Schuetz, Pierre Minssen, Romina Yalovetzky, Shouvanik Chakrabarti, Dylan Herman, Niraj Kumar, Grant Salton, Ruslan Shaydulin, Yue Sun, Marco Pistoia and Helmut G. Katzgraber. *Hardness of the maximum-independent-set problem on unit-disk graphs and prospects for quantum speedups*. Phys. Rev. Res., vol. 5, page 043277, Dec 2023. (Cited on page [38](#).)
- [Apers *et al.* 2022] Simon Apers, Shantanav Chakraborty, Leonardo Novo and Jérémie Roland. *Quadratic Speedup for Spatial Search by Continuous-Time Quantum Walk*. Phys. Rev. Lett., vol. 129, page 160502, Oct 2022. (Cited on page [23](#).)
- [Apolloni *et al.* 1989] B. Apolloni, C. Carvalho and D. de Falco. *Quantum stochastic optimization*. Stochastic Processes and their Applications, vol. 33, no. 2, pages 233–244, 1989. (Cited on pages [8](#), [19](#) and [36](#).)

- [Arai *et al.* 2023] Shunta Arai, Hiroki Oshiyama and Hidetoshi Nishimori. *Effectiveness of quantum annealing for continuous-variable optimization*. Phys. Rev. A, vol. 108, page 042403, Oct 2023. (Cited on page 23.)
- [Ausiello *et al.* 1999] Giorgio Ausiello, M. Protasi, A. Marchetti-Spaccamela, G. Gambosi, P. Crescenzi and V. Kann. Complexity and approximation: Combinatorial optimization problems and their approximability properties. Springer-Verlag, Berlin, Heidelberg, 1st édition, 1999. (Cited on page 31.)
- [Banks *et al.* 2024] Robert J. Banks, Dan E. Browne and P.A. Warburton. *Rapid quantum approaches for combinatorial optimisation inspired by optimal state-transfer*. Quantum, vol. 8, page 1253, February 2024. (Cited on page 23.)
- [Bapst & Semerjian 2012] Victor Bapst and Guilhem Semerjian. *On quantum mean-field models and their quantum annealing*. Journal of Statistical Mechanics: Theory and Experiment, vol. 2012, no. 06, page P06007, 2012. (Cited on page 104.)
- [Barahona *et al.* 1988] Francisco Barahona, Martin Grötschel, Michael Jünger and Gerhard Reinelt. *An Application of Combinatorial Optimization to Statistical Physics and Circuit Layout Design*. Oper. Res., vol. 36, no. 3, page 493–513, jun 1988. (Cited on pages 7 and 19.)
- [Basso *et al.* 2022] Joao Basso, Edward Farhi, Kunal Marwaha, Benjamin Villalonga and Leo Zhou. *The Quantum Approximate Optimization Algorithm at High Depth for MaxCut on Large-Girth Regular Graphs and the Sherrington-Kirkpatrick Model*. In François Le Gall and Tomoyuki Morimae, editors, 17th Conference on the Theory of Quantum Computation, Communication and Cryptography (TQC 2022), volume 232 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 7:1–7:21, Dagstuhl, Germany, 2022. Schloss Dagstuhl – Leibniz-Zentrum für Informatik. (Cited on pages 137, 155, 160, 161 and 165.)
- [Baynes 1875] Thomas Spencer Baynes. Encyclopedia britannica, volume 1. Charles Scribner’s Sons, 9 édition, 1875. (Cited on pages 2 and 14.)
- [Benioff 1980] Paul Benioff. *The computer as a physical system: A microscopic quantum mechanical Hamiltonian model of computers as represented by Turing machines*. Journal of statistical physics, vol. 22, pages 563–591, 1980. (Cited on pages 3, 15 and 19.)
- [Berman & Karpinski 1999] Piotr Berman and Marek Karpinski. *On Some Tighter Inapproximability Results (Extended Abstract)*. In Jirí Wiedermann, Peter van Emde Boas and Mogens Nielsen, editors, Automata, Languages and Programming, pages 200–209, Berlin, Heidelberg, 1999. Springer Berlin Heidelberg. (Cited on page 50.)

