

Imperial College London
Department of Physics

Continuous-time Quantum Computing

Adam Callison

**Imperial College
London**

Submitted in part fulfilment of the requirements for
the degree of Doctor of Philosophy in Physics of
Imperial College London, March 2021

Copyright Declaration

The copyright of this thesis rests with the author. Unless otherwise indicated, its contents are licensed under a Creative Commons Attribution-Non Commercial 4.0 International Licence (CC BY-NC). Under this licence, you may copy and redistribute the material in any medium or format. You may also create and distribute modified versions of the work. This is on the condition that: you credit the author and do not use it, or any derivative works, for a commercial purpose. When reusing or sharing this work, ensure you make the licence terms clear to others by naming the licence and linking to the licence text. Where a work has been adapted, you should indicate that the work has been changed and describe those changes. Please seek permission from the copyright holder for uses of this work that are not included in this licence or permitted under UK Copyright Law.

Abstract

Quantum computation using continuous-time evolution under a natural hardware Hamiltonian is a promising near- and mid-term direction toward powerful quantum computing hardware. Continuous-time quantum computing (CTQC) encompasses continuous-time quantum walk computing (QW), adiabatic quantum computing (AQC), and quantum annealing (QA), as well as other strategies which contain elements of these three. While much of current quantum computing research focuses on the discrete-time gate model, which has an appealing similarity to the discrete logic of classical computation, the continuous nature of quantum information suggests that continuous-time quantum information processing is worth exploring. A versatile context for CTQC is the transverse Ising model, and this thesis will explore the application of Ising model CTQC to classical optimization problems.

Classical optimization problems have industrial and scientific significance, including in logistics, scheduling, medicine, cryptography, hydrology and many other areas. Along with the fact that such problems often have straightforward, natural mappings onto the interactions of readily-available Ising model hardware makes classical optimization a fruitful target for CTQC algorithms. After introducing and explaining the CTQC framework in detail, in this thesis I will, through a combination of numerical, analytical, and experimental work, examine the performance of various forms of CTQC on a number of different optimization problems, and investigate the underlying physical mechanisms by which they operate.

Acknowledgements

Personal

Along with having its share of enjoyable and fulfilling times, this PhD has been a uniquely challenging experience. I have faced many obstacles, a non-trivial fraction of which may have been insurmountable without the support of my external supervisor. Viv has undeniably been an ideal PhD supervisor, having always been available to advise on research, writing, and any other aspect of academic life. I would similarly like to thank Nick, who has been a close collaborator throughout my PhD but who has effectively acted as an additional supervisor whom I could also turn to at any time for advice. I would also like to thank David, my collaborator on the Shortest Vector Problem work, who has been a pleasure to work and chat with.

I would have had a much more difficult time completing this PhD without the emotional support from my family, including my grandmother Terry, my brother Jamie and sister-in-law Rashi, my sister Laura, and in particular my parents Jacqui and John, and also without the joy that came with the arrival of my nephews Jawahar, Eddie and Max. Finally, I am effusively grateful for the day-to-day emotional and practical support from my partner Jamila, with whom I am fortunate enough to have been able move in with during this PhD, and without whom I would not have been able to complete it.

Funding

I am grateful for my PhD funding provided by UKRI EPSRC Grant No. EP/L016524/1 via the Imperial College London Centre for Doctoral Training in Controlled Quantum Dynamics.

Publications and Statement of Originality

I certify that, except where appropriately referenced, the work presented in this thesis is my own and has not been previously submitted to qualify for any other academic award. Chapter [3](#) (along with some relevant parts of Chapter [2](#)) is based on a paper published under the Creative Commons Attribution 3.0 licence as ([Callison et al., 2019](#)) and reuses some assets and text from that paper. Chapter [5](#) (along with some relevant parts of Chapter [2](#)) is based on a paper published under the Creative Commons Attribution 4.0 International licence as ([Callison et al., 2021](#)) and reuses some assets and text from that paper. Chapter [6](#) (along with some relevant parts of Chapter [2](#)) is based on a paper published in Physical Review A as ([Joseph et al., 2021b](#)) (with a similar version available under the arXiv.org perpetual, non-exclusive license at ([Joseph et al., 2021a](#))), and reuses some assets and text from that paper.

Contents

Copyright Declaration	3
Abstract	4
Acknowledgements	5
Publications and Statement of Originality	6
Contents	7
List of Figures	11
1 Introduction	14
2 Background Theory	19
2.1 Optimization Problems	19
2.1.1 Classical algorithms: the branch-and-bound framework	20
2.1.2 Classical algorithms: Simulated Annealing	21
2.1.3 Sherrington-Kirkpatrick spin glass	23
2.1.4 MAX2SAT	24
2.1.5 Shortest vector problem	25
2.2 CTQC: Ising model and problem mapping	26
2.2.1 Ising mapping of the Sherrington-Kirkpatrick ground-state problem	28

2.2.2	Ising mapping of the MAX2SAT problem	29
2.3	CTQC: Unstructured search problem	30
2.4	CTQC: Quantum Walks	30
2.4.1	Continuous-time quantum walks	31
2.4.2	Comparison to classical random walk	31
2.4.3	Computing using a quantum walk	33
2.4.4	Graph choice for quantum walk computing	35
2.4.5	Solving the search problem using quantum walks	37
2.5	CTQC: Adiabatic Quantum Computing	39
2.5.1	Proof of the adiabatic theorem	40
2.5.2	Solving the search problem using AQC	45
2.6	CTQC: Quantum Annealing	46
2.7	Numerical Methods	50
2.7.1	Simulation of quantum dynamics	51
2.7.2	Problem instance generation	51
2.7.3	Fitting and optimization	52
3	Using quantum walks to solve optimization problems	53
3.1	Introduction	53
3.2	Quantum walks with spin glasses	55
3.2.1	Defining success probability	55
3.2.2	Random energy model	59
3.2.3	Setting the hopping rate	59
3.2.4	Success probability	64
3.2.5	Mixing times	66
3.3	Computational mechanisms	69

3.3.1	Role of correlations in SK	69
3.3.2	Energy conservation dynamics	72
3.4	Chapter summary	75
4	MAX2SAT	78
4.1	Quantum walks on MAX2SAT and similarities with SK	79
4.2	Adiabatic quantum computing on MAX2SAT	85
4.3	Comparing QW and AQC	89
4.4	Classical algorithm for MAX2SAT	92
4.5	Chapter summary	96
5	Rapid quenches in quantum annealing	101
5.1	Introduction	101
5.2	Rapid quench examples	105
5.2.1	Two-stage quantum walk	105
5.2.2	Biased two-stage quantum walk	108
5.2.3	Pre-annealed quantum walk	110
5.3	Energy redistribution mechanism	112
5.4	Ensuring significant dynamics	119
5.4.1	Quantifying the strength of short-time dynamics	119
5.4.2	Analytical bounds on $\overline{\text{Dyn}}$	122
5.4.3	Example: two qubit system	127
5.4.4	Example: Sherrington-Kirkpatrick spin-glass	128
5.4.5	Example: unstructured search	129
5.5	Using dynamics to find heuristic quench parameters	131
5.5.1	Heuristic hopping rate for a quantum walk	131
5.5.2	Heuristic schedule for quantum annealing	135

5.6	Chapter summary	136
6	Applying quantum annealing to the shortest vector problem	138
6.1	Introduction	138
6.2	Ising model mappings for SVP	140
6.2.1	Size of qudit operators	141
6.2.2	Hamming-weight-encoded Qudits	144
6.2.3	Binary-encoded qudits	145
6.3	Results	146
6.3.1	Numerical Simulation	147
6.3.2	D-Wave Quantum Annealer	149
6.4	Chapter summary	155
7	Conclusion	157
7.1	Summary of Thesis Achievements	157
7.2	Future Work	159
	Bibliography	162

List of Figures

2.1	Example of an integer lattice	26
2.2	A graph with line structure	32
2.3	Comparison of classical random and quantum walk	34
2.4	A hypercube graph	36
2.5	Unstructured search by quantum walk	38
2.6	Schedules for unstructured search by adiabatic quantum computing	47
2.7	Success probability for unstructured search by adiabatic quantum computing . .	48
2.8	Quantum versus simulated annealing	49
3.1	Infinite time average success probability for unstructured search by quantum walk	57
3.2	Example of success probability versus hopping rate for quantum walks on spin- glasses	60
3.3	Optimal hopping rate distribution for quantum walks on spin-glasses	61
3.4	Hopping rate width varying with problem size	63
3.5	Examples of quantum walks on spin-glasses	65
3.6	Infinite time success-probability scaling for quantum walks on spin-glasses	65
3.7	Mixing time scaling for quantum walks on spin-glasses	67

3.8	Short time success probability scaling for quantum walks on spin-glasses	68
3.9	Importance of problem structure for success probability of quantum walks	71
3.10	Importance of problem structure for hopping rate width of quantum walks	71
3.11	Importance of low energy initial conditions in quantum walks	74
3.12	Energy conservation mechanism for quantum walks	75
4.1	Success probability scaling for quantum walks on MAX2SAT	80
4.2	Example of success probability versus hopping rate for quantum walks on MAX2SAT	81
4.3	Optimal hopping rate distribution for quantum walks on MAX2SAT	82
4.4	Hopping rate width varying with problem size for MAX2SAT	83
4.5	Scaling of hopping rate width varying with problem size for MAX2SAT	84
4.6	Importance of problem structure for success probability of quantum walks on MAX2SAT	85
4.7	Importance of low energy initial conditions in quantum walks on MAX2SAT	86
4.8	Scaling of mean adiabatic quantum computing on MAX2SAT	87
4.9	Scaling of median adiabatic quantum computing on MAX2SAT	88
4.10	Hardness comparisons between quantum walks and adiabatic quantum computing on MAX2SAT	91
4.11	Heatmap comparison of hardness between quantum walks and adiabatic quantum computing on MAX2SAT	92
4.12	Classical algorithm hardness scaling for MAX2SAT	94
4.13	Histograms of classical algorithm hardness for MAX2SAT	95
4.14	Heatmap comparisons of classical hardness with quantum walks and with adiabatic quantum computing for MAX2SAT	97

4.15	State of the art classical algorithm hardness at large sizes for MAX2SAT	98
5.1	Two stage quantum walk schedules	106
5.2	Two-stage quantum walk on simple Hamiltonian	107
5.3	Two-stage quantum walk on spin-glass Hamiltonian	108
5.4	Biased two-stage quantum walk on simple Hamiltonian	109
5.5	Pre-annealed quantum walk schedules	111
5.6	Pre-annealed quantum walks on spin-glasses	112
5.7	Success probability scalings for pre-annealed quantum walks on spin-glasses . . .	113
5.8	Comparison of average dynamic coefficient with spectral gap for spin-glass Hamil- tonians	123
5.9	Semi-analytical lower bound on average dynamic coefficient	127
5.10	Comparison of quantum walk success probability with average dynamic coeffi- cient for simple Hamiltonian	132
5.11	Success probability scaling of quantum walks on spin-glasses with new heuristic hopping rate	133
5.12	Heuristic quench schedules	135
6.1	Numerical results for quantum annealing on the shortest vector problem	147
6.2	Comparison of minimum spectral gaps between the two encodings	149
6.3	Chimera embedding diagram	150
6.4	Histograms of experimental results for quantum annealing on the shortest vector problem	151
6.5	Experimental results at different sizes for quantum annealing on the shortest vector problem	153

Chapter 1

Introduction

The concept of quantum computing (QC) was introduced in the 1980s ([Feynman, 1982](#); [Benioff, 1980](#); [Deutsch and Penrose, 1985](#); [Manin, 1980](#)), alongside other advances such as quantum key distribution (QKD) ([Bennett et al., 1983](#); [Bennett and Brassard, 1984](#)) and later quantum state teleportation ([Bennett et al., 1993](#)), as one of the first proposals in quantum information science for putting quantum systems to practical use. While teleportation and QKD were quickly understood to be experimentally feasible (with the first experimental implementations arriving in 1997-1998 ([Bouwmeester et al., 1997](#); [Boschi et al., 1998](#)) and 2008 ([Dixon et al., 2008](#)) respectively), it remained unclear for some time whether QC could ever be achieved experimentally, due to the scale on which the inherently fragile quantum coherence must be maintained.

In the mid 1990s, a major breakthrough occurred in the development by [Shor \(1994\)](#) of an efficient quantum algorithm for integer factorization, a problem for which efficient classical algorithms are impossible and which is also a central problem in computer science, being the basis for much of modern computer security. Shor’s work was the first “useful” quantum algorithm and, if experimentally feasible, would threaten the security of secure classical communication and information storage. Around the same time, it was shown ([Shor, 1995](#); [Calderbank and Shor, 1996](#); [Steane, 1996](#); [Knill et al., 1996](#); [Aharonov and Ben-Or, 1996](#)) that quantum error correction (QEC) could be used to protect quantum computations from local decoherence by spreading quantum information out non-locally, suddenly making experimental implementations of QC much more plausible. These two discoveries together kickstarted the race to build

a real quantum computer.

Up to the late 1990s, work in QC tended to focus entirely on the gate model, based on the prescription introduced by [Deutsch and Penrose \(1985, 1989\)](#), which closely resembles classical digital computation. In a gate model algorithm, a set of qubits (the quantum analogs of classical bits that can be in the states $|0\rangle$, $|1\rangle$, or any arbitrary superposition $\alpha|0\rangle + \beta|1\rangle$ thereof) are each prepared in the state $|0\rangle$. Then, this qubit register is acted upon by a sequence of unitary operations, or quantum gates, of arbitrary fixed size, after which a measurement in the $\{|0\rangle, |1\rangle\}$ basis is performed to read out the algorithm output. The field then diversified, with the introduction of computation by discrete-time quantum walk ([Aharonov et al., 2001](#); [Shenvi et al., 2003](#)) (although at first these were for the most part intended to be run in the standard gate model setting), measurement-based quantum computing (MBQC) ([Raussendorf et al., 2003](#)), adiabatic quantum computing (AQC) ([Farhi et al., 2000](#)), computation by continuous-time quantum-walk (QW) ([Farhi and Gutmann, 1998](#)), and quantum annealing (QA) ([Kadowaki and Nishimori, 1998](#); [Finnila et al., 1994](#)). The latter three forms of QC are known collectively as continuous-time quantum computing (CTQC), and will be the focus of this thesis.

More broadly, QC is in principle universal, as was shown for the gate model by [Deutsch and Penrose \(1985\)](#) via the quantum Turing machine construction (a quantum generalization of the classical Turing machine model). Classical computing is also universal; that is, there is nothing that can be computed in QC that cannot also be computed in classical computing. Thus, the ways in which QC is more powerful than classical computing lie not in computability but in complexity: there exist computational problems that can be solved much more quickly (or with much less of some other resource) in QC than in classical computing. Most importantly, there exist some problems which can be solved efficiently in QC but not in classical computing, such as integer factorization. Other QC models have been shown to be polynomially equivalent to the gate model, including discrete-time quantum walks ([Lovett et al., 2010](#)), MBQC ([Raussendorf et al., 2003](#)), and both AQC ([Aharonov et al., 2004](#); [Kempe et al., 2006](#); [Siu, 2005](#); [Oliveira and Terhal, 2008](#); [Kitaev et al., 2002](#); [Deift et al., 2007](#)) and QW ([Childs, 2009](#); [Childs et al., 2013](#)) in CTQC.

While the universality of QC is very important theoretically, a quantum computer need not be universal to be useful: a special purpose device intended for specific computational tasks that is not universal can be much more experimentally feasible in the near term, while still offering

significant improvements beyond classical hardware. A well studied example is that of quantum simulation (QS). First proposed in the early 1980s ([Manin, 1980](#); [Feynman, 1982](#)) and further developed by [Lloyd \(1996\)](#), QS involves using a more easily controllable quantum system to learn information about a less easily controllable one. The study of which Hamiltonians can be used to simulate which other Hamiltonians goes back to the early 2000s ([Wocjan et al., 2001a,b, 2002](#); [Dodd et al., 2002](#); [Nielsen et al., 2002](#)). Remarkably, it has recently been shown ([Cubitt et al., 2018](#)) that all of the physics of every quantum many-body system is replicated in a certain class of simple, universal quantum spin-lattice models. While the original proposals envisioned breaking the unitary dynamics of the system under study into discrete gates and performing them on a universal quantum computer, more recent work ([Greiner et al., 2002](#); [Manousakis, 2002](#); [Smirnov et al., 2007](#); [Friedenauer et al., 2008](#)) has focused on *continuous-time* QS, in which the Hamiltonian of the system under study is mapped directly onto the Hamiltonian of a special purpose simulator.

Another major area of interest in special purpose QC is that of quantum algorithms for classical optimization problems. Such a problem involves finding the input to a cost function, defined in terms of some collection of real world constraints, that minimizes (either exactly or approximately) the value of that cost function. While some work ([Farhi et al., 2014b](#)) involves mapping the cost function to a gate sequence to run on a digital device, many approaches involve encoding the cost functions into Hamiltonians that induce computationally useful dynamics, a form of continuous-time QC. This has been considered for optimization problems in many fields, including finance ([Marzec, 2016](#); [Orús et al., 2019](#); [Venturelli and Kondratyev, 2019](#)), traditional computer science ([Chancellor et al., 2016a](#); [Choi, 2010, 2011b](#)), machine learning ([Adachi and Henderson, 2015](#); [Denil and de Freitas, 2011](#); [Amin et al., 2018b](#)), scheduling ([Venturelli et al., 2000](#); [Crispin and Syrichas, 2013](#); [Tran et al., 2016](#)), computational biology ([Perdomo-Ortiz et al., 2012](#)), flight gate assignment ([Stollenwerk et al., 2019](#)), air traffic management ([Stollenwerk et al., 2020](#)), and hydrology ([O'Malley, 2018](#)). Many of these CTQC approaches to optimization consider devices that implement the quantum Ising model. I will describe optimization by CTQC in the Ising model in detail in section 2.2. In particular, much of the experimental work has been performed on the D-Wave devices ([D-Wave, 1999–](#)), which are large (a few thousand qubits) QA devices that operate in a highly dissipative, but still quantum mechanical, regime.

More recently, the first credible claim to quantum supremacy was made ([Arute et al., 2019](#)) using a 53-qubit digital gate-model (though too small to be considered universal) device, though performing useful computations of this size (i.e a quantum advantage) is still unlikely to be possible in the near future.

In this thesis, I will present a combination of numerical, analytical, and experimental work to further develop the understanding of CTQC applied to classical optimization problems.