- [Bernstein & Vazirani 1997] Ethan Bernstein and Umesh Vazirani. *Quantum Complexity Theory*. SIAM Journal on Computing, vol. 26, no. 5, pages 1411–1473, 1997. (Cited on pages 9, 20 and 31.)
- [Berry *et al.* 2006] Dominic W. Berry, Graeme Ahokas, Richard Cleve and Barry C. Sanders. *Efficient Quantum Algorithms for Simulating Sparse Hamiltonians*. Communications in Mathematical Physics, vol. 270, no. 2, page 359–371, December 2006. (Cited on page 64.)
- [Blekos *et al.* 2023] Kostas Blekos, Dean Brand, Andrea Ceschini, Chiao-Hui Chou, Rui-Hao Li, Komal Pandya and Alessandro Summer. *A Review on Quantum Approximate Optimization Algorithm and its Variants*. Preprint at <https://arxiv.org/abs/2306.09198>, 2023. (Cited on page 23.)
- [Boost C++ Libraries 2022] Boost C++ Libraries. *Boost ODEINT: Ordinary Differential Equation Integration Library*. Boost, 2022. (Cited on page 149.)
- [Born & Fock 1928] M. Born and V. Fock. *Beweis des Adiabaticensatzes*. Zeitschrift für Physik, vol. 51, no. 3-4, pages 165–180, March 1928. (Cited on pages 4, 16 and 54.)
- [Brady *et al.* 2021] Lucas T. Brady, Christopher L. Baldwin, Aniruddha Bapat, Yaroslav Kharkov and Alexey V. Gorshkov. *Optimal Protocols in Quantum Annealing and Quantum Approximate Optimization Algorithm Problems*. Phys. Rev. Lett., vol. 126, page 070505, Feb 2021. (Cited on pages 24, 40, 131 and 169.)
- [Braida & Martiel 2021] Arthur Braida and Simon Martiel. *Anti-crossings and spectral gap during quantum adiabatic evolution*. Quantum Information Processing, vol. 20, no. 8, aug 2021. (Cited on pages 9 and 25.)
- [Braida *et al.* 2022] Arthur Braida, Simon Martiel and Ioan Todinca. *On constant-time quantum annealing and guaranteed approximations for graph optimization problems*. Quantum Science and Technology, vol. 7, no. 4, page 045030, 2022. (Cited on pages 9 and 25.)
- [Braida *et al.* 2024a] Arthur Braida, Simon Martiel and Ioan Todinca. *Avoided level crossings with exponentially closing gaps in quantum annealing*. Phys. Rev. A, vol. 109, page 022415, Feb 2024. (Cited on pages 9, 25 and 52.)
- [Braida *et al.* 2024b] Arthur Braida, Simon Martiel and Ioan Todinca. *Tight Lieb-Robinson Bound for approximation ratio in Quantum Annealing*. Accepted at npj Quantum Information, page “to appear”, 2024. (Cited on pages 9 and 25.)
- [Bravyi *et al.* 2006] S. Bravyi, M. B. Hastings and F. Verstraete. *Lieb-Robinson Bounds and the Generation of Correlations and Topological Quantum Order*. Phys. Rev. Lett., vol. 97, page 050401, Jul 2006. (Cited on page 140.)

- [Chen & Lucas 2021] Chi-Fang Chen and Andrew Lucas. *Operator Growth Bounds from Graph Theory*. Communications in Mathematical Physics, vol. 385, no. 3, page 1273–1323, July 2021. (Cited on page 140.)
- [Chen *et al.* 2023] Chi-Fang (Anthony) Chen, Andrew Lucas and Chao Yin. *Speed limits and locality in many-body quantum dynamics*. Reports on Progress in Physics, vol. 86, no. 11, page 116001, sep 2023. (Cited on pages 64, 130, 150, 151 and 152.)
- [Childs *et al.* 2002] A. Childs, E. Farhi, J. Goldstone and S. Gutmann. *Finding cliques by quantum adiabatic evolution*. Quantum Information and Computation, vol. 2, no. 3, Apr 2002. (Cited on pages 22, 44, 45, 173 and 174.)
- [Choi 2011] Vicky Choi. *Different adiabatic quantum optimization algorithms for the NP-complete exact cover and 3SAT problems*. Quantum Info. Comput., vol. 11, no. 7–8, page 638–648, jul 2011. (Cited on pages 23 and 40.)
- [Choi 2020] Vicky Choi. *The effects of the problem Hamiltonian parameters on the minimum spectral gap in adiabatic quantum optimization*. Quantum Information Processing, vol. 19, no. 3, page 90, 2020. (Cited on pages 9, 24, 25, 58, 67, 69, 74, 83 and 167.)
- [Cook 1971] Stephen A. Cook. *The complexity of theorem-proving procedures*. In Proceedings of the Third Annual ACM Symposium on Theory of Computing, STOC '71, page 151–158, New York, NY, USA, 1971. Association for Computing Machinery. (Cited on pages 6, 18, 30 and 31.)
- [Crosson & Lidar 2021] E. J. Crosson and D. A. Lidar. *Prospects for quantum enhancement with diabatic quantum annealing*. Nature Reviews Physics, vol. 3, no. 7, page 466–489, May 2021. (Cited on page 169.)
- [Cubitt & Montanaro 2016] Toby Cubitt and Ashley Montanaro. *Complexity Classification of Local Hamiltonian Problems*. SIAM Journal on Computing, vol. 45, no. 2, pages 268–316, 2016. (Cited on page 22.)
- [Cvetkovic *et al.* 1980] D.M. Cvetkovic, D.M. Cvetković, M. Doob and H. Sachs. *Spectra of graphs: Theory and application. Pure and applied mathematics : a series of monographs and textbooks*. Academic Press, 1980. (Cited on page 107.)
- [Dalzell *et al.* 2023] Alexander M. Dalzell, Nicola Pancotti, Earl T. Campbell and Fernando G.S.L. Brandão. *Mind the Gap: Achieving a Super-Grover Quantum Speedup by Jumping to the End*. In Proceedings of the 55th Annual ACM Symposium on Theory of Computing, STOC '23. ACM, June 2023. (Cited on pages 52 and 109.)