In Chapter 2, I will give an overview of classical combinatorial optimization problems and describe the examples that will be relevant to this thesis. I will then describe CTQC in the Ising model setting more completely, giving the relevant mappings of optimization problems into the model and a detailed explanation of how QWs, AQC, and QA work in this setting. I will also describe the numerical methods I have used for the work in this thesis.

In Chapter 3, I will explore the ability of QWs in the Ising model to solve optimization problems by applying them to the Sherrington-Kirkpatrick spin-glass ground-state problem (see subsection 2.1.3). While studies have previously applied the AQC and QA time-dependent CTQC methods to optimization problems ([Choi, 2010](#); [Crosson et al., 2014](#); [Knysh, 2016](#)), due to their heuristic similarities with classical optimization methods such as simulated annealing ([Kirkpatrick et al., 1983](#)), QWs, which are time-independent, had not been considered for these problem types prior to my work, although a similar approach was being explored by [Hastings \(2019\)](#) around the same time.

In Chapter 4, I will show that the QW results from Chapter 3 generalise to optimization problems beyond the Sherrington-Kirkpatrick spin-glass ground-state problem by applying many of the same numerical analyses to the MAX2SAT problem (see subsection 2.1.4). I will also analyse the performance of AQC on MAX2SAT, and compare to the performance of QW (both at a whole problem level and on a per-instance basis). Finally, I will demonstrate the performance of a powerful classical algorithm, showing the need for finding more advanced CTQC algorithms (this will be one of the aims of Chapter 5).

In Chapter 5, I will extend some of the results from Chapter 3 away from QWs into the *rapid quench* QA regime. While theoretical tools exist to analyse how the slow dynamics of AQC can solve optimization problems there are relatively few tools to understand the opposite limit of rapid quenches. In Chapter 5 I will present several results and tools for this regime. I will show

that sampling from the output state of a rapid quench protocol will always lead to better results than sampling from the input state, and I will present a theoretical framework for determining when this improvement will be substantial. I will also show how these tools can provide efficient heuristic estimates for Hamiltonian parameters, a key requirement for practical application of quantum annealing.

In Chapter 6, I will explore the potential for QA to solve a particular type of optimization problem, known as the shortest vector problem (SVP), that is important for the field of post-quantum cryptography. When [Shor \(1994\)](#) showed that quantum computers would threaten the security of the widely used cryptographic schemes, the need for new ‘post-quantum’ schemes that would not be vulnerable to the same kind of attack became apparent. Many such schemes that have been developed ([Regev, 2005](#); [Gentry, 2009](#); [Jeffrey et al., 1998](#); [Goldreich et al., 1997](#)) rely on difficult lattice-based problems such as SVP. Through experimental work on a D-Wave device, alongside numerical work, I will demonstrate how an Ising model QA/AQC attack on SVP can work, and present some preliminary results regarding its effectiveness.

Chapter 2

Background Theory

In this Chapter, I will give an overview of classical combinatorial optimization and describe the examples that will be relevant to this thesis. I will then describe continuous-time quantum computing (CTQC) in the Ising model setting more completely, giving the relevant mappings of optimization problems into the model and detailed explanations of how computation by continuous-time quantum walk (QW), adiabatic quantum computing (AQC), and quantum annealing (QA) work in this setting. I will also describe the numerical methods I have used for the work in this thesis.

2.1 Optimization Problems

A combinatorial optimization problem can be defined as follows. Given a finite set $\mathcal{S} = \{\mathbf{x}_j\}_{j=1}^N$ of N candidate solutions $\mathbf{x}_j$ (which are denoted in bold-face because they are in general specified by multiple degrees of freedom) and a ‘cost function’ f on $\mathcal{S}$, find an optimal $\mathbf{x}_* \in \mathcal{S}$ such that $f(\mathbf{x}_*) = \min[f(\mathbf{x}_j)]_{j=1}^N$.

Optimization problems need to be solved in a broad range of areas, such as scheduling, route planning, supply chains and finance. This is often computationally intensive, so the prospect of quantum enhanced solution methods is an important research direction for practical quantum computing.

In subsections [2.1.1](#) and [2.1.2](#), I will describe two different kinds of classical optimization al-

gorithm, one which guarantees a true optimal solution in finite time, and one which does not but can produce solutions faster. Then, in subsections 2.1.3, 2.1.4 and 2.1.5, I will define and describe the optimization problems that will be relevant for the rest of the thesis.

2.1.1 Classical algorithms: the branch-and-bound framework

Branch-and-bound (Lawler and Wood, 1966) is an algorithmic framework for solving combinatorial optimization problems by treating the candidate solutions as leaves of a tree-structure, and computing lower bounds on different branches in order to ‘prune’ them away. Correctly-configured branch-and-bound algorithms are guaranteed to find the true optimal solution $\mathbf{x}_*$ of an optimization problem in finite time.

For the branch-and-bound framework, the concept of *partial candidate solutions* is needed. This means specifying only a subset of the degrees of freedom, and leaving the others as free variables. A partial assignment $S \subset \mathcal{S}$ can be expressed as a subset of candidate solutions that are identical in the fixed degrees of freedom, but are otherwise different.

To describe a branch and bound algorithm for solving the problem P , three ingredients are necessary:

- **Cost.** It should be possible to compute the cost function f that defines the problem P . It’s hard to imagine how any algorithm to solve P could exist if f cannot be computed. Moreover, good algorithms require that this can be calculated efficiently.
- **Bound.** Given any partial candidate solution $S \subset \mathcal{S}$, it should be possible to compute a lower bound on the value of the cost function f obtained from any candidate solution $\mathbf{x} \in S$ in the partial solution. It’s usually not possible for this lower bound to be always tight, but tighter bounds will lead to better algorithms. Also, good algorithms require that this bound can be calculated efficiently.
- **Branch.** It should be possible, given a partial solution S , to produce some number of disjoint partial solutions $S_1, S_2, \dots$ such that $S_1 \cup S_2 \cup \dots = S$. Good algorithms require that this can be done efficiently.

With these three ingredients, a branch-and-bound algorithm is structured as follows:

1. Initialise a stack (last in, first out) L containing just the empty partial solution, a variable ‘best_cost’ to store the best cost found so far (typically initialised to ∞) and a variable ‘best_solution’ to store the (full) candidate solution that gave the best cost so far (typically initialised to e.g ‘None’)
2. While L is not empty, do:
 - (a) Pop the next object S from the stack (e.g $S = L.pop()$, S is no longer on the stack)
 - (b) IF STATEMENT:
 - i. If S is a complete candidate solution, compute its **Cost** and possibly update ‘best_cost’ and ‘best_solution’ if appropriate
 - ii. Else if S is only a partial solution, compute the **Bound**. IF STATEMENT:
 - A. If **Bound** $>$ ‘best_cost’, S cannot contain the optimal solution $\mathbf{x}_*$, and can be ruled out, so do nothing and go to next loop iteration
 - B. Else **Bound** $\leq$ ‘best_cost’, so S might contain the optimal solution $\mathbf{x}_*$, so branch to find $S_1, S_2, \dots$, and append these to L .
3. At this point, all candidate solutions $\mathbf{x}_j$ will have either been checked or ruled out by a bound. So, return ‘best_solution’ and ‘best_cost’.

Algorithms structured this way are guaranteed to find the optimal solution. It can be modified in various ways, for example by initialising the ‘best...’ variables with some known guess to start things off, or by using some sort of (first in, first out) queue or priority (best-first) queue instead of a stack for L .

Recently, [Montanaro \(2020\)](#) discovered a way to accelerate branch-and-bound algorithms using gate-model quantum computing. This quantum algorithmic framework can essentially take any classical branch-and-bound algorithm and produce a quantum version that achieves an almost quadratic speedup.

2.1.2 Classical algorithms: Simulated Annealing

Simulated annealing (SA) is a probabilistic technique for solving optimization problems ([Kirkpatrick et al., 1983](#)) that does not guarantee the true optimal solution $\mathbf{x}_*$ (except in the infinite

runtime limit), but can sometimes produce solutions faster than approaches like branch-and-bound which do have this guarantee.

To describe a SA type algorithm, four ingredients are needed:

- **Cost.** It should be possible to compute the cost function f that defines the problem P . It's hard to imagine how any algorithm to solve P could exist if f cannot be computed. Moreover, good algorithms require that this can be calculated efficiently.
- **Temperature.** It is necessary to define a monotonically decreasing simulated temperature. $T_0 = \mathbf{Temperature}(0)$ should be some moderate to large value, while $\mathbf{Temperature}(k_{\max})$ (where k indexes the iterations of the algorithm) should be some small value. A typical choice might be $\mathbf{Temperature}(k) = \frac{T_0}{\ln k}$
- **Neighbour.** It should be possible to efficiently choose a random 'neighbour' of a given candidate solution. A typical method of finding a neighbour might be found by randomly choosing one of the degrees of freedom and changing it by one step (e.g, a single bit-flip in the case where the candidate solutions are binary strings).
- **Probability.** It should be possible to efficiently compute an acceptance probability for a move from candidate solution state a to some other candidate solution b , based on the **Costs** of these candidates. A typical choice is to use Boltzmann distribution, giving $\mathbf{Probability}[\mathbf{Cost}(b), \mathbf{Cost}(a)] = \exp \left[\frac{\mathbf{Cost}(a) - \mathbf{Cost}(b)}{\mathbf{Temperature}} \right]$ if $\mathbf{Cost}(b) > \mathbf{Cost}(a)$ and 1 otherwise.

With these four ingredients, the structure of a SA type algorithm is as follows:

1. Initialise a 'current_state' variable with a random choice of a candidate solution $\mathbf{x}_{k=0} \in \mathcal{S}$
2. For $k = 1$ to $k_{\max}$:
 - (a) set the temperature variable $T = \mathbf{Temperature}(k)$
 - (b) Pick a new state: $\mathbf{new_state} = \mathbf{Neighbour}(\mathbf{current_state})$
 - (c) IF STATEMENT:
 - i. If $\mathbf{Cost}(\mathbf{new_state}) \leq \mathbf{Cost}(\mathbf{current_state})$, accept the transition:
 $\mathbf{current_state} = \mathbf{new_state}$.

- ii. Else, accept the transition (`current_state` = `new_state`) with probability $\text{Probability}[\mathbf{Cost}(\text{new_state}), \mathbf{Cost}(\text{current_state})]$. If rejecting, leave `current_state` unchanged

3. Return `current_state`

With sensible choices for the **Temperature**, **Neighbour** and **Probability**, SA algorithms are guaranteed to produce the optimal solution $\mathbf{x}_*$ in this limit of large $k_{\max}$. In practice, however, it might be preferable to use a $k_{\max}$ that is small enough to be practical, perform a number of repetitions and use the most optimal solution returned. This does not guarantee the true optimal solution, however. Furthermore, similar but more advanced methods have been developed, such as parallel tempering (PT). In PT, many different copies of the simulation are kept at different temperatures, and a probabilistic rule is used to swap configurations between the different simulations.

2.1.3 Sherrington-Kirkpatrick spin glass

The Sherrington-Kirkpatrick (SK) spin glass Hamiltonian H_{SK} ([Sherrington and Kirkpatrick, 1975](#)) is defined on n spins as

$$H_{\text{SK}} = -\frac{1}{2} \sum_{(j \neq k)=0}^{n-1} J_{jk} S_j S_k \quad (2.1)$$

where S_j are the classical spins ($S_j \in \{-1, 1\}$) and the couplings J_{jk} are drawn independently from the normal distribution $\mathcal{N}(\mu, \sigma_{\text{SK}}^2)$ with mean μ and variance σ_{SK}^2 . Finding the ground state of this Hamiltonian is NP-hard ([Choi, 2010](#)), and *uniformly* hard, due to its finite-temperature phase transition ([Sherrington and Kirkpatrick, 1975](#)).

It is computationally convenient to break the spin inversion symmetry by adding single-body field terms of the form $\sum_{j=0}^{n-1} h_j S_j$, where h_j are the field strength values. Like the couplings J_{jk} , the fields h_j are also drawn independently from $\mathcal{N}(\mu, \sigma_{\text{SK}}^2)$. When the fields strengths h_j are drawn from the same distribution as the coupling strengths J_{jk} , the hardness of finding the ground state follows directly from the hardness of the $h_j = 0$ case. The SK spin glass with such fields is mathematically equivalent to a zero field spin glass with one more spin which is “fixed”

in one orientation. This is not true in general for different distributions of field strength h_j . There are known examples in which fields can destroy spin glass behaviour (see, e.g., [Young and Katzgraber, 2004](#); [Feng et al., 2014](#)). In particular, if the field strengths are much larger than the coupling strengths ($|h_j| \gg |J_{jk}|$ for all j, k), then the energy is minimized trivially when all the spins each minimize the energy with respect to their individual fields. While the distribution of field strengths could be used to tune the problem hardness, I will not use it this way in this thesis; I will only consider cases where the field and coupling strengths are drawn from the same distribution.

Due to the way I have defined H_{SK} , “un-fixing” the spin which corresponds to the fields (making all states two-fold degenerate and converting the system to a double cover of the original system), the new couplings will effectively be on average stronger than the rest by a factor of $\sqrt{2}$. As this increase in coupling strength does not scale with the number of spins, it is going to become less and less significant as the size of the system is scaled up, and the hardness will be preserved.

2.1.4 MAX2SAT

Satisfiability problems are an important and well-known class of decision and optimization problems, many of which are known to be NP-complete or NP-Hard ([Karp, 1972](#)). One of the simplest to define satisfiability problems is MAX2SAT, which is known to be NP-Hard ([Garey et al., 1976](#)).

MAX2SAT can be defined as follows. Let $\{v_j\}_{j=1}^n$ be a set of n binary variables $v_j \in \{0, 1\}$. Let $l_j \in \{v_j, \neg v_j\}$ denote a ‘literal’ involving the variable v_j where $\neg$ is a logical NOT (i.e. $\neg 0 = 1$ and $\neg 1 = 0$). A two-literal inclusive-OR clause on literals l_j and l_k can be written $c_a = l_j^{(a)} \vee l_k^{(a)}$ (a just indexes the clause and will be relevant later) where $\vee$ is the logical inclusive-OR (i.e. $0 \vee 0 = 0$ and $0 \vee 1 = 1 \vee 0 = 1 \vee 1 = 1$). A MAX2SAT problem instance on n binary variables can be defined by a set $\{c_a\}_{a=1}^m$ of m inclusive-OR clauses and asks the question “what is the maximum number of the m clauses $\{c_a\}_{a=1}^m$ that can be satisfied (i.e. can be made to evaluate to 1) by any assignment of the n binary variables $\{v_j\}_{j=1}^n$?”

2.1.5 Shortest vector problem

A full-rank lattice $\mathcal{L} \subset \mathbb{R}^N$ can be defined as the discrete set of all integer linear combinations of some set of N linearly independent basis vectors $\mathbf{B} = \{\mathbf{b}_1, \dots, \mathbf{b}_N\}$; that is,

$$\mathcal{L} = \left\{ \sum_{i=1}^N x_i \mathbf{b}_i \right\} = \{\mathbf{B} \cdot \mathbf{x} : \mathbf{x} \in \mathbb{Z}^N\}. \quad (2.2)$$

Fig 2.1 shows an example of a lattice in two dimensions.

Every lattice contains the zero vector, denoted $\mathbf{0}$, for which $\mathbf{0} \cdot \mathbf{b}_j = \mathbf{0} \forall 1 \leq j \leq N$. The set $\mathbf{B}$ of N linearly independent basis vectors together is known collectively as ‘a basis’. Each lattice has infinitely many bases, and every basis can be mapped to every other basis by a unimodular transformation (that is, a transformation whose matrix has determinant ± 1). There is a notion of good and bad bases, where good means that the basis vectors are quite orthogonal and quite short, where ‘quite’ varies according to the context, but can be defined in terms of bounds on the angles between basis vectors or ratios of their lengths. Knowledge of the specifics of these conditions will not be important in this thesis, but the key concept to appreciate is that *short and close-to-orthogonal is good*.

Necessarily, there are only a small number of good bases (a finite number by any definition of good), and infinitely-many bad bases. The subject of turning arbitrary bad lattice bases into ‘good enough’ bases is a widely studied topic at the centre of lattice cryptanalysis ([Lenstra et al., 1982](#); [Helfrich, 1985](#); [Schnorr and Euchner, 1994](#)).

With the term ‘lattice’ defined, the shortest vector problem can be defined as follows: given a basis $\mathbf{B} = \{\mathbf{b}_1, \dots, \mathbf{b}_n\}$ describing a lattice $\mathcal{L}$, find the closest (non-zero) lattice point to the origin,

$$\mathbf{v}_{\text{shortest}} = \operatorname{argmin}\{\|\mathbf{v}\| : \mathbf{v} \in \mathcal{L} \setminus \{\mathbf{0}\}\}. \quad (2.3)$$

This can alternatively be stated as finding the shortest distance between any two distinct points in the lattice.

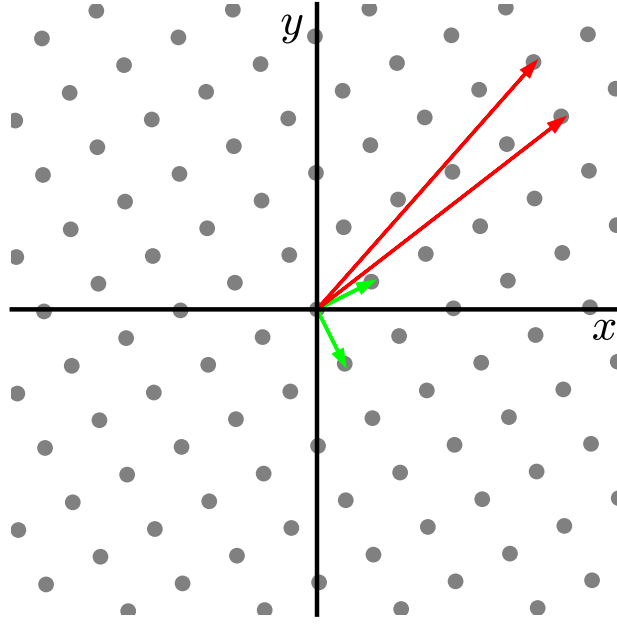


Figure 2.1: An example of a two-dimensional lattice with lattice points depicted in black. Red arrows represent a bad basis with long vectors that are far from orthogonal (though still linearly-independent); green arrows are a good basis for the same lattice, with short and highly orthogonal vectors. Both bases (by definition) span the lattice.