- [de Broglie 1924] Louis de Broglie. *Recherches sur la théorie des Quanta*. Theses, Migration - université en cours d'affectation, November 1924. (Cited on pages 3 and 15.)
- [de Falco *et al.* 1988] Diego de Falco, B. Apolloni and Nicolò Cesa-Bianchi. *A numerical implementation of quantum annealing*, 07 1988. (Cited on pages 8, 19, 36 and 52.)
- [Deutsch & Jozsa 1992] David Deutsch and Richard Jozsa. *Rapid solution of problems by quantum computation*. Proceedings of the Royal Society of London. Series A: Mathematical and Physical Sciences, vol. 439, no. 1907, pages 553–558, 1992. (Cited on pages 9 and 20.)
- [Deutsch & Penrose 1985] David Deutsch and Roger Penrose. *Quantum theory, the Church–Turing principle and the universal quantum computer*. Proceedings of the Royal Society of London. A. Mathematical and Physical Sciences, vol. 400, no. 1818, pages 97–117, 1985. (Cited on pages 8 and 19.)
- [Ebadi *et al.* 2022] S. Ebadi, A. Keesling, M. Cain, T. T. Wang, H. Levine, D. Bluvstein, G. Semeghini, A. Omran, J.-G. Liu, R. Samajdar, X.-Z. Luo, B. Nash, X. Gao, B. Barak, E. Farhi, S. Sachdev, N. Gemelke, L. Zhou, S. Choi, H. Pichler, S.-T. Wang, M. Greiner, V. Vuletić and M. D. Lukin. *Quantum optimization of maximum independent set using Rydberg atom arrays*. Science, vol. 376, no. 6598, pages 1209–1215, 2022. (Cited on pages 38 and 91.)
- [Efstathiou & Efstathiou 2018] Kyriakos Efstathiou and Marianna Efstathiou. *Celestial Gearbox*. Mechanical Engineering, vol. 140, no. 09, pages 31–35, 09 2018. (Cited on pages 1 and 13.)
- [Elgart & Hagedorn 2012] Alexander Elgart and George A. Hagedorn. *A note on the switching adiabatic theorem*. Journal of Mathematical Physics, vol. 53, no. 10, page 102202, 09 2012. (Cited on page 34.)
- [Farhi & Harrow 2019] Edward Farhi and Aram W Harrow. *Quantum Supremacy through the Quantum Approximate Optimization Algorithm*, 2019. (Cited on page 48.)
- [Farhi *et al.* 2000] Edward Farhi, Jeffrey Goldstone, Sam Gutmann and Michael Sipser. *Quantum computation by adiabatic evolution*. Preprint at <https://arxiv.org/abs/quant-ph/0001106>, 2000. (Cited on page 52.)
- [Farhi *et al.* 2001] E. Farhi, J. Goldstone, S. Gutmann, J. Lapan, A. Lundgren and D. Preda. *A Quantum Adiabatic Evolution Algorithm Applied to Random Instances of an NP-Complete Problem*. Science, vol. 292, no. 5516, page 472–475, Apr 2001. (Cited on pages 20 and 38.)
- [Farhi *et al.* 2014] Edward Farhi, Jeffrey Goldstone and Sam Gutmann. *A Quantum Approximate Optimization Algorithm*. Preprint at <https://arxiv.org/abs/>