2.2 CTQC: Ising model and problem mapping

One way to tackle optimization in a quantum setting is to use a device which realises an Ising Hamiltonian with a transverse field. Computing using the Ising Hamiltonian works as follows: The optimization problem is encoded into the Ising Hamiltonian $\hat{H}_I$

$$\hat{H}_I = - \sum_{(j \neq k)=0}^{n-1} J_{jk} \hat{Z}_j \hat{Z}_k - \sum_{j=0}^{n-1} h_j \hat{Z}_j, \quad (2.4)$$

on n qubits, such that the solution corresponds to the ground state of $\hat{H}_I$. In this notation, the operator $\hat{Z}_j$ on the full Hilbert space applies the single qubit Pauli- Z operator $\hat{Z}$ to the j th qubit,

$$\hat{Z}_j = \left(\bigotimes_{r=0}^{j-1} \hat{\mathbb{1}}_2 \right) \otimes \hat{Z} \otimes \left(\bigotimes_{r=j+1}^{n-1} \hat{\mathbb{1}}_2 \right), \quad (2.5)$$

where $\hat{\mathbb{1}}_2$ is the identity operator on a single qubit. The (real) values of the coupling strengths J_{jk} and fields h_j define the optimization problem, and efficient methods are known for expressing optimization problems in terms of these coupling and field strengths (e.g., [Choi, 2010](#)). The

transverse field term $\hat{H}_T$

$$\hat{H}_T = -\Gamma \sum_{j=0}^{n-1} \hat{X}_j, \quad (2.6)$$

drives transitions between states, where Γ is a real-valued transverse field strength, and $\hat{X}_j$ is the operator on the full Hilbert space that applies the single qubit Pauli- X operator to the j th qubit, defined by analogy with $\hat{Z}_j$ in Eq. (2.5). The qubits are initialised in the ground state of $\hat{H}_T$, which is easy to do by applying a strong transverse field to align all the qubits in the state $|+\rangle = 2^{-1/2}(|0\rangle + |1\rangle)$. Then, the computation is carried out by applying the full transverse Ising Hamiltonian

$$\hat{H}_{\text{TI}}(t) = A(t)\hat{H}_T + B(t)\hat{H}_I, \quad (2.7)$$

where t is time and $A(t)$, $B(t)$ are real-valued control functions. To obtain a candidate solution to the optimization problem, the qubit register is measured after a time t_f . For some problems, sampling from the distribution of low energy states provides the required solution – this can be done by repeating the computation, which will in general not produce the lowest energy state with certainty.

The Ising Hamiltonian is a natural choice for encoding problems for two reasons. First, it is proven to be universal for classical problems (De las Cuevas and Cubitt, 2016). There are efficient methods for mapping NP-hard optimization problems to the Ising model (Lucas, 2014; Choi, 2010), providing a practical route to quantum algorithms. Since many optimization problems are NP-hard, an exponential speed up is not expected, but even modest polynomial improvements are useful for practical applications. There is increasing interest in how to obtain polynomial advantages through quantum algorithms (Moylett et al., 2017; Montanaro, 2018; Ambainis et al., 2019). Interesting results have been presented for a wide range of applications, such as mathematics (Bian et al., 2013; Li et al., 2017), computer science (Chancellor et al., 2016a), computational biology (Perdomo-Ortiz et al., 2012), finance (Marzec, 2016), and aerospace (Coxson et al., 2014). Second, the Ising Hamiltonian can be implemented in a range of different physical systems. The quantum Ising Hamiltonian is the basic interaction Hamiltonian in the D-Wave Systems Inc. programmable superconducting devices (D-Wave, 1999–; Boixo et al., 2013; Johnson et al., 2011). Implementations in other promising architectures

include Rydberg systems (Bernien et al., 2017a) and trapped ions (Kim et al., 2011). The Ising Hamiltonian is also the basic tool for specialised optimization hardware, such as coherent Ising machines (Inagaki et al., 2016; McMahon et al., 2016). Optimization using the Ising Hamiltonian can be implemented in digital quantum architectures by using the quantum approximate optimization algorithm (QAOA) (Farhi et al., 2014b,a; Marsh and Wang, 2019) or quantum alternating operator ansatz (Hadfield et al., 2019). Studies by Zhou et al. (2018) show how to exploit non-adiabatic effects in QAOA on early quantum hardware.

There are several known methods for driving the quantum system from its initial state into the ground state of a Hamiltonian defining the problem to be solved. These methods correspond to different choices for the control functions $A(t)$ and $B(t)$ in Eq. (2.7). QWs evolve the system under a time-independent Hamiltonian for a suitable time before measurement of the final state. The QW will be described in detail in section 2.4. AQC (Farhi et al., 2000, 2001) keeps the quantum system in the ground state while the initial Hamiltonian is slowly changed into the problem Hamiltonian. AQC will be described in detail in section 2.5. QA (Finnila et al., 1994; Kadowaki and Nishimori, 1998) takes advantage of open quantum systems effects to cool the system towards the ground state. QA will be described in detail in section 2.6.

Computation by QW and AQC are end points of a family of continuous-time protocols that use the same Hamiltonian terms but are applied with different time dependent modulation (Morley et al., 2019).

2.2.1 Ising mapping of the Sherrington-Kirkpatrick ground-state problem

The mapping of the SK ground-state problem described in subsection 2.1.3 into the quantum Ising model is almost trivial: the classical spin variables S_j in Eq. (2.1) are simply mapped to Pauli-Z operators. Thus, the problem Hamiltonian $\hat{H}_{\text{SK}}$ becomes

$$\hat{H}_{\text{SK}} = -\frac{1}{2} \sum_{(j \neq k)=0}^{n-1} J_{jk} \hat{Z}_j \hat{Z}_k - \sum_{j=0}^{n-1} h_j \hat{Z}_j. \quad (2.8)$$

2.2.2 Ising mapping of the MAX2SAT problem

Mappings of the MAX2SAT problem defined in subsection 2.1.4 are known (Lucas, 2014; Choi, 2010), and they are more involved than the mapping for SK. I will present such a mapping here. The first step is to map the n binary variables on to n qubits by making the following choice:

$$\hat{Z}_j |v_1, v_2, \dots, v_{n-1}, v_n\rangle = (1 - 2v_j) |v_1, v_2, \dots, v_{n-1}, v_n\rangle$$

where $\hat{Z}_j$ is the Pauli Z operator on qubit j . That is, one can identify the j^{th} qubit being in the $+1$ eigenstate of its $\hat{Z}_j$ operator with the variable assignment $v_j = 0$, and similarly identify the j^{th} qubit being in the -1 eigenstate of its $\hat{Z}_j$ operator with the variable assignment $v_j = 1$. With this definition, it is possible to map the MAX2SAT problem instance onto these qubits as follows.

A single clause c_a can be mapped to an Ising model term $\hat{C}_a$ in the following way:

$$\hat{C}_a \equiv \frac{1}{4} \left(\mathbb{1} - s_j^{(a)} \hat{Z}_j \right) \left(\mathbb{1} - s_k^{(a)} \hat{Z}_k \right) \quad (2.9)$$

$$= \frac{1}{4} \left(\mathbb{1} + s_j^{(a)} s_k^{(a)} \hat{Z}_j \hat{Z}_k - s_j^{(a)} \hat{Z}_j - s_k^{(a)} \hat{Z}_k \right) \quad (2.10)$$

where $\mathbb{1}$ is the identity operator and $s_j^{(a)}$ is a ‘sign’ defined as

$$s_j(a) = \begin{cases} 1 & \text{if variable } v_j \text{ is not negated in literal } l_j^{(a)} \\ -1 & \text{if variable } v_j \text{ is negated in literal } l_j^{(a)} \end{cases}.$$

Careful inspection of $\hat{C}_a$ (first line, Eq. (2.9) is clearest) will reveal that

$$\hat{C}_a |v_1, v_2, \dots, v_{n-1}, v_n\rangle = 0 \times |v_1, v_2, \dots, v_{n-1}, v_n\rangle \quad (2.11)$$

if the variables v_j and v_k in the state $|v_1, v_2, \dots, v_{n-1}, v_n\rangle$ satisfy the clause c_a , and

$$\hat{C}_a |v_1, v_2, \dots, v_{n-1}, v_n\rangle = 1 \times |v_1, v_2, \dots, v_{n-1}, v_n\rangle \quad (2.12)$$

if the variables v_j and v_k in the state $|v_1, v_2, \dots, v_{n-1}, v_n\rangle$ fail to satisfy the clause c_a .

The full problem Hamiltonian $\hat{H}_{\text{M2S}}$ can now be defined by simply summing of the clause operators $\hat{C}_a$ to get

$$\hat{H}_{\text{M2S}} = \sum_{a=1}^m \hat{C}_a. \quad (2.13)$$

The eigenstates of $\hat{H}_{\text{M2S}}$ are the states $|v_1, v_2, \dots, v_{n-1}, v_n\rangle$ and, due to Eqs. (2.11) and (2.12), the eigenvalue of the state $|v_1, v_2, \dots, v_{n-1}, v_n\rangle$ ranges between 0 and m and ‘counts’ the number of clauses that the variable assignment fails to satisfy. This means that any ground-state of $\hat{H}_{\text{M2S}}$ corresponds to a variable assignment that maximises the number of satisfied clauses. Thus, finding such a ground-state solves the MAX2SAT problem.

2.3 CTQC: Unstructured search problem

A simple example of a problem that can be solved in the CTQC setting is the search problem. The problem is to find the marked state, a single bit-string $m \in \{0, 1\}^n$ out of $N = 2^n$ possible bit strings. Finding a marked state was shown to have a gate-model quantum algorithm with a quadratic speed up over classical algorithms by [Grover \(1996\)](#). To map this problem to the continuous-time Hamiltonian setting, the marked basis state $|m\rangle$ is given one less unit of energy than all the rest of the basis states, by defining the problem Hamiltonian $\hat{H}_S$ as

$$\hat{H}_S = \mathbb{1} - |m\rangle\langle m|. \quad (2.14)$$

By construction, the problem Hamiltonian $\hat{H}_S$ has the marked state $|m\rangle$ as its ground state.

This construction will be used in sections 2.4 and 2.5 to show that the problem can be solved by QW and AQC, respectively, with the same quantum speedup as in the gate-model.

2.4 CTQC: Quantum Walks

While both discrete (coined) quantum walks ([Aharonov et al., 2001](#); [Shenvi et al., 2003](#)) and continuous-time quantum walks ([Farhi and Gutmann, 1998](#); [Childs et al., 2003](#)) are used for computation, in this thesis I will only discuss the continuous-time quantum walk.

2.4.1 Continuous-time quantum walks

A QW is defined on an undirected graph $G(V, E)$, with $V = \{j\}_{j=0}^{N-1}$ the set of N vertex labels and E the set of label-pairs (j, k) associated with edges. The vertices correspond to the positions of the walker, and the edges indicate the allowed transitions between vertices. This is conveniently encoded in the adjacency matrix A of the graph, which has entries $A_{jk} = 1$ for $(j, k) \in E$ and $A_{jk} = 0$ otherwise. The Laplacian of G is $L = A - D$, where D is a diagonal matrix formed from the degree of each vertex, $D_{jj} = \deg(j)$, where $\deg(j)$ is the number of edges connected to vertex j . Both the adjacency matrix A and Laplacian L are symmetric matrices which can thus be used to define a quantum Hamiltonian for the dynamics of the QW on the graph. In this thesis, only regular graphs are needed, for which $\deg(j)$ is constant with respect to j . For regular graphs, the only difference between using the adjacency matrix A or Laplacian L is an irrelevant global phase (Childs and Goldstone, 2004). The quantum walk Hamiltonian $\hat{H}_G$ for a quantum walk on graph G can be defined by

$$\langle j | \hat{H}_G | k \rangle = -\gamma L_{jk}, \quad (2.15)$$

where γ is the hopping rate between connected vertices per unit time. The states $|j\rangle, |k\rangle$ for $j, k \in V$ are associated with the vertices of G and form a basis for a Hilbert space of dimension N . In the Ising model context, the dimension of the Hilbert space is $N = 2^n$ where n is the number of qubits, and $\{|j\rangle\}_{j=0}^{N-1}$ is the computational basis. For a quantum walk starting in state $|\psi(0)\rangle$, the state of the walker evolves according to the Schrödinger equation, with formal solution

$$|\psi(t)\rangle = \exp\{-i\hat{H}_G t\} |\psi(0)\rangle, \quad (2.16)$$

using units in which $\hbar = 1$.

2.4.2 Comparison to classical random walk

Quantum walks can, under some circumstances, explore a graph G significantly faster than the classical counterpart on the same graph, as shown in (Farhi and Gutmann, 1998) in the context of decision tree graphs. In this subsection, I will show this phenomenon numerically for a very simple case: the quantum and classical walks on the line.

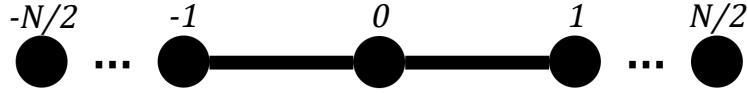


Figure 2.2: Finite line with $N + 1$ vertices (for even N), centred on the vertex with index 0.

On the line, the vertex with index j is connected by edges to the vertices with indices $j - 1$ and $j + 1$ (except the end vertices in the case of the finite line, which are only connected to one other vertex). For the purposes of this demonstration, I will consider the finite line with $N + 1$ vertices (for even N), centred on the vertex with index 0. This graph is illustrated in Fig. 2.2.

The walk Hamiltonian $\hat{H}_G$ for this given by

$$\langle j | \hat{H}_G | j \rangle = \begin{cases} \gamma & \text{for } |j| = N/2 \\ 2\gamma & \text{otherwise} \end{cases} \quad (2.17)$$

$$\langle j | \hat{H}_G | k (\neq j) \rangle = \begin{cases} -\gamma & \text{if } |j - k| = 1 \\ 0 & \text{otherwise.} \end{cases} \quad (2.18)$$

Following (Farhi and Gutmann, 1998), a continuous-time classical random walk can be defined by setting the (classical) probability $p_{ml}(t)$ for the walker to go from vertex l to vertex m in time t when moving between connected vertices with a probability per unit time γ to

$$p_{ml}(t) = \langle m | e^{-\hat{H}_G t} | l \rangle. \quad (2.19)$$

The dynamics of the quantum walk are given by Eq. (2.16). Choosing the initial state to be $|\psi(0)\rangle = |0\rangle$ in both the quantum and classical cases, the dynamics can be simulated.

I have simulated such a classical random walk and quantum walk on a line with $N = 521$. Fig. 2.3(a) shows the probability $P(t_f)$ of a measurement at time $t_f = \frac{100}{\gamma}$ resulting in finding the walker at vertex x , in the classical (red) and quantum (blue) cases. It can be seen that, while the probability for the classical walk is concentrated around the 0 vertex with very little probability beyond $|x| = 50$, the quantum walk extends much further from the the 0 vertex, with significant probability around $|x| = 200$.

Fig. 2.3 further illustrates the faster spreading of the quantum walk by showing the expectation distance $\langle |x| \rangle$ from the 0 vertex as a function of measurement time t_f . It can be seen that the

expected distance increases linearly with time in the quantum case (ballistic dynamics), but only increases with $\sqrt{t_f}$ in the classical case (diffusive dynamics).

2.4.3 Computing using a quantum walk

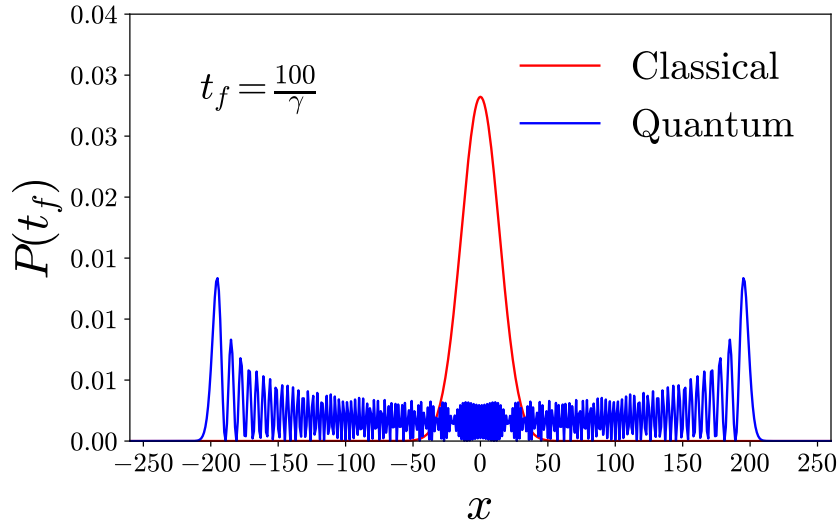
In subsection 2.4.2, it was demonstrated that a quantum walk can, in some circumstances, explore a graph significantly faster than its classical counterpart on the same graph. This observation forms part of the heuristic justification for using quantum walks as a basis for quantum algorithms for optimization problems. In this subsection, I describe the general framework for doing this.

The task is to solve an optimization problem whose $N = 2^n$ candidate solutions j are represented in the computational basis $\{|j\rangle\}_{j=0}^{N-1}$, where j is a bit string corresponding to the state of n qubits. The problem is encoded in an Ising Hamiltonian $\hat{H}_P$, of the form described by $\hat{H}_I$ in Eq. (2.4) and whose eigenbasis is the computational basis. I write the basis state with eigenvalue $E_a^{(P)}$ as $|E_a^{(P)}\rangle$, with $a \in \{0 \dots N-1\}$, and adopt the convention that $E_a^{(P)} \leq E_{a+1}^{(P)}$. In other words, $\{|E_a^{(P)}\rangle\}_{a=0}^{N-1}$ is a reordering of $\{|j\rangle\}_{j=0}^{N-1}$ based on the corresponding eigenenergies of $\hat{H}_P$. The encoding is chosen such that the solution corresponds to the ground state $|E_0^{(P)}\rangle$ of the problem Hamiltonian $\hat{H}_P$.