- 1411.4028, 2014. (Cited on pages 10, 23, 46, 48, 51, 117, 119, 120, 123, 124, 130 and 137.)
- [Farhi *et al.* 2020] Edward Farhi, David Gamarnik and Sam Gutmann. *The Quantum Approximate Optimization Algorithm Needs to See the Whole Graph: Worst Case Examples*, 2020. (Cited on pages 51 and 64.)
- [Farhi *et al.* 2022] Edward Farhi, Jeffrey Goldstone, Sam Gutmann and Leo Zhou. *The Quantum Approximate Optimization Algorithm and the Sherrington-Kirkpatrick Model at Infinite Size*. Quantum, vol. 6, page 759, July 2022. (Cited on page 23.)
- [Feinstein *et al.* 2022] Natasha Feinstein, Louis Fry-Bouriaux, Sougato Bose and PA Warburton. *Effects of XX-catalysts on quantum annealing spectra with perturbative crossings*. arXiv preprint arXiv:2203.06779, 2022. (Cited on pages 24 and 82.)
- [Feynman 1982] Richard P. Feynman. *Simulating Physics with Computers*. International Journal of Theoretical Physics, vol. 21, pages 467–488, 1982. (Cited on pages 3, 8, 15 and 19.)
- [Goemans & Williamson 1995] Michel X. Goemans and David P. Williamson. *Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming*. J. ACM, vol. 42, no. 6, page 1115–1145, nov 1995. (Cited on pages 8, 19 and 50.)
- [Grover 1996] Lov K. Grover. *A Fast Quantum Mechanical Algorithm for Database Search*. In Proceedings of the Twenty-Eighth Annual ACM Symposium on Theory of Computing, STOC '96, page 212–219, New York, NY, USA, 1996. Association for Computing Machinery. (Cited on pages 9 and 20.)
- [Guéry-Odelin *et al.* 2019] D. Guéry-Odelin, A. Ruschhaupt, A. Kiely, E. Torrontegui, S. Martínez-Garaot and J. G. Muga. *Shortcuts to adiabaticity: Concepts, methods, and applications*. Rev. Mod. Phys., vol. 91, page 045001, Oct 2019. (Cited on pages 24 and 131.)
- [Haah *et al.* 2021] Jeongwan Haah, Matthew B. Hastings, Robin Kothari and Guang Hao Low. *Quantum Algorithm for Simulating Real Time Evolution of Lattice Hamiltonians*. SIAM Journal on Computing, Jan 2021. Published online. Special issue FOCS 2018. (Cited on pages 24, 64, 122, 152, 153 and 169.)
- [Hagberg *et al.* 2008] Aric Hagberg, Pieter Swart and Daniel S Chult. *Exploring network structure, dynamics, and function using NetworkX*. Technical report, Los Alamos National Lab.(LANL), Los Alamos, NM (United States), 2008. (Cited on page 148.)

- [Halperin *et al.* 2002] Eran Halperin, Dror Livnat and Uri Zwick. *MAX CUT in cubic graphs*. J. Algorithms, vol. 53, pages 169–185, 2002. (Cited on pages 8, 19 and 50.)
- [Hammer & Rudeanu 1968] Peter L Hammer and Sergiu Rudeanu. Boolean methods in operations research and related areas. Springer Berlin, Heidelberg, 1968. (Cited on page 22.)
- [Hastings 2004] M. B. Hastings. *Lieb-Schultz-Mattis in higher dimensions*. Phys. Rev. B, vol. 69, page 104431, Mar 2004. (Cited on pages 60 and 64.)
- [Hastings 2010] M. B. Hastings. *Locality in Quantum Systems*. In J. Frohlich, M. Salmhofer, V. Mastropietro, W. De Roeck and L. Cugliandolo, editors, Quantum Theory from Small to Large Scales: Lecture Notes of the Les Houches Summer School, chapter 3. Oxford University Press, Oxford, 2010. (Cited on page 62.)
- [Hastings 2019] Matthew B. Hastings. *Classical and quantum bounded depth approximation algorithms*. Quantum Inf. Comput., vol. 19, no. 13&14, pages 1116–1140, 2019. (Cited on pages 49, 51 and 130.)
- [Hauke *et al.* 2020] Philipp Hauke, Helmut G Katzgraber, Wolfgang Lechner, Hidetoshi Nishimori and William D Oliver. *Perspectives of quantum annealing: Methods and implementations*. Reports on Progress in Physics, vol. 83, no. 5, page 054401, 2020. (Cited on page 39.)
- [Henriet *et al.* 2020] Loïc Henriet, Lucas Beguin, Adrien Signoles, Thierry Lahaye, Antoine Browaeys, Georges-Olivier Reymond and Christophe Jurczak. *Quantum computing with neutral atoms*. Quantum, vol. 4, page 327, September 2020. (Cited on page 38.)
- [Herrman *et al.* 2021] Rebekah Herrman, Phillip C. Lotshaw, James Ostrowski, T. Humble and George Siopsis. *Multi-angle quantum approximate optimization algorithm*. Scientific Reports, vol. 12, 2021. (Cited on page 23.)
- [Hirvonen *et al.* 2017] Juho Hirvonen, Joel Rybicki, Stefan Schmid and Jukka Suomela. *Large Cuts with Local Algorithms on Triangle-Free Graphs*. The Electronic Journal of Combinatorics, pages P4–21, 2017. (Cited on page 49.)
- [Håstad 2001] Johan Håstad. *Some optimal inapproximability results*. J. ACM, vol. 48, no. 4, page 798–859, jul 2001. (Cited on page 50.)
- [James *et al.* 2017] Conrad D. James, James B. Aimone, Nadine E. Miner, Craig M. Vineyard, Fredrick H. Rothganger, Kristofor D. Carlson, Samuel A. Mulder, Timothy J. Draelos, Aleksandra Faust, Matthew J. Marinella, John H. Naegele and Steven J. Plimpton. *A historical survey of algorithms and hardware architectures for neural-inspired and neuromorphic computing applications*.