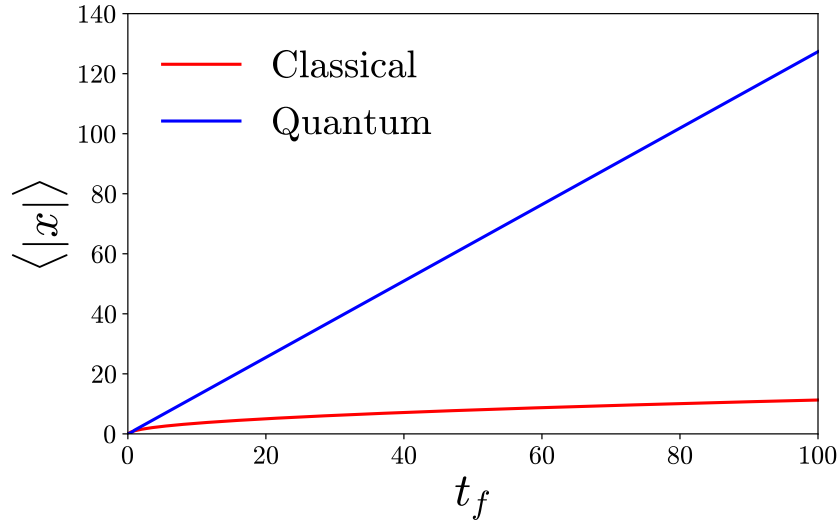
To use a quantum walk to solve the problem, it is first necessary to choose a suitable state in which to initialize the system. With no prior knowledge of the solution, the equal superposition of all basis states

$$|\psi(0)\rangle = N^{-1/2} \sum_{j=0}^{N-1} |j\rangle, \quad (2.20)$$

is a sensible choice that avoids bias. More generally, the initial state can be prepared as weighted or biased superposition, to incorporate prior knowledge about the solution (Perdomo-Ortiz et al., 2011; Duan et al., 2013; Chancellor, 2017; Graß and Lewenstein, 2017; Baldwin and Laumann, 2018; Kechedzhi et al., 2018; Graß, 2019). Next, a suitable walk graph G must be chosen. The main requirement is that the ground state of the quantum walk Hamiltonian $\hat{H}_G$ coincides with the initial state, either biased or unbiased (see section 3.3.2). A simple way to achieve a biased starting state would be to ‘tilt’ the driver fields so they are no longer completely transverse. Except where I explicitly state otherwise in Chapter 5, I will only



(a)



(b)

Figure 2.3: (a) Probability that a measurement at time $t_f = \frac{100}{\gamma}$ after a classical random walk (red) and quantum walk (blue) (both continuous-time) on a line of size $N = 521$ will find the particle at vertex x . It can be seen that the quantum walk is much more likely to be far from the origin than the classical random walk. (b) Expectation value $\langle |x| \rangle$ of the distance of the particle from the origin at time t_f . The expected distance increases linearly with time in the quantum case (ballistic dynamics), but only increases with $\sqrt{t_f}$ in the classical case (diffusive dynamics).

consider the unbiased case in this thesis. The full Hamiltonian $\hat{H}$ is defined by adding the (scaled version of) quantum walk Hamiltonian $\gamma\hat{H}_G$ to the problem Hamiltonian $\hat{H}_P$

$$\hat{H} \equiv \gamma\hat{H}_G + \hat{H}_P, \quad (2.21)$$

where the key parameter is the hopping rate γ . The computation is performed by evolving the initial state (Eq. (2.20)) under the full Hamiltonian $\hat{H}$ for a time t_f , then measuring the qubit register in the computational basis. The intuition, based on the faster spreading of quantum walks over classical described in subsection 2.4.2, is that the quantum walk dynamics provide rapid exploration of the basis states, while the energy structure of the problem Hamiltonian $\hat{H}_P$ causes localisation around low-energy states.

The success probability $P(t_f) = \left| \left\langle E_0^{(P)} \middle| \psi(t_f) \right\rangle \right|^2$ of finding the solution state when measuring will not in general be unity. It will typically be necessary to repeat the protocol multiple times to obtain a high probability of success over all the repeats.

As I will show in subsection 2.4.5, QWs can solve the search problem (Childs and Goldstone, 2004), achieving the same quadratic $O(N^{1/2})$ quantum speed up as is obtained by Grover's algorithm (Grover, 1996). For particular graphs, QWs can solve problems exponentially faster (e.g., Childs et al., 2003), and they are now widely used as subroutines in more complex quantum algorithms. However, in the continuous-time setting, the application of QWs to optimization problems has not been studied in detail.

2.4.4 Graph choice for quantum walk computing

There are many graph-based Hamiltonians with the initial state $|\psi(0)\rangle$ defined in Eq. (2.20) as the ground state. A common choice is the *complete graph* K , in which every vertex is connected to every other. This graph has the quantum walk Hamiltonian $\hat{H}_K$ that couples every computational basis state $|j\rangle$ state to every other,

$$\begin{aligned} \hat{H}_K &= N\mathbb{1} - \sum_{j,k=0}^{N-1} |k\rangle\langle j| \\ &= N[\mathbb{1} - |\psi(0)\rangle\langle\psi(0)|]. \end{aligned} \quad (2.22)$$

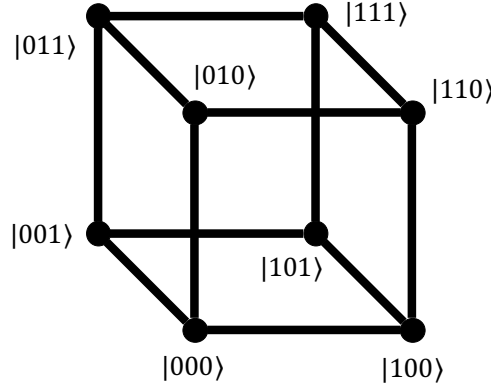


Figure 2.4: A 3-dimensional hypercube (a cube) graph in which the vertices are labeled by the $2^3 = 8$ computational basis states of 3-qubits, and the edges connect the states with Hamming distance 1 (single spin flips).

The complete graph is useful because it makes some algorithms analytically tractable (see, e.g., [Childs and Goldstone, 2004](#)). However, for implementation on qubit-based hardware, the complete graph is not in general practical, requiring higher order interaction terms than the transverse Ising term (Eq. (2.6)). In this qubit setting, an implementation of the complete graph requires a sum over every one-body term (e.g. $\hat{X}_j$), every two-body term (e.g. $\hat{X}_j\hat{X}_k$), every three-body term (e.g. $\hat{X}_j\hat{X}_k\hat{X}_l$) ... up to the n -body term $\prod_{j=0}^{n-1} \hat{X}_j$, a total of N terms. One- and two-body terms are relatively easy to implement, since they correspond to Hamiltonians found naturally. Terms in three or more Pauli- X operators are much more difficult and generally require extra qubits to engineer in real physical systems.

A more natural choice of graph for qubits is the hypercube. The n -bit labels are associated with the vertices of the graph such that the edges correspond to flipping one bit, as illustrated in figure 2.4. The hypercube quantum walk Hamiltonian $\hat{H}_h$ on n qubits is composed of single-body terms

$$\hat{H}_h = n\mathbb{1} - \sum_{j=0}^{n-1} \hat{X}_j. \quad (2.23)$$

With $\hat{H}_h$ as the graph Hamiltonian, the full quantum walk computational Hamiltonian $\hat{H}$ defined in Eq. (2.21) is a transverse Ising Hamiltonian in the form of $\hat{H}_{\text{TI}}$ in Eq. (2.7), with the control functions $A(t)$ and $B(t)$ kept constant throughout the computation. In this thesis, I will almost always be using the hypercube graph.

2.4.5 Solving the search problem using quantum walks

The QW search problem has been analytically solved (Childs and Goldstone, 2004) for several different walk graphs. For the complete graph and the hypercube graph, a quantum speed up is obtained for carefully chosen optimal values of the hopping rate γ . For the complete graph Hamiltonian $\hat{H}_K$, the optimal value is $\gamma_{\text{opt}}^{(K)} = 1/N$, while for the hypercube Hamiltonian, $\hat{H}_h$, the optimal hopping rate $\gamma_{\text{opt}}^{(h)}$ is given by

$$2\gamma_{\text{opt}}^{(h)} = \frac{1}{N} \sum_{r=1}^n \binom{n}{r} \frac{1}{r}, \quad (2.24)$$

where $\binom{n}{r} = \frac{n!}{r!(n-r)!}$ is the binomial coefficient. For a quantum speed up, the hopping rate must be set to $\gamma_{\text{opt}}^{(h)}$ as defined by Eq. (2.24) with high precision. It has been shown (Morley et al., 2019) that the fractional tolerance to misspecification of the optimal hopping rate $\gamma_{\text{opt}}^{(h)}$ falls as $O(N^{-1/2})$.

The measurement time must also be chosen appropriately. In the limit of large problem size N , the marked state can be found with unit success probability, $\lim_{N \rightarrow \infty} [P(t_f^{(\text{opt})})] = 1$, by measuring in the computational basis at an optimal measurement time $t_f^{(\text{opt})}$. For both the hypercube and complete graphs, the optimal time $t_f^{(\text{opt})}$ scales with the square-root of the problem size N as $t_f^{(\text{opt})} \simeq \frac{\pi}{2} N^{1/2}$. This corresponds to a quadratic speed up compared to the best classical algorithm. Due to the absence of structure in the search problem specifically, such a quadratic speed up has been proven to be the best possible quantum speed up (Bennett et al., 1997).

The variation of $P(t_f)$ with t_f is shown in Figure 2.5(a) for search on hypercube graphs of size $N = 2^{30}$ (i.e., $n = 30$ qubits) and $N = 2^{11}$ (i.e., $n = 11$ qubits), using the optimal hopping rate $\gamma_{\text{opt}}^{(h)}$. The sinusoidal oscillations of the probability $P(t_f)$ occur because the quantum walk is performing Rabi oscillations between the initial state and the marked state. The two lowest energy levels of the full Hamiltonian $\hat{H}$ with varying γ undergo an avoided level crossing at $\gamma_{\text{opt}}^{(h)}$ and the associated eigenstates $|E_0\rangle$ and $|E_1\rangle$ are approximately the orthogonal equal superpositions of the starting state and marked state, $|E_{0,1}\rangle \simeq (|\psi(0)\rangle \pm |m\rangle)/2^{1/2}$. The gap $E_1 - E_0$ scales with the problem size N as $O(N^{-1/2})$ (Childs and Goldstone, 2004).

These simple, two-level dynamics describe the quantum walk solution to the search problem

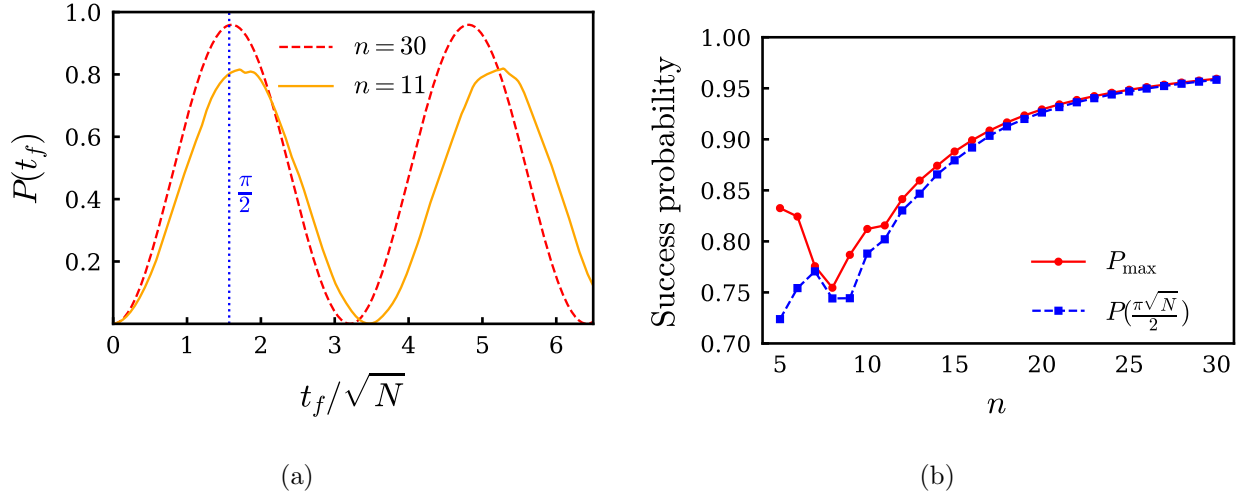


Figure 2.5: The search problem solved using a QW on the hypercube using the optimal hopping rate $\gamma_{\text{opt}}^{(h)}$ given by Eq. (2.24). (a) The probability $P(t_f)$ that a measurement at time t_f results in successfully finding the marked state $|m\rangle$ for two different numbers $n = 30$ (red, dashed line) and $n = 11$ (orange, solid line) of qubits (i.e problem sizes $N = 2^{30}$ and $N = 2^{11}$ respectively). (b) Comparison of instantaneous success probabilities $P(t_f)$ at the asymptotically optimal (blue squares, dashed line) and numerically determined best (red circles, solid line) measurement times t_f .

well for large problem size N : the oscillations in the $N = 2^{30}$ case have no visible irregularities. For smaller sizes, finite-size effects due to population of higher energy levels are apparent: the oscillations in the $N = 2^{11}$ case have lower probability peaks and show some irregular behaviour, such as the small dip on the first peak. These finite-size effects are further illustrated in Fig. 2.5(b), which shows the instantaneous success probability $P(t_f)$ at the asymptotically optimal and numerically determined best times. Both probabilities show a pronounced dip around $n = 8$ qubits, with smooth behaviour only settling in for $n > 12$ qubits.

The search problem in the CTQC setting has two important drawbacks. Firstly, implementing the problem Hamiltonian $\hat{H}_S$ directly on n qubits requires $O(2^n)$ terms of products of up to n Pauli- Z operators, similar to the problem with implementing the complete-graph Hamiltonian $\hat{H}_K$, defined in Eq. (2.22), on qubits. Implementing higher order Pauli- Z terms can be done using extra qubits as “gadgets”, e.g., (Jordan and Farhi, 2008). An alternative type of gadget, specifically for permutation-symmetric problems like search, is given in (Dodds et al., 2019), building on classical problem mapping techniques in (Chancellor et al., 2016a, 2017). Secondly, it is impossible to map the problem Hamiltonian to qubits without specifying the solution outright. Hence, the search problem serves as a useful toy problem, especially in contexts where having analytic, computational, and physical implementations available for comparisons

facilitates benchmarking and other testbed procedures.

2.5 CTQC: Adiabatic Quantum Computing

AQC is a time-dependent CTQC strategy, first studied experimentally in [Brooke et al. \(1999\)](#); [Brooke and Aeppli \(2001\)](#) (although the work in these references could be more correctly described as quantum annealing, see section 2.6) and formalised into a quantum algorithm in [Farhi et al. \(2000\)](#). In AQC, the system is prepared in the ground state of a problem-independent ‘driver’ Hamiltonian $\hat{H}_{\text{drive}}$ which is chosen specifically so its ground-state is easy to prepare. As with the quantum walks presented in section 2.4, the computational problem to be solved is encoded in a problem Hamiltonian $\hat{H}_P$, diagonal in the computational basis, such that its ground-state corresponds to the solution to the problem. Unlike the time-independent Hamiltonian used for quantum walks, however, AQC is performed with the time-dependent Hamiltonian

$$\hat{H}(t) = [1 - s(t)] \hat{H}_{\text{drive}} + s(t) \hat{H}_P \quad (2.25)$$

where the control function $s(t)$ is known as the ‘schedule’ and satisfies the boundary conditions

$$\begin{aligned} s(0) &= 0 \\ s(t_f) &= 1 \end{aligned} \quad (2.26)$$

where t_f is the total duration of the adiabatic protocol, a parameter that must be chosen. If the total duration t_f is chosen to be sufficiently large (i.e, the schedule proceeds sufficiently slowly), performing a measurement in the computational basis at the end of the protocol at time t_f will result in the ground-state of the problem Hamiltonian $\hat{H}_P$ with success probability $P(t_f) \approx 1$ close to unity. This result is proven in subsection 2.5.1. In fact ([Roland and Cerf, 2002](#)), if t_f is large enough that the inequality

$$\left| \frac{ds}{dt} \right| \leq \epsilon \frac{g^2(s)}{\left| \left\langle \frac{d}{ds} \hat{H}(s) \right\rangle_{2,1} \right|} \quad (2.27)$$

is satisfied, where $\hat{\tilde{H}}(s(t)) = \hat{H}(t)$, $g(s)$ is the gap between the ground-state $E_1(s)$ and first-excited state $E_2(s)$ of $\tilde{H}(s)$, and $\left\langle \frac{d}{ds} \hat{\tilde{H}}(s) \right\rangle_{2,1} = \langle E_2(s) | \frac{d}{ds} \hat{\tilde{H}}(s) | E_1(s) \rangle$, then the success probability will be $P(t_f) = 1 - O(\epsilon)$.

Since then, there have been many advances in the theoretical foundations of adiabatic quantum computing, including proofs that it is universal in certain settings (Kempe et al., 2006; Aharonov et al., 2009), improved versions of the adiabatic condition (Amin, 2009; Cheung et al., 2011; Lidar et al., 2009), and an extension of the adiabatic theorem to open systems (Venuti et al., 2016). A comprehensive review of these and other advances is available in Albash and Lidar (2018a). More recently, it has been shown (Hastings, 2020) that, even when no sign problem exists, there is a superpolynomial oracle separation between adiabatic quantum computing and classical computing. There have also been recent advances in ways to map problems; these include parity AQC (Lechner et al., 2015; Rocchetto et al., 2016; Leib et al., 2016; Chancellor et al., 2017), a collection of methods for encoding more connected graphs than the native hardware connections using parity. These methods provide possibly experimentally-easier alternatives to the more traditional minor embedding techniques (Choi, 2008, 2011a).