- Biologically Inspired Cognitive Architectures, vol. 19, pages 49–64, 2017. (Cited on pages 3 and 15.)
- [Jansen *et al.* 2007] Sabine Jansen, Mary-Beth Ruskai and Ruedi Seiler. *Bounds for the adiabatic approximation with applications to quantum computation*. Journal of Mathematical Physics, vol. 48, no. 10, page 102111, 10 2007. (Cited on page 34.)
- [Jones *et al.* 01] Eric Jones, Travis Oliphant, Pearu Peterson *et al.* *SciPy: Open source scientific tools for Python*, 2001–. (Cited on page 103.)
- [Kadian *et al.* 2021] Karuna Kadian, Sunita Garhwal and Ajay Kumar. *Quantum walk and its application domains: A systematic review*. Computer Science Review, vol. 41, page 100419, 2021. (Cited on page 23.)
- [Kadowaki & Nishimori 1998] Tadashi Kadowaki and Hidetoshi Nishimori. *Quantum annealing in the transverse Ising model*. Phys. Rev. E, vol. 58, pages 5355–5363, Nov 1998. (Cited on pages 8, 20 and 36.)
- [Karp 1972] Richard M. Karp. Reducibility among combinatorial problems, pages 85–103. Springer US, Boston, MA, 1972. (Cited on pages 6, 18 and 31.)
- [Kato 1950] Tosio Kato. *On the Adiabatic Theorem of Quantum Mechanics*. Journal of the Physical Society of Japan, vol. 5, no. 6, pages 435–439, 1950. (Cited on pages 5 and 17.)
- [Khot *et al.* 2007] Subhash Khot, Guy Kindler, Elchanan Mossel and Ryan O’Donnell. *Optimal Inapproximability Results for MAX-CUT and Other 2-Variable CSPs?* SIAM Journal on Computing, vol. 37, no. 1, pages 319–357, 2007. (Cited on page 50.)
- [Knysh & Smelyanskiy 2006] S. Knysh and V. N. Smelyanskiy. *Quantum Adiabatic Evolution Algorithm and Quantum Phase Transition in 3-Satisfiability Problem*, 2006. (Cited on page 22.)
- [Landau 1932] Lev Landau. *Zur theorie der energieubertragung. II*. Physikalische Zeitschrift der Sowjetunion, vol. 2, page 46, 1932. (Cited on pages 5 and 17.)
- [Larousse 2023] Larousse. *calculus*, 2023. (Cited on pages 1 and 13.)
- [Laumann *et al.* 2015] C R Laumann, R Moessner, A Scardicchio and S L Sondhi. *Quantum annealing: The fastest route to quantum computation?* The European Physical Journal Special Topics, vol. 224, no. 1, pages 75–88, 2015. (Cited on pages 23 and 58.)
- [Levin 1973] Leonid Anatolevich Levin. *Universal sequential search problems*. Problemy peredachi informatsii, vol. 9, no. 3, pages 115–116, 1973. (Cited on pages 6, 18, 30 and 31.)