2.5.1 Proof of the adiabatic theorem

The Schrodinger equation is

$$\frac{d|\psi(t)\rangle}{dt} = -iH(t)|\psi(t)\rangle. \quad (2.28)$$

Let $|E_j(t)\rangle$ be an eigenstate of $H(t)$ with eigenvalue $E_j(t)$. Then,

$$H(t)|E_j(t)\rangle = E_j(t)|E_j(t)\rangle. \quad (2.29)$$

Writing the state $|\psi(t)\rangle$ in the eigenbasis of $H(t)$ as

$$|\psi(t)\rangle = \sum_j \alpha_j(t) |E_j(t)\rangle. \quad (2.30)$$

and substituting Eq. (2.30) into the Schrodinger equation (Eq. (2.28)) gives

$$\sum_j \left[\frac{d\alpha_j(t)}{dt} |E_j(t)\rangle + \alpha_j(t) \frac{d|E_j(t)\rangle}{dt} \right] = -i \sum_j \alpha_j(t) E_j(t) |E_j(t)\rangle. \quad (2.31)$$

Let $\alpha_j(t) = \beta_j(t)e^{i\theta_j(t)}$ where $\theta_j(t) = -\int_0^t dt' E_j(t')$. Then

$$\frac{d\alpha_j(t)}{dt} = \frac{d\beta_j(t)}{dt} e^{i\theta_j(t)} - i\alpha_j(t) E_j(t). \quad (2.32)$$

Substituting Eq. (2.32) back into Eq. (2.31) then gives

$$\sum_j \left[\frac{d\beta_j(t)}{dt} e^{i\theta_j(t)} |E_j(t)\rangle + \alpha_j(t) \frac{d|E_j(t)\rangle}{dt} \right] = 0.$$

Premultiplying by the eigenstate $|E_k(t)\rangle$ gives

$$\begin{aligned} \frac{d\beta_k(t)}{dt} e^{i\theta_k(t)} &= - \sum_j \left[\alpha_j(t) \langle E_k(t) | \frac{d|E_j(t)\rangle}{dt} \right] \\ \frac{d\beta_k(t)}{dt} &= - \sum_j \left[\beta_j(t) e^{i[\theta_j(t) - \theta_k(t)]} \langle E_k(t) | \frac{d|E_j(t)\rangle}{dt} \right] \\ &= -\beta_j(t) \langle E_j(t) | \frac{d|E_j(t)\rangle}{dt} - \sum_{j \neq k} \left[\beta_j(t) e^{i[\theta_j(t) - \theta_k(t)]} \langle E_k(t) | \frac{d|E_j(t)\rangle}{dt} \right] \end{aligned} \quad (2.33)$$

Now, differentiating Eq. (2.29) gives

$$\left(\frac{d}{dt} H(t) \right) |E_j(t)\rangle + H(t) \left(\frac{d}{dt} |E_j(t)\rangle \right) = \left(\frac{d}{dt} E_j(t) \right) |E_j(t)\rangle + E_j(t) \left(\frac{d}{dt} |E_j(t)\rangle \right)$$

and premultiplying by $\langle E_k(t) |$ (for $k \neq j$) gives

$$\begin{aligned} \langle E_k(t) | \left(\frac{d}{dt} H(t) \right) |E_j(t)\rangle + E_k(t) \langle E_k(t) | \left(\frac{d}{dt} |E_j(t)\rangle \right) &= E_j(t) \langle E_k(t) | \left(\frac{d}{dt} |E_j(t)\rangle \right) \\ (E_j(t) - E_k(t)) \langle E_k(t) | \left(\frac{d}{dt} |E_j(t)\rangle \right) &= \langle E_k(t) | \left(\frac{d}{dt} H(t) \right) |E_j(t)\rangle \\ \langle E_k(t) | \left(\frac{d}{dt} |E_j(t)\rangle \right) &= \frac{\langle E_k(t) | \left(\frac{d}{dt} H(t) \right) |E_j(t)\rangle}{\Delta_{jk}(t)} \end{aligned}$$

where $\Delta_{jk}(t) = (E_j(t) - E_k(t))$. So, Eq. (2.33) becomes

$$\frac{d\beta_k(t)}{dt} = -\beta_j(t)\langle E_j(t) | \frac{d|E_j(t)\rangle}{dt} - \sum_j \left[\beta_j(t) e^{i[\theta_j(t) - \theta_k(t)]} \frac{\langle E_k(t) | \left(\frac{d}{dt} H(t) \right) | E_j(t) \rangle}{\Delta_{jk}(t)} \right] \quad (2.34)$$

Now, using the definition of differentiation (and recognizing that $\langle E_k(t) | H(t) | E_j(t) \rangle = 0$ by definition)

$$\begin{aligned} \langle E_k(t) | \left(\frac{d}{dt} H(t) \right) | E_j(t) \rangle &= \lim_{\epsilon \rightarrow 0} \left[\frac{\langle E_k(t) | H(t + \epsilon) | E_j(t) \rangle + \langle E_k(t) | H(t) | E_j(t) \rangle}{\epsilon} \right] \\ &= \lim_{\epsilon \rightarrow 0} \left[\frac{\langle E_k(t) | H(t + \epsilon) | E_j(t) \rangle}{\epsilon} \right]. \end{aligned} \quad (2.35)$$

It is possible to approximate the term $\langle E_k(t) | H(t + \epsilon) | E_j(t) \rangle$ using time-dependent perturbation theory in the eigenbasis of $H(t)$. Let $c_{k,j}(t + \epsilon)$ be the coefficient multiplying the state $|E_j(t)\rangle$ when $|E_k(t + \epsilon)\rangle$ expanded in the eigenbasis of $H(t)$. Then, writing it out order by order gives $c_{k,j}(t + \epsilon) = c_{k,j}^{(0)} + c_{k,j}^{(1)}(t + \epsilon) + c_{k,j}^{(2)}(t + \epsilon) + \dots$, where $c_{k,j}^{(0)} = 0$ by definition. Writing $H_{jl}(t')$ to be the matrix element $\langle E_j(t) | H(t') | E_l(t) \rangle$, it can be seen that to first order in ϵ

$$\begin{aligned} c_{k,j}(t + \epsilon) &= c_{k,j}^{(1)}(t + \epsilon) + O(\epsilon^2) \\ &= -i \sum_l \int_t^{t+\epsilon} dt' H_{jl}(t') c_{k,l}^{(0)} e^{i\Delta_{jl}t'} + O(\epsilon^2) \\ &= -i \int_t^{t+\epsilon} dt' H_{jk}(t') e^{i\Delta_{jk}t'} + O(\epsilon^2) \\ &= -i \left[H_{jk}(t') \frac{e^{i\Delta_{jk}t'}}{i\Delta_{jk}} \right]_t^{t+\epsilon} + i \int_t^{t+\epsilon} dt' \left(\frac{dH_{jk}(t')}{dt'} \right) \frac{e^{i\Delta_{jk}t'}}{i\Delta_{jk}} + O(\epsilon^2) \\ &= -\frac{e^{i\Delta_{jk}t}}{\Delta_{jk}} \left[\left(H_{jk}(t) + \frac{dH_{jk}}{dt'} \Big|_t (t' - t) \right) (1 + i\Delta_{jk}(t' - t)) + O((t' - t)^2) \right]_t^{t+\epsilon} + \\ &\quad \frac{1}{\Delta_{jk}} \int_t^{t+\epsilon} dt' \left(\frac{dH_{jk}(t')}{dt'} \right) e^{i\Delta_{jk}t'} + O(\epsilon^2). \end{aligned}$$

By definition, $H_{jk}(t) = 0$, so this (along with $\left. \frac{dH_{jk}}{dt'} \right|_t = \frac{dH_{jk}}{dt}$) gives

$$c_{k,j}(t + \epsilon) = -\frac{e^{i\Delta_{jk}t}}{\Delta_{jk}} \left[\left(\frac{dH_{jk}}{dt}(t' - t) \right) (1 + i\Delta_{jk}(t' - t)) + O((t' - t)^2) \right]_t^{t+\epsilon} + \quad (2.36)$$

$$\frac{1}{\Delta_{jk}} \int_t^{t+\epsilon} dt' \left(\frac{dH_{jk}(t')}{dt'} \right) e^{i\Delta_{jk}t'} + O(\epsilon^2). \quad (2.37)$$

Consider the derivative $\frac{dH_{jk}(t')}{dt'}$. Expanding in powers of $1/T$ gives

$$\begin{aligned} \frac{dH_{jk}(t')}{dt'} &= \left. \frac{dH_{jk}(t')}{dt'} \right|_{1/T \rightarrow 0} + O(1/T) \\ &= \frac{dH_{jk}(t)}{dt} + O(1/T) \end{aligned}$$

where, in the second step, the fact that the derivative at time t' approaches that at time t as the total duration diverges $T \rightarrow \infty$ has been used. So, for sufficiently large T , the derivative $\frac{dH_{jk}(t')}{dt'}$ in the integrand of Eq. (2.37) can be treated as constant, taking its value at the beginning of the integration range. This gives

$$\begin{aligned} c_{k,j}(t + \epsilon) &\approx -\frac{e^{i\Delta_{jk}t}}{\Delta_{jk}} \left[\left(\frac{dH_{jk}}{dt}(t' - t) \right) (1 + i\Delta_{jk}(t' - t)) + O((t' - t)^2) \right]_t^{t+\epsilon} + \\ &\quad \frac{1}{\Delta_{jk}} \frac{dH_{jk}}{dt} \int_t^{t+\epsilon} dt' e^{i\Delta_{jk}t'} + O(\epsilon^2) \\ &\approx -\frac{e^{i\Delta_{jk}t}}{\Delta_{jk}} \left[\left(\frac{dH_{jk}}{dt}(t' - t) \right) (1 + i\Delta_{jk}(t' - t)) + O((t' - t)^2) \right]_t^{t+\epsilon} + \\ &\quad \frac{1}{\Delta_{jk}} \frac{dH_{jk}}{dt} \left[\frac{e^{i\Delta_{jk}t'}}{i\Delta_{jk}} \right]_t^{t+\epsilon} + O(\epsilon^2) \\ &\approx -\frac{e^{i\Delta_{jk}t}}{\Delta_{jk}} \left[\left(\frac{dH_{jk}}{dt}\epsilon \right) (1 + i\Delta_{jk}\epsilon) + O(\epsilon^2) \right] - \\ &\quad \frac{ie^{i\Delta_{jk}t}}{(\Delta_{jk})^2} \frac{dH_{jk}}{dt} \left[e^{i\Delta_{jk}(t'-t)} \right]_t^{t+\epsilon} + O(\epsilon^2) \\ &\approx -\frac{e^{i\Delta_{jk}t}}{\Delta_{jk}} \left[\left(\frac{dH_{jk}}{dt}\epsilon \right) (1 + i\Delta_{jk}\epsilon) + O(\epsilon^2) \right] - \\ &\quad \frac{ie^{i\Delta_{jk}t}}{(\Delta_{jk})^2} \frac{dH_{jk}}{dt} [1 + i\Delta_{jk}(t' - t) + O((t' - t)^2)]_t^{t+\epsilon} + O(\epsilon^2) \\ &\approx -\frac{e^{i\Delta_{jk}t}}{\Delta_{jk}} \left(\frac{dH_{jk}}{dt}\epsilon + O(\epsilon^2) \right) - \frac{ie^{i\Delta_{jk}t}}{(\Delta_{jk})^2} \frac{dH_{jk}}{dt} (i\Delta_{jk}\epsilon) + O(\epsilon^2) \end{aligned}$$

$$\begin{aligned}
&\approx -\frac{e^{i\Delta_{jk}t}}{\Delta_{jk}} \frac{dH_{jk}}{dt} \epsilon + \frac{e^{i\Delta_{jk}t}}{\Delta_{jk}} \frac{dH_{jk}}{dt} \epsilon + O(\epsilon^2) \\
&\approx O(\epsilon^2).
\end{aligned}$$

This gives

$$\langle E_k(t) | H(t + \epsilon) | E_j(t) \rangle \approx O(\epsilon^2)$$

and so

$$\begin{aligned}
\langle E_k(t) | \left(\frac{d}{dt} H(t) \right) | E_j(t) \rangle &\approx N \lim_{\epsilon \rightarrow 0} \left[\frac{O(\epsilon^2)}{\epsilon} \right] \\
&\approx 0
\end{aligned}$$

meaning that Eq. (2.34) becomes approximately

$$\frac{d\beta_k(t)}{dt} \approx -\beta_j(t) \langle E_j(t) | \frac{d|E_j(t)\rangle}{dt} \rangle \quad (2.38)$$

Now, it can be shown that the term $\langle E_j(t) | \frac{d|E_j(t)\rangle}{dt} \rangle$ has no real part. To do this, start with the normalisation condition

$$\begin{aligned}
1 &= \langle E_j(t) | E_j(t) \rangle \\
0 &= \frac{d}{dt} (\langle E_j(t) | E_j(t) \rangle) \\
&= \frac{d}{dt} (\langle E_j(t) | | E_j(t) \rangle + \langle E_j(t) | \frac{d}{dt} (| E_j(t) \rangle)) \\
&= 2\text{Re} \left[\langle E_j(t) | \frac{d}{dt} (| E_j(t) \rangle) \right].
\end{aligned}$$

This means that Eq. (2.38) has the solution

$$\beta_k(t) \approx \beta_k(0) e^{i\gamma_k t}$$

where the real valued quantity $\gamma_k = -i \int_0^t dt' \left(\langle E_j(t') | \frac{d|E_j(t')\rangle}{dt'} \right)$ represents a phase that can usually be made to vanish by an appropriate choice of phases on the Hamiltonian eigenstates. Finally,

$$\alpha_k(t) \approx \alpha_k(0) e^{i\theta_j(t)} e^{i\gamma_k t}$$

where $\theta_k(t)$ is known as the dynamical phase.

For the purposes of AQC, the initial state is usually prepared as the ground-state of the initial Hamiltonian. That is, $\alpha_k(0) = \delta_{k,1}$. This means that, at the end of the adiabatic sweep, $\alpha_1(T) \approx e^{i\theta_j(T)}e^{i\gamma_k T}$ and the success probability will be

$$P(T) \approx 1.$$

2.5.2 Solving the search problem using AQC

It can be shown (Roland and Cerf, 2002; Morley et al., 2019) that AQC can be used to solve the search problem of size N , defined in section 2.3, in a time that scales as $O(\sqrt{N})$, recovering the quadratic speedup seen in Grover's algorithm (Grover, 1996) and in the QW algorithm described in subsection 2.4.5.

Using a transverse field Hamiltonian as the AQC driver

$$\hat{H}_{\text{drive}} = \frac{1}{2} \left[n\mathbb{1} - \sum_{j=0}^{n-1} \hat{X}_j \right], \quad (2.39)$$

which is similar to the hypercube Hamiltonian defined for QW in Eq. (2.23), it can be shown that an optimal schedule

$$s^{(c)}(t) = \frac{2\sqrt{R_2}}{\sqrt{N}(1+R_1)^2} \tan \left\{ \frac{8\epsilon\sqrt{R_2}R_1^2 t}{n\sqrt{N}R_2^2} - c \right\} + \frac{1}{1+R_1} \quad (2.40)$$

where

$$R_1 = \frac{1}{N} \sum_{r=1}^n \binom{n}{r} \frac{1}{r}, \quad (2.41)$$

$$R_2 = \frac{1}{N} \sum_{r=1}^n \binom{n}{r} \frac{1}{r^2}, \quad (2.42)$$

$$c = \arctan \left\{ \frac{(1+R_1)\sqrt{N}}{2\sqrt{R_2}}, \right\} \quad (2.43)$$

run from $t = 0$ to

$$t_f = \frac{n\sqrt{N}R_2^2}{8\epsilon\sqrt{R_2R_1^2}} \left[\arctan \left\{ \frac{\sqrt{N}R_1(1+R_1)}{2\sqrt{R_2}} + c \right\} \right] \quad (2.44)$$

$$\simeq \frac{\pi\sqrt{N}}{4\epsilon} \text{ for large } N \quad (2.45)$$

will result in a success probability

$$P(t_f) \geq 1 - \epsilon^2 \quad (2.46)$$

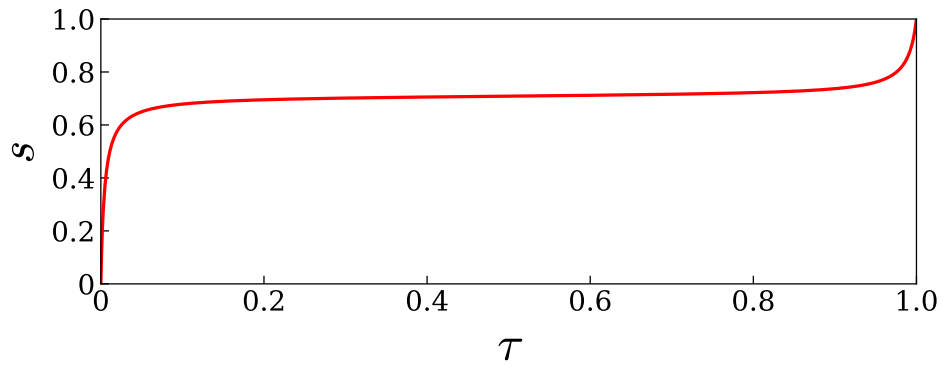
To illustrate these results, the optimal schedule $s^{(c)}(t)$ defined in Eq. (2.40) is plotted against dimensionless time t/t_f in Fig. 2.6 for $n = 11$ (Fig. 2.6(a)) and for $n = 30$ (Fig. 2.6(b)). In both cases, it can be seen that a majority of the schedule consists of very slow change around some critical value of s , with rapid change only near $s = 0$ and $s = 1$. This is more pronounced for larger values of n .

The probability $P_{\text{AQC}} = |\langle E_1^{(P)} | \psi(t_f) \rangle|^2$ that a measurement in the computational basis at (dimensionless) time τ , following the optimized schedules $s^{(c)}(t)$ with $\epsilon = 0.01$, would result in the problem ground state $|E_1^{(P)}\rangle$ is shown in Fig. 2.7 for $n = 11$ (Fig. 2.7(a)) and for $n = 30$ (Fig. 2.7(b)). In both cases, it can be seen that the probability P_{AQC} slowly increases from its initial value $P_{\text{AQC}} = 1/N$ to its final value $P_{\text{AQC}} \approx 1$.

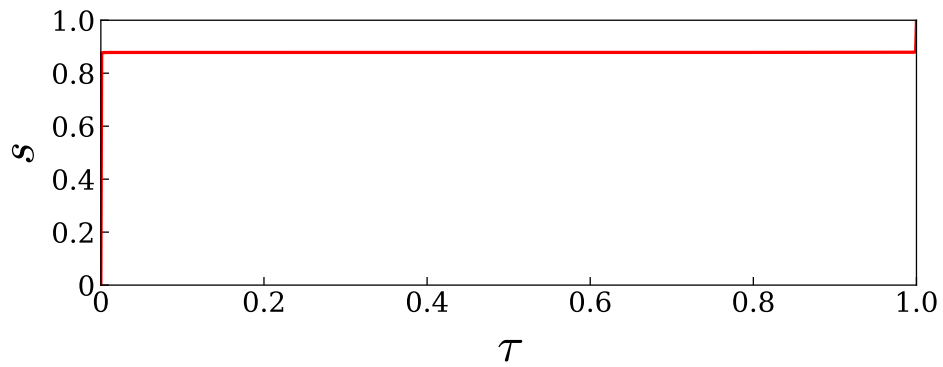
2.6 CTQC: Quantum Annealing

QA (Finnila et al., 1994; Kadowaki and Nishimori, 1998) is a time-dependent CTQC method that operates similarly to AQC, and is often parametrised the same way according to Eq. (2.25). However, unlike AQC, the schedule $s(t)$ usually proceeds much faster than the adiabatic limit, and as such does not benefit from the formal guarantee of achieving a success probability close to unity, and thus often requires multiple repeats to achieve a high success probability.