- [Lieb & Robinson 1972] Elliott H. Lieb and Derek W. Robinson. *The finite group velocity of quantum spin systems*. Communications in Mathematical Physics, vol. 28, no. 3, pages 251 – 257, 1972. (Cited on pages 5, 18 and 60.)
- [Lieb et al. 1961] Elliott Lieb, Theodore Schultz and Daniel Mattis. *Two soluble models of an antiferromagnetic chain*. Annals of Physics, vol. 16, no. 3, pages 407–466, 1961. (Cited on page 64.)
- [Low & Chuang 2017] Guang Hao Low and Isaac L. Chuang. *Optimal Hamiltonian Simulation by Quantum Signal Processing*. Phys. Rev. Lett., vol. 118, page 010501, Jan 2017. (Cited on page 64.)
- [Lucas 2014] Andrew Lucas. *Ising formulations of many NP problems*. Frontiers in Physics, vol. 2, page 5, 2014. (Cited on pages 33 and 38.)
- [Lykov et al. 2023] Danylo Lykov, Jonathan Wurtz, Cody Poole, Mark Saffman, Tom Noel and Yuri Alexeev. *Sampling frequency thresholds for the quantum advantage of the quantum approximate optimization algorithm*. npj Quantum Information, vol. 9, no. 1, page 73, 2023. (Cited on page 23.)
- [Manin 1980] Yuri Manin. *Computable and Uncomputable*. Sovetskoye Radio, vol. 128, page 15, 1980. (Cited on pages 3, 15 and 19.)
- [Marwaha 2021] Kunal Marwaha. *Local classical MAX-CUT algorithm outperforms $p = 2$ QAOA on high-girth regular graphs*. Quantum, vol. 5, page 437, April 2021. (Cited on page 51.)
- [Moosavian et al. 2022] Ali Hamed Moosavian, Seyed Sajad Kahani and Salman Beigi. *Limits of Short-Time Evolution of Local Hamiltonians*. Quantum, vol. 6, page 744, June 2022. (Cited on pages 25, 51 and 64.)
- [Morita & Nishimori 2008] Satoshi Morita and Hidetoshi Nishimori. *Mathematical foundation of quantum annealing*. Journal of Mathematical Physics, vol. 49, no. 12, page 125210, 12 2008. (Cited on page 34.)
- [Nielsen & Chuang 2002] Michael A Nielsen and Isaac Chuang. *Quantum computation and quantum information*, 2002. (Cited on pages 17 and 32.)
- [Pelofske et al. 2024] Elijah Pelofske, Andreas Bärtzchi and Stephan Eidenbenz. *Short-depth QAOA circuits and quantum annealing on higher-order ising models*. npj Quantum Information, vol. 10, no. 1, page 30, 2024. (Cited on page 23.)
- [Planck 1903] Planck. *Treatise on thermodynamics*. Longmans, Green, and Co., 1903. (Cited on pages 3 and 15.)
- [Preskill 2018] John Preskill. *Quantum Computing in the NISQ era and beyond*. Quantum, vol. 2, page 79, August 2018. (Cited on page 23.)

- [Roland & Cerf 2002] Jérémie Roland and Nicolas J. Cerf. *Quantum search by local adiabatic evolution*. Phys. Rev. A, vol. 65, page 042308, Mar 2002. (Cited on pages 22, 25, 40, 52 and 131.)
- [Sachdev 1999] Subir Sachdev. *Quantum phase transitions*. Physics world, vol. 12, no. 4, page 33, 1999. (Cited on pages 5, 17, 36 and 51.)
- [Saffman *et al.* 2010] M. Saffman, T. G. Walker and K. Mølmer. *Quantum information with Rydberg atoms*. Rev. Mod. Phys., vol. 82, pages 2313–2363, Aug 2010. (Cited on page 38.)
- [Schrödinger 1926] E. Schrödinger. *An Undulatory Theory of the Mechanics of Atoms and Molecules*. Phys. Rev., vol. 28, pages 1049–1070, Dec 1926. (Cited on pages 3 and 15.)
- [Schützhold & Schaller 2006] Ralf Schützhold and Gernot Schaller. *Adiabatic quantum algorithms as quantum phase transitions: First versus second order*. Phys. Rev. A, vol. 74, page 060304, Dec 2006. (Cited on page 22.)
- [Seki & Nishimori 2012] Yuya Seki and Hidetoshi Nishimori. *Quantum annealing with antiferromagnetic fluctuations*. Phys. Rev. E, vol. 85, page 051112, May 2012. (Cited on page 24.)
- [Serret *et al.* 2020] Michel Fabrice Serret, Bertrand Marchand and Thomas Ayrat. *Solving optimization problems with Rydberg analog quantum computers: Realistic requirements for quantum advantage using noisy simulation and classical benchmarks*. Phys. Rev. A, vol. 102, page 052617, Nov 2020. (Cited on page 38.)
- [Shaydulin & Alexeev 2019] R. Shaydulin and Y. Alexeev. *Evaluating Quantum Approximate Optimization Algorithm: A Case Study*. In 2019 Tenth International Green and Sustainable Computing Conference (IGSC), pages 1–6, Los Alamitos, CA, USA, oct 2019. IEEE Computer Society. (Cited on page 23.)
- [Shor 1994] P.W. Shor. *Algorithms for quantum computation: discrete logarithms and factoring*. In Proceedings 35th Annual Symposium on Foundations of Computer Science, pages 124–134, 1994. (Cited on pages 9 and 20.)
- [Somma *et al.* 2012] Rolando D. Somma, Daniel Nagaj and Mária Kieferová. *Quantum Speedup by Quantum Annealing*. Phys. Rev. Lett., vol. 109, page 050501, Jul 2012. (Cited on page 82.)
- [Suomela 2013] Jukka Suomela. *Survey of local algorithms*. ACM Comput. Surv., vol. 45, no. 2, pages 24:1–24:40, 2013. (Cited on pages 49 and 119.)
- [Suzuki 1976] Masuo Suzuki. *Generalized Trotter’s formula and systematic approximants of exponential operators and inner derivations with applications to many-body problems*. Communications in Mathematical Physics, vol. 51, no. 2, pages 183–190, 1976. (Cited on page 46.)