QA can be thought of as a heuristic quantum analogue of classical SA (see subsection 2.1.2). In SA, a Monte Carlo-type method with a gradually-decreasing temperature parameter is used to ‘relax’ a simulated classical system into its minimum-energy configuration amidst a potentially

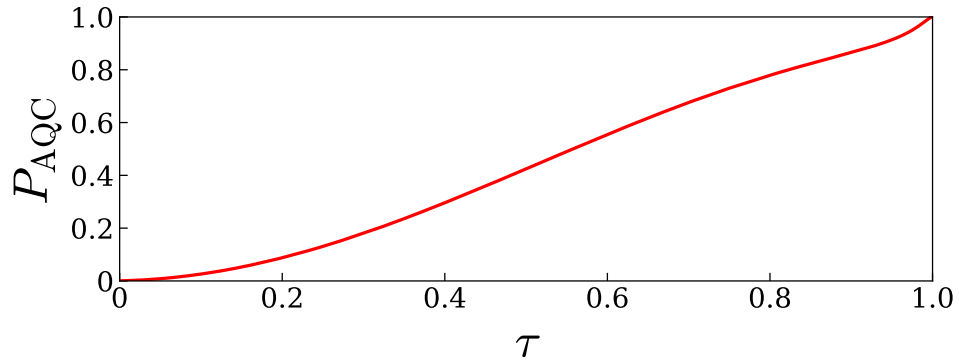


(a)

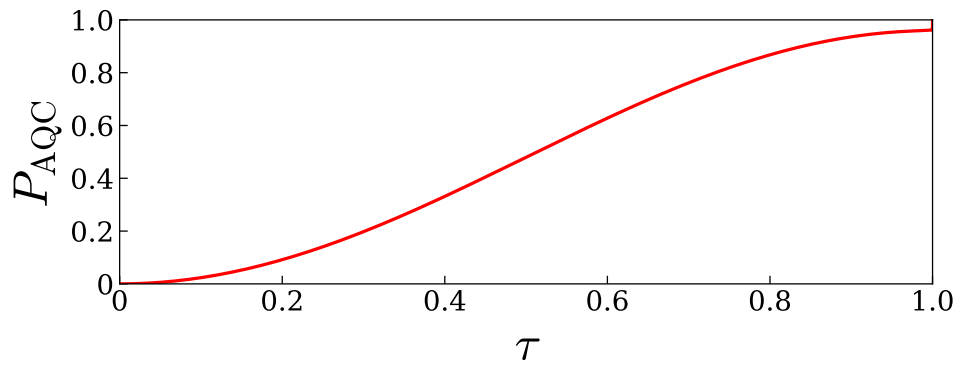


(b)

Figure 2.6: Analytically calculated AQC schedule $s^{(c)}(t)$ against fractional time $\tau = t/t_f$ (where t_f is the total duration) for (a) $n = 11$ and (b) $n = 30$.



(a)



(b)

Figure 2.7: The probability $P_{\text{AQC}}(t)$ that a computational-basis measurement on the instantaneous state at time t along the analytically calculated AQC schedule $s^{(c)}(t)$ would result in the solution state, plotted against fractional time $\tau = t/t_f$ (where t_f is the total duration) for (a) $n = 11$ and (b) $n = 30$.

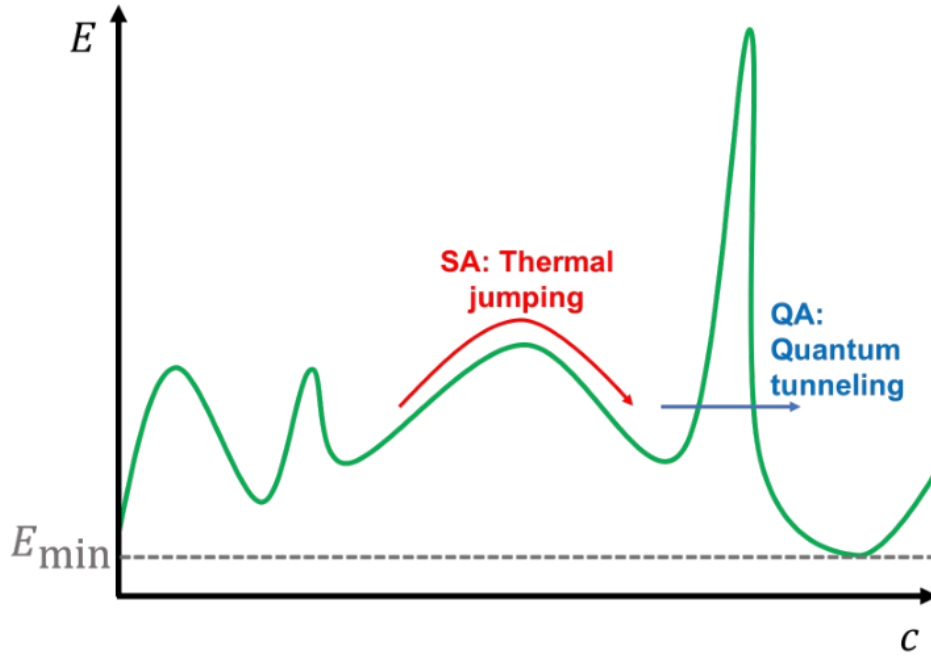


Figure 2.8: An illustration of the differing mechanisms by which SA and QA explore an energy landscape E (plotted here against a 2D configuration space c). Thermal fluctuations in SA can jump over wide but short energy barriers, whereas QA can tunnel through thin but tall energy barriers.

complicated energy landscape defined by a problem cost function. To do this, the SA algorithm jumps around the configuration space, with transition probabilities governed by a thermal distribution as a function of the temperature T , which is decreased over time, leading the system into a minimum which is hopefully the global one. However, if the landscape has many tall barriers, the SA algorithm will get stuck in a local minimum between two such barriers, as T will become too low to generate large enough fluctuations.

In QA, the simulated thermal fluctuations are replaced by the quantum fluctuations, on actual Ising model hardware, induced by the transverse field driver Hamiltonian $\hat{H}_{\text{drive}}$. This mechanism allows the system to tunnel through the kind of ‘thin but tall’ energy barriers that would cause a problem for SA, although it would be exponentially suppressed in the case of wide barriers. The two different mechanisms are illustrated (schematically in 2D) in Fig. 2.8.

Applications of QA have been explored in many diverse fields including traditional computer science (Chancellor et al., 2016a; Choi, 2010, 2011b), decoding communications (Chancellor et al., 2016b), finance (Marzec, 2016; Orús et al., 2019; Venturelli and Kondratyev, 2019), error correction of quantum memories (Roffe et al., 2019), scheduling (Venturelli et al., 2000; Crispin

and Syrichas, 2013; Tran et al., 2016), computational biology (Perdomo-Ortiz et al., 2012), flight gate assignment (Stollenwerk et al., 2019), air traffic management (Stollenwerk et al., 2020), and hydrology (O’Malley, 2018). This has partially been because of the success of experimental quantum annealing, a notable example being the superconducting circuit quantum annealing devices produced by D-Wave Systems Inc.

While highly successful, the D-Wave superconducting circuit devices currently operate in a highly dissipative, but still strongly quantum mechanical, regime. This dissipation can actually play a positive role (Dickson et al., 2013) and is the driving mechanism behind the new ‘reverse annealing’ feature which these devices implement (Chancellor, 2017; D-Wave Systems Inc., 2019b). Furthermore, there is increasing interest in quenches (Amin et al., 2018a) or pauses (Marshall et al., 2019; Passarelli et al., 2019) in quantum annealing, which effectively run an open-system version of a quantum walk during part of the computation. Thermal relaxation effects dominate in the regime currently accessible by flux qubit quantum annealers, which is the focus of these works.

While QW and AQC have analytical underpinnings, QA has largely been motivated by heuristic arguments (and experimental successes) of the kind described in this subsection. Chapter 5 will focus on QA, including introducing an analytical approach to determining if a QA protocol will likely be successful, and to calculating heuristic parameters for such protocols.

2.7 Numerical Methods

Much of the work presented in this thesis has been done via numerical simulation and optimization, due to the lack of analytical tractability. All of the numerical work was performed using the Python language (Van Rossum and Drake, 2003), aided extensively by the IPython (Pérez and Granger, 2007) interpreter and the Jupyter Notebook (Kluyver et al., 2016) system. Most of the computational heavy-lifting was performed with the NumPy (Oliphant, 2006) and SciPy (Jones et al., 2001–) libraries for performing numerical and scientific calculations at a lower level than available in pure Python. Most of the data-processing was done via dataframes provided by the pandas library (McKinney et al., 2010). All plots were made using the Matplotlib (Hunter, 2007) library. Where additional computing power beyond what is available on a personal computer was required, calculations were run on the Imperial and Durham University

high performance computing facilities.

2.7.1 Simulation of quantum dynamics

In almost all cases where full simulation of unitary quantum dynamics have been performed, this was done via the quimb (Gray, 2018) library as a convenient interface to the `scipy.integrate.complex_ode` adaptive time-step integrator using an 8th order Runge-Kutta method. The exceptions to this are the mixing-time calculations in subsection 3.2.5, for which the `scipy.sparse.linalg.expm_multiply` method (based on the methods in Al-Mohy and Higham (2011)) was used to calculate the action of the unitary propagator on the quantum state, and the numerical results for quantum annealing on the shortest vector problem in subsection 6.3.1, for which a slow but stable split-step Fourier method was used to overcome numerical instabilities in the other methods.

The calculation of the infinite time success probability P_∞ , defined and explained in section 3.2.1, requires full diagonalisation of a Hamiltonian. Where P_∞ has been calculated, the diagonalisation was performed using `scipy.linalg.eigh` method for Hermitian matrices, which in turn uses LAPACK routines in FORTRAN.

2.7.2 Problem instance generation

Chapters 3 and 5 both use the dataset at Chancellor et al. (2019) containing 10,000 Sherrington-Kirkpatrick spin-glass problem instances for each number of spins $5 \leq n \leq 20$ (defined in 2.1.3). For each instance, the coupling strengths J_{jk} and field strengths h_j were generated from a normal distribution with mean 0 and standard deviation 1 using `numpy.random.normal`. Likewise, the random energy model instances (also 10,000 at each size) defined and used in Chapter 3, also contained in Chancellor et al. (2019), were created by generating the $N = 2^n$ problem energies directly from a normal distribution with mean 0 and standard deviation 1 using `numpy.random.normal`. The generation of MAX2SAT problem instances used in Chapter 4 is described in the opening section of that chapter; the function `numpy.random.randint` was used to generate random integers as necessary for generating the clauses. Where ‘scrambled’ versions of problems have been used (i.e for subsection 3.3.1 and Fig. 4.6), the scrambling was

performed via `numpy.random.permutation`.

2.7.3 Fitting and optimization

Throughout the thesis, whenever a linear model has been fit to some data, this has been done using a least-squares method via `scipy.optimize.curve_fit`. For Fig. [5.9](#), the numerical optimization required for finding the lower bound was performed using `scipy.optimize.minimize`.

Chapter 3

Using quantum walks to solve optimization problems

3.1 Introduction

In this chapter, I tackle the question of if, and how, a continuous-time quantum walk (QW) can be useful for practical quantum optimization in the Ising model, using the Sherrington-Kirkpatrick (SK) spin-glass ground-state problem defined in subsection 2.1.3 as a prototypical example. This is a useful topic to explore, due to the broad importance of optimization problems (see section 2.1) and the appeal of the Ising model as a practical route to quantum algorithms (see section 2.2). Furthermore, an algorithm which is essentially a QW on a spin glass, although presented using different terminology, has been analysed by [Hastings \(2019\)](#). Along with the same energy conservation arguments I describe in section 3.3.2, Hastings' findings suggest that QWs on spin glasses will be interesting to explore.

As discussed in subsection 2.4.5, QWs can be used to solve the unstructured search problem with the same quantum speedup as achieved by Grover's algorithm ([Grover, 1996](#)) in the gate model. Given that QWs provide a better performance for searching than AQC, especially when limited coherence time and other practical factors, such as precision of control settings, are considered ([Morley et al., 2019](#)), it is important to understand how they perform for a wider range of problems. The SK problem considered in this chapter has features in common with real life hard optimization problems which, unlike the search problem, do not typically admit

analytic solutions. The search problem as solved by QWs provides useful comparisons with these spin glass problems.

Finding the ground state of a SK spin glass is known to be not only NP-hard, but also *uniformly-hard*, as suggested by its finite-temperature spin glass transition. Without a finite temperature spin glass transition, a problem cannot be uniformly hard, since the lack of a transition implies that typical cases will be easy for the Monte Carlo family of algorithms, as discussed in (Katzgraber et al., 2014). As has been shown for a random problem type used in early benchmarks of quantum annealing hardware (Katzgraber et al., 2014), uniform hardness is crucial: without this property, *randomly* generated instances of NP-hard problems are not necessarily hard to solve (Beier and Vöcking, 2004; Krivelevich and Vilenchik, 2006; Lucas, 2014).

The SK problem Hamiltonian differs from the unstructured search problem by having structure, produced by the $\hat{Z}_j \hat{Z}_k$ terms. As a result, the covariances between the energies of two basis states depends on the Hamming-distance between them (Baldwin and Laumann, 2018). Knowing the energy of one state gives some information about the energy of states that differ by a small number of bit-flips. This results in a distribution of the eigenenergies that is almost normal (as can be seen by plotting the distributions and numerically calculating moments), but which deviates from normal in the tails of the distribution.

To distinguish the effects of the correlations between energy difference and Hamming distance in the spin glass from the effects of simply having a more random eigenenergy distribution, a random energy model (REM) (Derrida, 1980) is used for comparison. A problem with perfect correlations is easy to solve, like finding the ground state of a spin system with only local fields, no couplings. A completely random problem, such as finding the ground state of a random energy model instance, has no correlation to exploit and so is very hard to solve, essentially requiring random guessing. However, a completely random model is fully characterised by average values of its properties, and finding exact ground states of specific instances is typically not interesting. Intermediate problems with some correlations are both hard and interesting, with complex behaviour and phase diagrams, like spin models with frustration and spin glass phases. Real optimization problems typically have correlations; they are often hard to solve but also produce interesting solutions. The inherent complexity of a problem comes from the structure of the problem and its correlations, not the structure of the solution itself. One

illustration of this is the construction of hard benchmarking problems with ‘planted’ solutions defined at the time of construction, which therefore have no special structure related to the problem’s hardness, see for example (Hen, 2019; Hamze et al., 2019).

This chapter is structured as follows: In section 3.2, I present the main results showing how QW can find spin glass ground states more effectively than a quantum unstructured search algorithm. In section 3.3, I identify the computational mechanisms and important aspects of the problem structure that contribute to the effectiveness of computation by QW. Finally, in section 3.4, I will summarize the chapter.

3.2 Quantum walks with spin glasses

In order to implement a QW algorithm for finding the ground states of the spin glasses defined in subsection 2.1.3, I have followed the procedure described in subsection 2.4.3: Choose a quantum walk graph G and associated Hamiltonian $\hat{H}_G$ (and premultiply it by a hopping rate γ) and add the spin glass Hamiltonian to get the full QW Hamiltonian $\hat{H} = \gamma\hat{H}_G + \hat{H}_{\text{SK}}$. Since the hypercube is the natural choice of graph for qubit implementations, I have used this graph, with walk Hamiltonian $\hat{H}_h$ defined in Eq. (2.23), unless otherwise indicated. For the initial state $|\psi(0)\rangle$, I have used the equal superposition (Eq. (2.20)), which is the ground state of the hypercube Hamiltonian $\hat{H}_h$.

3.2.1 Defining success probability

In general, it will be best to use different measurement times t_f for each repeat. Different measurement times will produce different success probabilities $P(t_f)$, and varying the measurement time avoids repeatedly measuring at a time for which the probability $P(t_f)$ happens to be atypically small. More precisely, an *average single run success probability* $\bar{P}(t, \Delta t)$ can be defined by choosing the measurement time t_f uniformly at random in an interval $[t, t + \Delta t]$; that is,

$$\bar{P}(t, \Delta t) \equiv \frac{1}{\Delta t} \int_t^{t+\Delta t} dt_f P(t_f). \quad (3.1)$$

Operationally, choosing the measurement time t_f randomly in the interval $[t, t + \Delta t]$ samples success probabilities from the distribution with $\bar{P}(t, \Delta t)$ as its mean. Sampling measurement times in this way means that the protocol typically needs to be repeated $M_{\text{rep}} \sim 1/\bar{P}(t, \Delta t)$ times to achieve an overall $O(1)$ success probability. Note that it is not generally possible to check whether the state measured is indeed the ground state of $\hat{H}_P$. However, it is easy to calculate the energy of the state measured in each repeat. If only the lowest energy state is accepted, it is only necessary for the ground state of $\hat{H}_P$ to be measured once out of all the repeats. The more repeats, the more confidence is gained that the lowest energy state found is the ground state.

In the limit of small interval width Δt , the average success probability defined in Eq. (3.1) reduces to the single time probability $P(t_f) = \lim_{\Delta t \rightarrow 0} \bar{P}(t_f, \Delta t)$. The long time limit of this average,

$$P_\infty \equiv \bar{P}(0, \infty) \equiv \lim_{\Delta t \rightarrow \infty} \bar{P}(0, \Delta t), \quad (3.2)$$

is particularly useful, because it predicts the short time average well (see subsection 3.2.5), and can be calculated via a numerical diagonalization of the Hamiltonian, as I will now show.