- [Tran *et al.* 2019] Minh C. Tran, Andrew Y. Guo, Yuan Su, James R. Garrison, Zachary Eldredge, Michael Foss-Feig, Andrew M. Childs and Alexey V. Gorshkov. *Locality and Digital Quantum Simulation of Power-Law Interactions*. Phys. Rev. X, vol. 9, page 031006, Jul 2019. (Cited on pages 25, 64 and 122.)
- [Trotter 1959] Hale F Trotter. *On the product of semi-groups of operators*. Proceedings of the American Mathematical Society, vol. 10, no. 4, pages 545–551, 1959. (Cited on page 46.)
- [Tsuda *et al.* 2013] Junichi Tsuda, Yuuki Yamanaka and Hidetoshi Nishimori. *Energy Gap at First-Order Quantum Phase Transitions: An Anomalous Case*. Journal of the Physical Society of Japan, vol. 82, no. 11, page 114004, 2013. (Cited on page 23.)
- [Ulmann 2008] Bernd Ulmann. *Simulating a car suspension system*. Website at https://www.analogmuseum.org/english/examples/vehicle_simulation/, 2008. (Cited on pages 2 and 14.)
- [von Neumann & Wigner 1929] J. von Neumann and E. P. Wigner. *Über das Verhalten von Eigenwerten bei adiabatischen Prozessen*. Physikalische Zeitschrift, vol. 30, pages 467–470, January 1929. (Cited on pages 52, 54 and 75.)
- [Wang & Hazzard 2020] Zhiyuan Wang and Kaden R.A. Hazzard. *Tightening the Lieb-Robinson Bound in Locally Interacting Systems*. PRX Quantum, vol. 1, page 010303, Sep 2020. (Cited on pages 25, 139, 140, 141 and 184.)
- [Werner *et al.* 2023] Matthias Werner, Artur García-Sáez and Marta P. Estarellas. *Bounding first-order quantum phase transitions in adiabatic quantum computing*. Phys. Rev. Res., vol. 5, page 043236, Dec 2023. (Cited on page 86.)
- [Wikipedia 2005] Wikipedia. *Antikythera Mechanism*, 2005. (Cited on pages 2 and 14.)
- [Wikipedia 2024] Wikipedia. *Analog Computer*, 2024. (Cited on pages 2 and 14.)
- [Wilkinson 1987] M Wilkinson. *Narrowly avoided crossings*. Journal of Physics A: Mathematical and General, vol. 20, no. 3, page 635, feb 1987. (Cited on pages 5, 17 and 52.)
- [Wilkinson 1989] Michael Wilkinson. *Statistics of multiple avoided crossings*. Journal of Physics A: Mathematical and General, vol. 22, no. 14, page 2795, 1989. (Cited on pages 5, 17, 52, 54 and 75.)
- [Wurtz & Love 2021] Jonathan Wurtz and Peter Love. *MaxCut quantum approximate optimization algorithm performance guarantees for p greater than 1*. Phys. Rev. A, vol. 103, page 042612, Apr 2021. (Cited on pages 137 and 148.)

- [Yang *et al.* 2017] Zhi-Cheng Yang, Armin Rahmani, Alireza Shabani, Hartmut Neven and Claudio Chamon. *Optimizing Variational Quantum Algorithms Using Pontryagin’s Minimum Principle*. Phys. Rev. X, vol. 7, page 021027, May 2017. (Cited on page 24.)
- [Yarkoni *et al.* 2022] Sheir Yarkoni, Elena Raponi, Thomas Bäck and Sebastian Schmitt. *Quantum annealing for industry applications: introduction and review*. Reports on Progress in Physics, vol. 85, no. 10, page 104001, sep 2022. (Cited on page 23.)
- [Zener 1932] Clarence Zener. *Non-Adiabatic Crossing of Energy Levels*. Proceedings of the Royal Society of London. Series A, Containing Papers of a Mathematical and Physical Character, vol. 137, no. 833, pages 696–702, 1932. (Cited on pages 5 and 17.)
- [Zhang 2021] Yueheng Zhang. *On the principal eigenvector of a graph*. arXiv preprint arXiv:2107.14421, 2021. (Cited on pages 88, 91 and 175.)
- [Zwiebach 2018] Barton Zwiebach. *Chapter 1: Non-degenerate and degenerate Perturbation Theory*. In Quantum Physics III —MIT Course. MIT OpenCourseWare, 2018. (Cited on page 53.)

Arthur BRAIDA

Calcul Analogique Quantique pour des Problèmes Combinatoires sur Graphes NP-difficile

Résumé :

L'objectif principal de cette thèse est de fournir un éclairage théorique de la complexité du calcul quantique en temps continu (QA et AQC), de la compréhension du phénomène physique (AC) qui conduit à l'échec de l'AQC jusqu'à des preuves de performance de QA en temps court et constant. Pour atteindre cet objectif, nous utilisons différents outils analytiques empruntés à la physique théorique, comme l'analyse perturbative des systèmes quantiques et la borne de Lieb-Robinson sur la vitesse de corrélation dans les systèmes quantiques. La manipulation des graphes et la théorie spectrale des graphes sont nécessaires pour obtenir des résultats sur des classes spécifiques de graphes. Nous avons également introduit une nouvelle version paramétrée du QA standard afin d'affiner l'analyse.

Tout d'abord, nous souhaitons obtenir une définition mathématique d'un AC afin d'en faciliter la compréhension lors de l'étude d'une classe spécifique de graphes sur lesquels nous souhaitons résoudre le problème de Maximum Cut. En plus de l'appui analytique que nous développons, nous apportons une étude numérique pour justifier la nature plus générale de notre définition par rapport à la précédente. Grâce à une analyse perturbative, nous parvenons à montrer que sur les graphes bipartis, un gap de fermeture exponentielle peut apparaître si le graphe est suffisamment irrégulier. Notre nouvelle définition de l'AC nous permet de remettre en question l'efficacité de l'AQC pour le résoudre malgré le temps d'exécution exponentiellement long que le théorème adiabatique impose pour garantir la solution optimale.

Le deuxième axe est consacré à la performance du QA en temps constant court. Bien que le QA soit intrinsèquement non-locale, la borne de LR nous permet de l'approximer avec une évolution locale. Une première approche est utilisée pour développer la méthode et montrer la non trivialité du résultat, c'est-à-dire au-dessus du choix aléatoire. Ensuite, nous définissons une notion d'analyse locale en exprimant le ratio d'approximation avec la seule connaissance de la structure locale. Une borne LR fine et adaptative est développée, nous permettant de trouver une valeur numérique du ratio d'approximation surpassant les algorithmes quantiques et classiques (strictement) locaux.

Tous ces travaux de recherche ont été poursuivis entre l'équipe QuantumLab d'Eviden et l'équipe Graphe, Algorithme et Modèle de Calcul (GAMOC) du Laboratoire d'Informatique Fondamentale d'Orléans (LIFO). Le travail numérique a été implémenté en utilisant le langage de programmation Julia ainsi que Python avec le logiciel QAPTIVA d'Eviden pour simuler efficacement l'équation de Schrödinger.

Mots clés : Recuit quantique, Calcul quantique adiabatique, ratio d'approximation, MaxCut, problèmes combinatoires, borne de Lieb-Robinson, théorie perturbative, théorie spectrale des graphes

Arthur BRAIDA

Analog Quantum Computing for NP-Hard Combinatorial Graph Problems

Abstract :

The main objective of this thesis is to provide theoretical insight into the computational complexity of continuous-time quantum computing (QA and AQC), from understanding the physical phenomenon (AC) that leads to AQC failure to proving short constant-time QA efficiency.

To achieve this goal, we use different analytical tools borrowed from theoretical physics like perturbative analysis of quantum systems and the Lieb-Robinson bound on the velocity of correlation in quantum systems. Graph manipulation and spectral graph theory are necessary to derive results on a specific class of graph. We also introduced a new parametrized version of the standard QA to tighten the analysis

First, we want to obtain a mathematical definition of an AC to be easier to grasp when studying a specific class of graph on which we want to solve the Maximum Cut problem. We support our new definition with a proven theorem that links it to exponentially small minimum gap and numerical evidence is brought to justify its more general nature compared to the previous one. With a perturbative analysis, we manage to show that on bipartite graphs, exponentially closing gap can arise if the graph is irregular enough. Our new definition of AC allows us to question the efficiency of AQC to solve it despite the exponentially long runtime the adiabatic theorem imposes to guarantee the optimal solution.

The second axis is dedicated to the performance of QA at short constant times. Even though QA is inherently non-local, the LR bound allows us to approximate it with a local evolution. A first approach is used to develop the method and to show the non-triviality of the result, i.e. above random guess. Then we define a notion of local analysis by expressing the approximation ratio with only knowledge of the local structure. A tight and adaptive LR bound is developed allowing us to find a numerical value outperforming quantum and classical (strictly) local algorithms.

All this research work has been pursue between Eviden QuantumLab team and the Graphe, Algorithme et Modèle de Calcul (GAMOC) team at the Laboratoire d'Informatique Fondamentale d'Orléans (LIFO). The numerical work has been implemented using Julia programming Language as well as Python with the QAPTIVA software of Eviden to efficiently simulate the Schrödinger equation.

Keywords : Quantum Annealing, Adiabatic Quantum Computing, approximation ratio, MaxCut, combinatorial problems, Lieb-Robinson bound, perturbative theory, spectral graph theory