Writing the spectral expansion of the full QW Hamiltonian as

$$\hat{H} = \sum_{a=0}^{N-1} E_a |E_a\rangle \langle E_a|, \quad (3.3)$$

with indices ordered such that $E_a \leq E_{a+1}$ and $|E_a\rangle$ the eigenstate with eigenvalue E_a , the instantaneous probability can be written in terms of the spectral expansions as

$$\begin{aligned} P(t) &= \left| \langle E_0^{(P)} | \exp(-it\hat{H}) | \psi(0) \rangle \right|^2 \\ &= \left| \sum_{a=0}^{N-1} \exp(-itE_a) \langle E_0^{(P)} | E_a \rangle \langle E_a | \psi(0) \rangle \right|^2 \\ &= \sum_{a=0}^{N-1} \left| \langle E_0^{(P)} | E_a \rangle \right|^2 |\langle E_a | \psi(0) \rangle|^2 \\ &\quad + \sum_{a \neq b=0}^{N-1} \left[\exp(-it(E_a - E_b)) \langle E_0^{(P)} | E_a \rangle \times \right. \\ &\quad \left. \langle E_a | \psi(0) \rangle \langle E_b | E_0^{(P)} \rangle \langle \psi(0) | E_b \rangle \right]. \end{aligned} \quad (3.4)$$

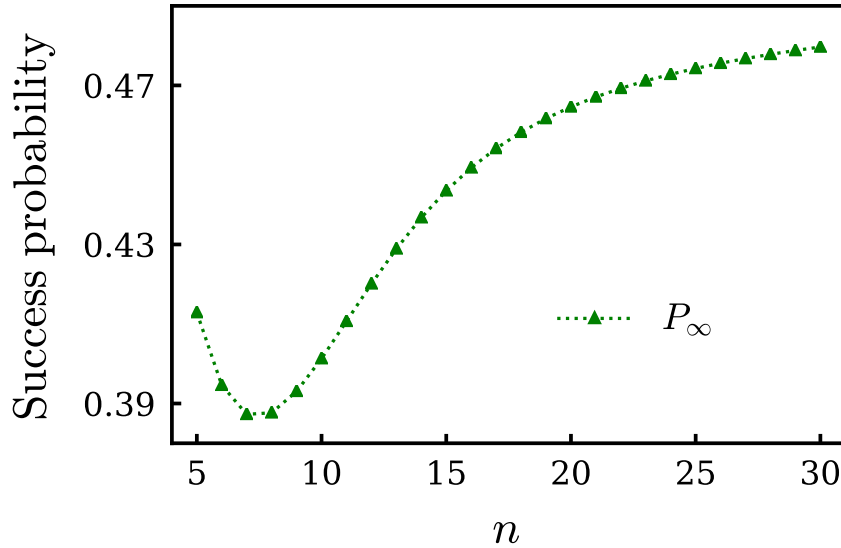


Figure 3.1: Infinite time average success probability P_∞ (green triangles, dotted line), defined in Eq. (3.2), for the unstructured search problem at different numbers of qubits n .

Assuming no degeneracy (that is, all gaps $E_a - E_b$ are nonzero), which is justified for the randomized nature of the SK and REM problems, the oscillatory terms cancel in the infinite limit (because $\int_0^\infty dt \exp(-it\theta) = 0$ for nonzero θ) to leave the infinite-time average probability P_∞ given by

$$P_\infty = \sum_{a=0}^{N-1} \left| \left\langle E_0^{(P)} \middle| E_a \right\rangle \right|^2 |\langle E_a | \psi(0) \rangle|^2. \quad (3.5)$$

To illustrate that the infinite-time average probability P_∞ is a meaningful measure of success probability, I have applied it to the unstructured search problem described in section 2.3. The result is shown in Fig. 3.1. As with instantaneous success probability $P(t_f)$ at the asymptotically optimal and numerically determined best times shown in Fig. 2.5(b), the infinite-time average probability P_∞ in Fig. 3.1 shows the pronounced dip around $n = 8$ qubits, with smooth behaviour only settling in for $n > 12$ qubits. The main difference is that the infinite-time probability P_∞ asymptotes to a half instead of 1, due to being an average over the oscillations.

Hence, QW applied to unstructured search with a random measurement time should on average only need to be repeated twice to locate the marked state; knowing the exact time to measure for the optimal success probability is not necessary for the success of the algorithm. Fixed point quantum search algorithms (Yoder et al., 2014; Dalzell et al., 2017) are another approach that avoids the need to know how long to run the algorithm for.

In this chapter, I will often use the long time average P_∞ as an indication of the success probability achievable in a single run, and thus the number of repeats required to achieve $O(1)$ success probability overall. The timescale required to reach this probability in each run will be addressed separately in section 3.2.5.

An alternative success metric that is often used is the time-to-solution (TTS) (Albash and Lidar, 2018b), defined as

$$\text{TTS}(t_f) = t_f R(t_f), \quad (3.6)$$

where t_f is a chosen duration for each repeat (including any initialisation time) and $R(t_f)$ is the required number of repeats to amplify the success probability $P(t_f)$ from a single run of duration t_f to some specified constant (e.g 0.99). Typically, $\text{TTS}(t_f)$ is optimized over t_f to find the optimal TTS. In order to simplify my analysis, and because I am exploring the QW mechanism rather than aiming to present an optimized algorithm, I have opted for the simpler approach of using the various time-averaged success probabilities $\bar{P}(t, \Delta t)$ as the metrics of success.

To see the relationship between $\bar{P}(t, \Delta t)$ and $\text{TTS}(t_f)$, consider the modified $\text{TTS}(t, \Delta t)$ that arises from picking a measurement time t_f uniformly from the interval of width of Δt , centred on $t + \Delta t/2$. The success probability of a single run is $\bar{P}(t, \Delta t)$ and so the required number of repeats is $R(t, \Delta t) = O(1/\bar{P}(t, \Delta t))$. Using the centre of the time window as the average runtime of each repeat, a sensible modified TTS is

$$\text{TTS}(t, \Delta t) = \left(t + \frac{\Delta t}{2}\right) R(t, \Delta t). \quad (3.7)$$

As I will show numerically in subsection 3.2.5, using a time window for which both t and Δt scale at worst logarithmically in problem size N results in a success probability that can't be improved upon by using longer times, while the number of repeats required scales much more harshly. As such, it is reasonable to characterise the total runtime as being $\sim \tilde{O}(1/\bar{P})$.

3.2.2 Random energy model

To isolate the effect of the correlations in the SK problem, I will compare it with the random energy model (REM) (Derrida, 1980), in which the eigenenergies themselves are independently drawn from a normal distribution. The problem Hamiltonian $\hat{H}_{\text{REM}}$ for REM is

$$\hat{H}_{\text{REM}} = \sum_{j=0}^{N-1} F_j |j\rangle \langle j|, \quad (3.8)$$

with $\{|j\rangle\}_{j=0}^{N-1}$ the computational (Z) basis and the energies F_j drawn independently from the normal distribution $\mathcal{N}(0, 1)$.

REM has a similar energy level distribution to that of SK, apart from the tails. By definition it lacks the correlations: knowing the energy of one state gives no information about the energies of other states. Comparison between these two models highlights the effect of the pairwise structure in the SK model.

3.2.3 Setting the hopping rate

In contrast to the search problem described in section 2.3, for SK and REM it is impossible to efficiently calculate the optimal hopping rate $\gamma_{\text{opt}}^{(h)}$ that maximizes the success probability. It is not even clear which measure of success probability should be maximized because, unlike the search problem, there will be no efficient way to find the optimal measurement time $t_f^{(\text{opt})}$ for any choice of hopping rate γ . To bootstrap the investigation, I have chosen to define the optimal hopping-rate $\gamma_{\text{opt}}^{(h)}$ with respect to one of the average probabilities defined in Eq. (3.1); in particular, I choose the hopping rate that maximizes the infinite-time average probability P_∞ defined in Eq. (3.2). I make this choice because the infinite-time average probability P_∞ is numerically convenient to calculate, and because it has been seen to be a relevant measure of probability in the search example, see Fig. 3.1. I will show in subsection 3.2.5 that the probability P_∞ typically agrees well with probabilities averaged over shorter and more practical time windows.

Some plots of the infinite-time probability P_∞ against hopping rate γ for typical 11-qubit examples of the SK and REM are shown in Fig. 3.2. Note that the maximal success probability

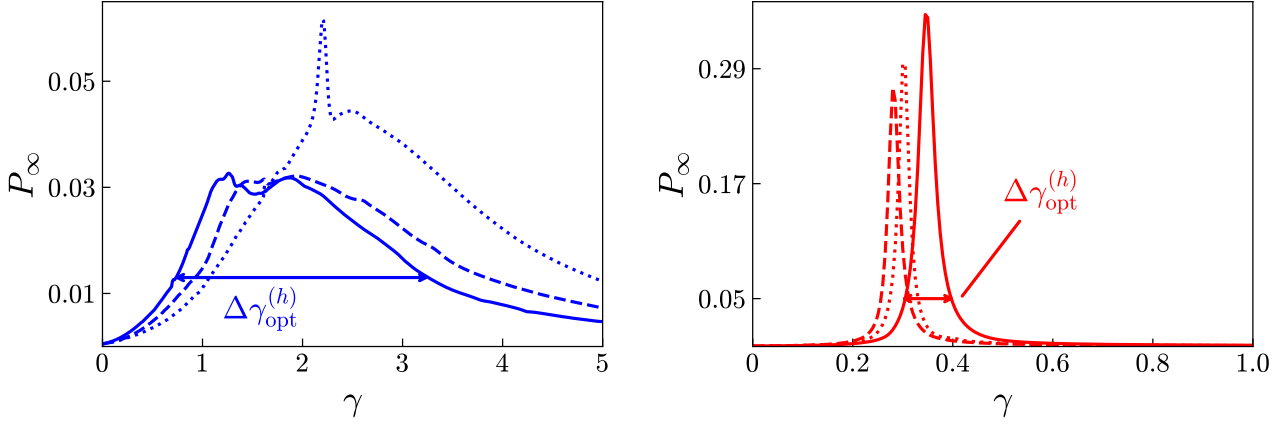


Figure 3.2: Infinite-time success probability P_∞ against hopping rate γ for 3 typical 11-qubit examples of SK (left) and REM (right). Also indicated (for one example in each plot) is the width $\Delta\gamma_{\text{opt}}^{(h)}$ of the peak.

varies by an order of magnitude between the two problem-types, with REM higher and SK lower. While the optimal hopping rate $\gamma_{\text{opt}}^{(h)}$ is instance-dependent, these plots show that the dependence of infinite-time probability P_∞ on hopping rate γ is typically characterised by broad, bumpy peaks for SK, and by narrow, well-defined peaks for REM. This implies that a precise value of the hopping rate γ is needed for REM, while there is some tolerance to non-optimal values of the hopping rate γ for SK for the sizes studied here.

To investigate the success probability more systematically, I performed a brute-force numerical search to find the optimal hopping rate $\gamma_{\text{opt}}^{(h)}$ that maximizes the success probability P_∞ for each SK and REM instance from the data sets of 10k random instances for $5 \leq n \leq 11$. This gives a baseline maximum average single run success probability for the QW algorithm.

The optimal hopping rates $\gamma_{\text{opt}}^{(h)}$ correspond to the best a QW algorithm on the hypercube can possibly do in a single run. For practical algorithms, it is necessary to find a heuristic method for choosing the hopping rate that can be calculated from the known parameters. For the QW algorithm for unstructured search, the optimal hopping rate balances the energy between the two components of the Hamiltonian, $\hat{H}_P$ and $\gamma_{\text{opt}}^{(h)}\hat{H}_h$. Guided by this, I define the heuristic hopping rate $\gamma_{\text{heur}}^{(h)}$ for SK and REM such that it balances these overall energy-scales on average. To do this, I simply matched the energy-spread $E_{N-1}^{(h)} - E_0^{(h)}$ of the hypercube walk Hamiltonian $\hat{H}_h$ with the *average* energy-spread $\langle E_{N-1}^{(P)} - E_0^{(P)} \rangle$ of the problem Hamiltonian $\hat{H}_P$. For the hypercube Hamiltonian $\hat{H}_h$ defined in Eq. (2.23), the energy spread $E_{N-1}^{(h)} - E_0^{(h)}$ is equal to

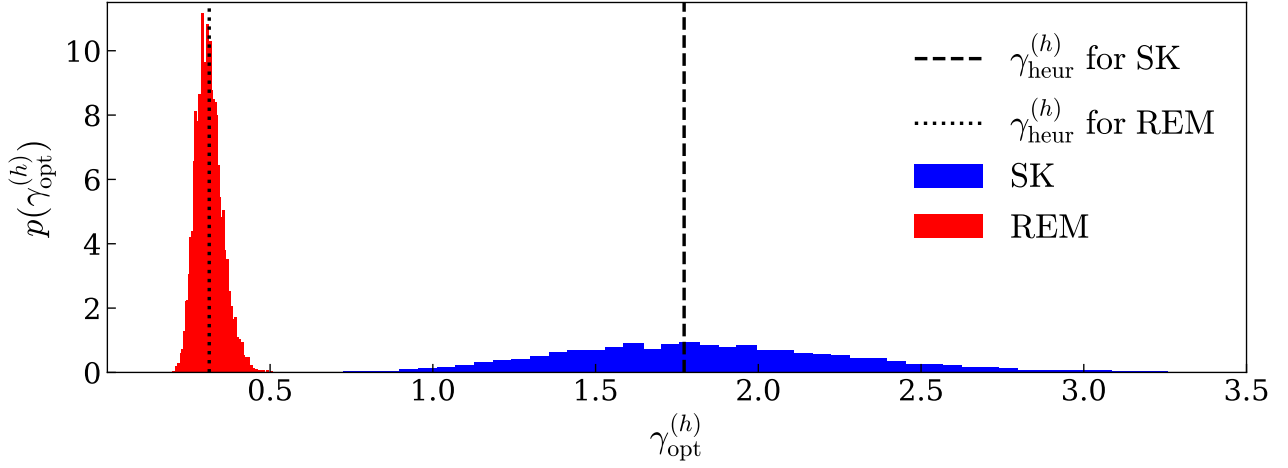


Figure 3.3: Histograms (showing relative frequency $p(\gamma_{\text{opt}}^{(h)})$) of the numerically-found optimal hopping rates $\gamma_{\text{opt}}^{(h)}$ for the 10,000 11-qubit instances of SK (blue) and REM (red). The dashed and dotted lines show the heuristic hopping rate $\gamma_{\text{heur}}^{(h)}$, calculated according to Eq. (3.9), for SK and REM respectively.

$2n\gamma$; hence, I define the heuristic hopping rate $\gamma_{\text{heur}}^{(h)}$ by

$$\gamma_{\text{heur}}^{(h)} \equiv \frac{1}{2n} \langle E_{N-1}^{(P)} - E_0^{(P)} \rangle. \quad (3.9)$$

To demonstrate that this heuristic is sensible, Fig. 3.3 shows a comparison of the distributions of optimal hopping rates $\gamma_{\text{opt}}^{(h)}$ for SK (blue) and REM (red), as well as the heuristic hopping rates (black, dashed and dotted lines for SK and REM respectively) calculated according to Eq. (3.9), for the 11-qubit data set. For both SK and REM, the heuristic hopping rate $\gamma_{\text{heur}}^{(h)}$ falls in the centre of the $\gamma_{\text{opt}}^{(h)}$ distributions. Note that the SK distribution is much broader than for REM: not only are the individual peaks for $\gamma_{\text{opt}}^{(h)}$ for SK much broader than for REM (Fig. 3.2), but the distribution of the maxima of those peaks is also much broader (Fig. 3.3). This may seem to be a problem for specifying a heuristic value for γ for SK from average energies, but as I will show, it is actually REM that fails for the heuristic γ , while SK works well.

For a normal distribution of energy levels, the average problem energy spread can be estimated as

$$\langle E_{N-1}^{(P)} - E_0^{(P)} \rangle \simeq -(2^{\frac{3}{2}} \sigma_P^{(\text{energy})}) \text{erf}^{-1}\left(\frac{1}{N} - 1\right), \quad (3.10)$$

where $\sigma_P^{(\text{energy})}$ is the standard deviation of the energy eigenvalues of the problem Hamiltonian. For REM, the standard deviation $\sigma_{\text{REM}}^{(\text{energy})}$ is equal to 1 by definition (see subsection 3.2.2). For SK, the standard deviation $\sigma_{\text{SK}}^{(\text{energy})}$ can be shown to be equal to $\frac{1}{2}[n(n+3)]^{\frac{1}{2}}$. Eq. (3.10)

is accurate for REM (which has normally-distributed energy levels by definition) but, as already noted, the distribution of the eigenenergies in SK deviates from normal, especially in the tails. Through numerical fitting, I have found that there is a multiplicative constant factor of approximately 0.887 that corrects the formula in Eq. (3.10) for SK for the effects of the non-normal tails. For the numerical analysis using $\gamma_{\text{heur}}^{(h)}$ in this chapter, I have used the numerically calculated average energy-spread at each number of qubits n .

Fig. 3.4(a) compares the heuristic hopping rate $\gamma_{\text{heur}}^{(h)}$ and average optimal hopping rate $\langle\gamma_{\text{opt}}^{(h)}\rangle$ at different numbers of qubits $5 \leq n \leq 11$. The full width at half maximum (FWHM) has also been calculated for each instance, to estimate the tolerance $\Delta\gamma_{\text{opt}}^{(h)}$ to deviations from the optimal hopping rate $\gamma_{\text{opt}}^{(h)}$ (illustrated in Fig. 3.2). The width of the shaded regions in Fig. 3.4(a) corresponds to the average tolerance range $\langle\Delta\gamma_{\text{opt}}^{(h)}\rangle$ at each number n of qubits. While the heuristic hopping rate differs slightly from the average optimal hopping rate for SK, the average tolerance range $\langle\Delta\gamma_{\text{opt}}^{(h)}\rangle$ is much broader, and does not shrink with increasing number of qubits n . For REM, however, while there is close agreement on average, the tolerance range shrinks quickly with the number of qubits n as the peaks (as in Fig. 3.2, right) become narrower. This means that the heuristic hopping rate $\gamma_{\text{heur}}^{(h)}$ is more likely to lie further than $2\Delta\gamma_{\text{opt}}^{(h)}$ outside of the actual probability peak for each instance, even though it agrees well with the average optimal hopping rate $\langle\gamma_{\text{opt}}^{(h)}\rangle$. Consequently, a QW with the heuristic hopping rate $\gamma_{\text{heur}}^{(h)}$ does not perform well for most REM instances.

It is instructive to quantify this sensitivity to deviations from the optimal hopping rate $\gamma_{\text{opt}}^{(h)}$. Fig. 3.4(b) shows log-linear and log-log plots of the average *fractional* tolerance range $\langle\Delta\gamma_{\text{opt}}^{(h)}/\gamma_{\text{opt}}^{(h)}\rangle$ against number n of qubits for SK (blue circles), REM (red squares) and search (green triangles) on the hypercube. For SK, the fractional tolerance range $\langle\Delta\gamma_{\text{opt}}^{(h)}/\gamma_{\text{opt}}^{(h)}\rangle$ decreases as approximately $1/n$, while for REM and search the decrease is approximately $N^{-0.5}$. This decrease is expected theoretically for search (Childs and Goldstone, 2004). The fitted lines do not show exactly a square-root dependence (exponent of -0.5) due to the finite size effects for small numbers of qubits $n \leq 12$.

Thus, it can be seen that REM behaves like the search problem in a QW setting. For a precisely optimal hopping rate $\gamma_{\text{opt}}^{(h)}$, the success probability is high, but this instance-dependent hopping rate is hard to predict, unlike for the analytically tractable QW algorithm for unstructured search. Without this precise hopping rate, QWs perform no better than guessing for the search

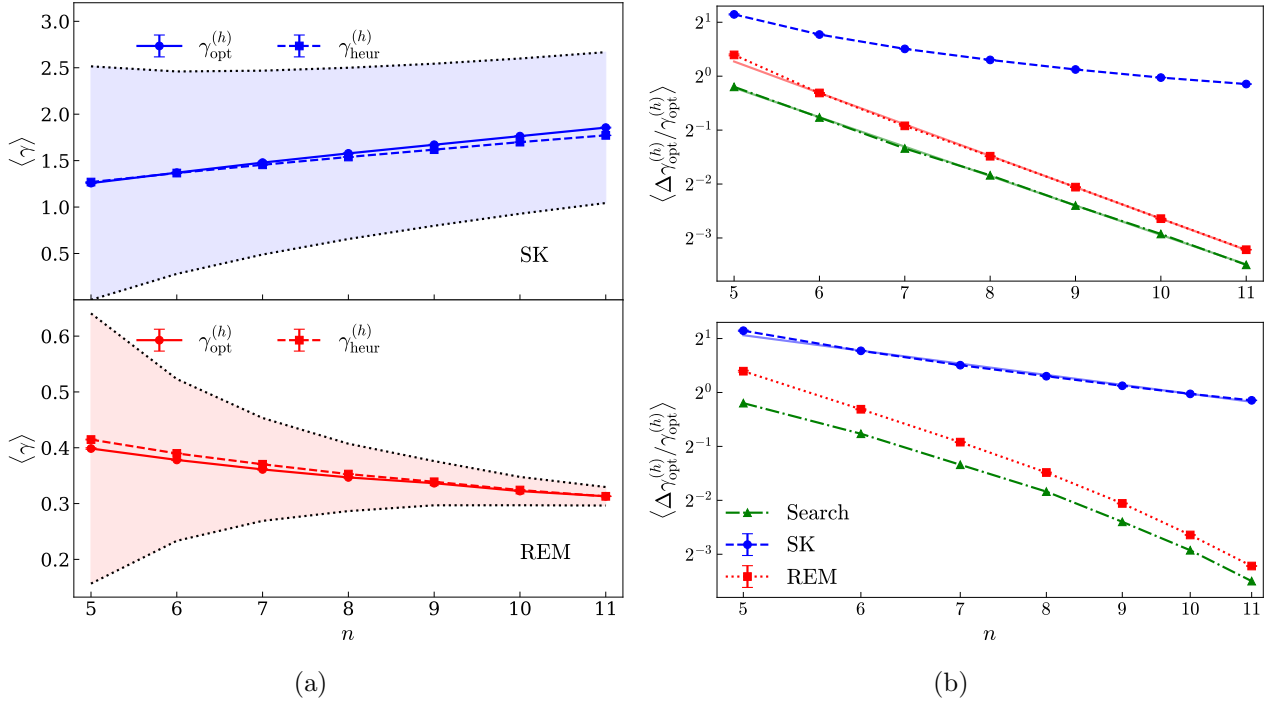


Figure 3.4: (a) Average optimal (circles, solid line) and heuristic (squares, dashed line) hopping rates, $\langle \gamma_{\text{opt}}^{(h)} \rangle$ and $\gamma_{\text{heur}}^{(h)}$, against number n of qubits for SK (top, blue) and REM (bottom, red). The shaded regions bordered by dotted lines indicate the average tolerance range $\langle \Delta \gamma_{\text{opt}}^{(h)} \rangle$ to non-optimal hopping rates, defined as full width at half maximum (FWHM) of the probability peak surrounding $\gamma_{\text{opt}}^{(h)}$, as illustrated in Fig. 3.2. (b) Log-linear plot (top) and log-log plot (bottom) of average *fractional* tolerance range $\langle \Delta \gamma_{\text{opt}}^{(h)} / \gamma_{\text{opt}}^{(h)} \rangle$. REM (red squares, dotted line) shows an exponential decrease, fitting to a line (red, solid line) with a gradient of -0.583 ± 0.006 in the log-linear plot. SK (blue circles, dotted line) shows a polynomial decrease, fitting to a line (solid blue) in the log-log plot with a gradient of -1.09 ± 0.04 . The same quantity for the search problem calculated the same way is also shown (green triangles, dash-dotted line), and it fits well to a line (solid green) in the log-linear plot with a gradient of -0.546 ± 0.004 .

problem and for REM. In contrast, QWs applied to SK give a better-than-guessing success probability $P_\infty > 1/N$ for the heuristic hopping rate $\gamma_{\text{heur}}^{(h)}$ calculated according to Eq. (3.9).

With the conditions under which a better-than-guessing success probability can be achieved characterised for the three problem types, SK, REM, and search, I will turn to the scaling of this success probability with problem size N .

3.2.4 Success probability

Fig. 3.5 shows how the single-time success probability $P(t_f)$ varies with the measurement time t_f for two typical 11-qubit examples of SK and REM. In the REM case, the behaviour is similar to that shown in Fig. 2.5(a) for search: an oscillatory nature indicating the dominance of a two-level avoided-crossing feature, but with evidence of the population of other energy-levels that lead to finite-size effects in search. For REM, these finite-size effects are more pronounced, and are instance-dependent. The random nature of the REM problems means there is not such a clear cut off size, as there is for the search problem, above which finite size effects are negligible. In any case, based on search, it is to be expected that finite size effects to be significant at $n = 11$. For SK, the behaviour is quite different from search or REM. There is no indication of dominant oscillatory behaviour; instead, these plots show unpredictable, highly instance-dependent fluctuating dynamics for all the sizes I have used. This indicates that for SK, the behaviour is determined by the excitation of many energy levels.

As with finding a suitable hopping-rate γ , both REM and SK differ from the search problem in that there is no practical way to find the optimal measurement time $t_f^{(\text{opt})}$; a different approach must be taken instead. As already noted for the search problem, this can be handled by using the time averaged probabilities defined in Eq. (3.1). I first consider the infinite-time probability P_∞ , as defined in Eq. (3.2), since it is easy to calculate. Fig. 3.6 shows the average infinite-time success probability $\langle P_\infty \rangle$ against the number n of qubits for the two problems using both the optimal $\gamma_{\text{opt}}^{(h)}$ and heuristic $\gamma_{\text{heur}}^{(h)}$ hopping rates. For SK, this gives exponential decay with the number of qubits n in both cases: the average probability $\langle P_\infty \rangle$ changes with n according to

$$\langle P_\infty \rangle = \begin{cases} \tilde{O}(N^{-0.402 \pm 0.001}) & \text{with } \gamma_{\text{opt}}^{(h)} \\ \tilde{O}(N^{-0.417 \pm 0.002}) & \text{with } \gamma_{\text{heur}}^{(h)} \end{cases}, \quad (3.11)$$

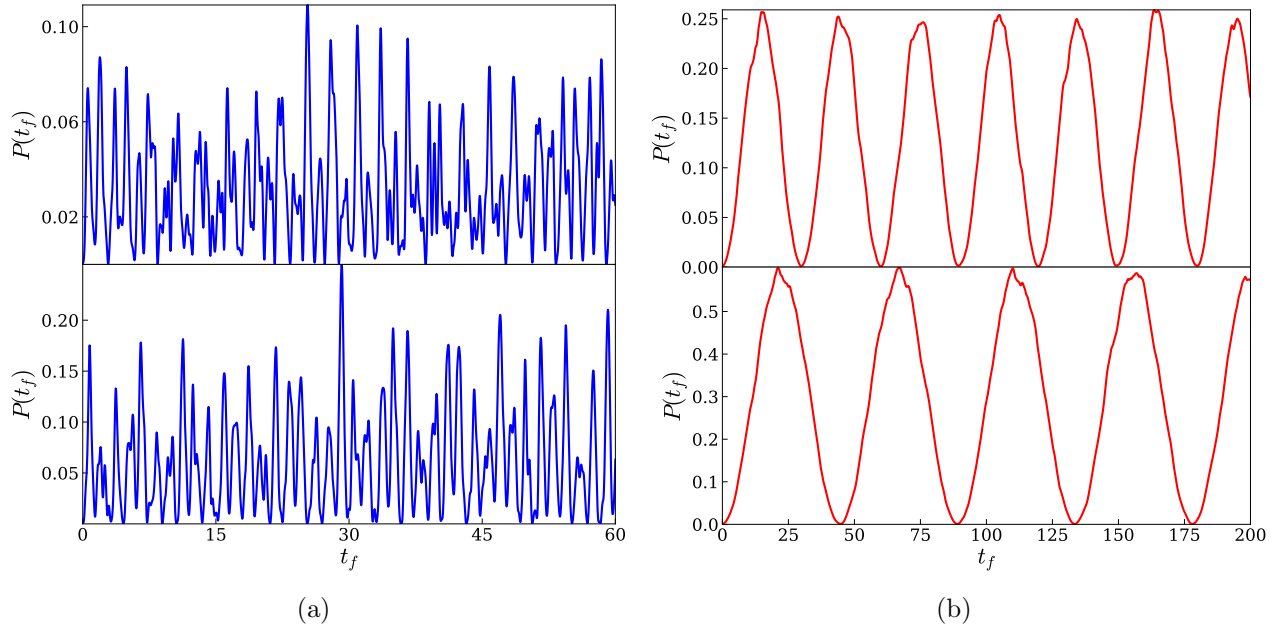


Figure 3.5: Instantaneous success probability $P(t_f)$ against measurement time t_f for a QW on 2 typical 11-qubit SK examples (a) and on 2 typical 11-qubit REM examples (b), using $\gamma_{\text{opt}}^{(h)}$.

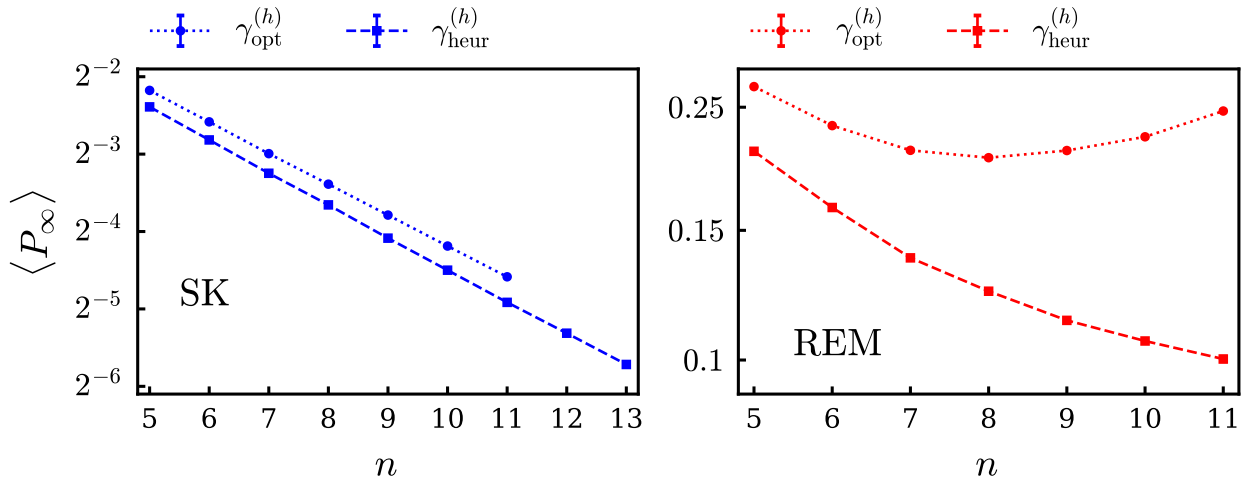


Figure 3.6: Blue, left: Log-linear plot of average infinite time success probability $\langle P_\infty \rangle$ against number of qubits n for SK, using optimal (circles, dotted line) and heuristic (squares, dashed line) hopping rates $\gamma_{\text{opt}}^{(h)}$ and $\gamma_{\text{heur}}^{(h)}$. The data fit $\log_2 \langle P_\infty \rangle = (-0.402 \pm 0.001)n + (-0.174 \pm 0.008)$ and $\log_2 \langle P_\infty \rangle = (-0.417 \pm 0.002)n + (-0.32 \pm 0.01)$ respectively. Red, right: Log-linear plot of the same quantities for REM. In this case, the probability stays at constant order for the optimal rate and decays for the heuristic rate.

where $\tilde{O}$ may neglect factors logarithmic in its argument. That is, using the heuristic hopping rate $\gamma_{\text{heur}}^{(h)}$ instead of the optimal hopping rate $\gamma_{\text{opt}}^{(h)}$ has only a minor impact on the average success probability $\langle P_{\infty} \rangle$.

For REM, the behaviour is quite different. With the optimal hopping rate $\gamma_{\text{opt}}^{(h)}$ the success probability P_{∞} is of constant order but with a pronounced dip. This behaviour is similar to that seen for the search problem, where the dip seen in Fig. 2.5(b) is a finite-size effect. This similarity is expected, given the similarity between the dynamical behaviour shown in Fig. 2.5(a) for search and in Fig. 3.5(b) for REM. With the heuristic hopping rate $\gamma_{\text{heur}}^{(h)}$ for REM, a significantly reduced success probability P_{∞} is seen compared to the optimal case. That is, the heuristic is performing poorly, despite the good agreement shown in Fig. 3.4(a).

The clear difference in behaviour between SK and REM can be explained by the different tolerances $\Delta\gamma_{\text{opt}}^{(h)}$ to deviations from the optimal hopping rate $\gamma_{\text{opt}}^{(h)}$ shown in Fig. 3.4(a) and Fig. 3.4(b). For SK, the tolerance range is broad enough for the heuristic to lie within it, while for REM the heuristic hopping rate $\gamma_{\text{heur}}^{(h)}$ almost always misses this range entirely even though it is close to the average optimal hopping rate $\langle \gamma_{\text{opt}}^{(h)} \rangle$.

3.2.5 Mixing times

The results in subsection 3.2.4 show a numerically determined an average success probability scaling with problem size of $\sim \tilde{O}(N^{-0.42})$ for a QW finding SK spin glass ground states, using the heuristic hopping rate $\gamma_{\text{heur}}^{(h)}$. This is based on the infinite-time success probability P_{∞} , i.e., uniform sampling from the distribution of all possible runtimes. I subsequently investigated the time dependence in more detail to answer the question “is it possible to sample from a finite run time and still obtain the same speed up?” Since $P(0) = 1/N$ corresponds to random guessing, there must be a minimum time before which it is not effective to measure.

I have defined a *mixing-time* $\tau_{\text{mix}}^{(\epsilon)}$ to be the latest time, t , for which the time averaged probabilities $\bar{P}(0, t)$ and $\bar{P}(0, 2t)$ at the two times t and $2t$ differ by a fraction greater than the fluctuation parameter ϵ ,

$$\tau_{\text{mix}}^{(\epsilon)} = \max\{t : \left| \frac{\bar{P}(0, t) - \bar{P}(0, 2t)}{\bar{P}(0, t)} \right| > \epsilon\}. \quad (3.12)$$

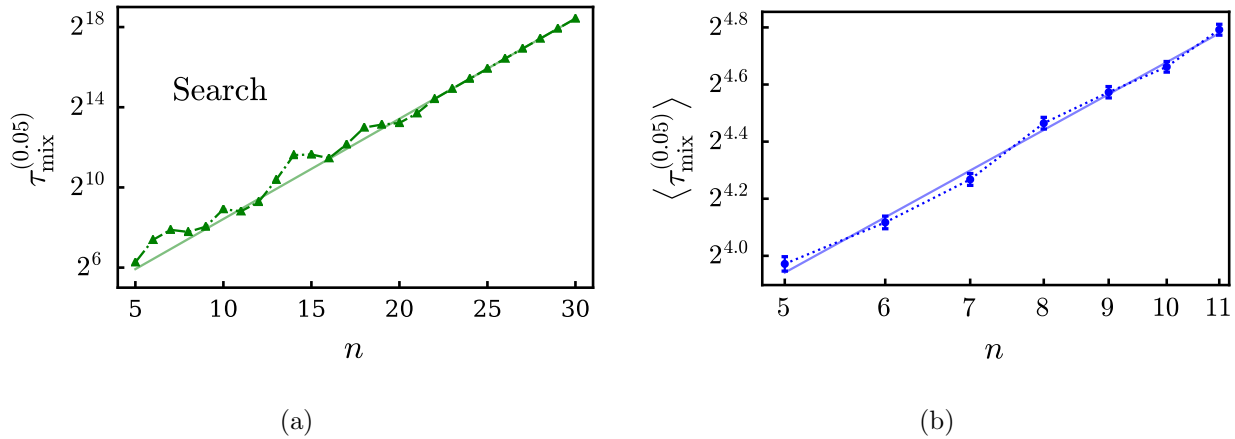


Figure 3.7: (a) Log-linear plot of the mixing time $\tau_{\text{mix}}^{(0.05)}$ for search, using $\gamma_{\text{opt}}^{(h)}$. The solid line of best fit is $\log_2 \tau_{\text{mix}}^{(0.05)} = (0.5000 \pm 0.0002)n + (3.424 \pm 0.006)$, with finite-size effects dominating at small numbers of qubits $n \lesssim 20$. (b) Log-log plot of the average mixing time against system-size n for SK, using $\gamma_{\text{opt}}^{(h)}$. The solid line of best fit is $\log_2 \langle \tau_{\text{mix}}^{(0.05)} \rangle = (0.74 \pm 0.03) \log_2 n + (2.23 \pm 0.08)$.

This definition of $\tau_{\text{mix}}^{(\epsilon)}$ is based on similar definitions found in prior work (Aharonov et al., 2001), with modifications for computational convenience. I have numerically estimated the mixing-time $\tau_{\text{mix}}^{(0.05)}$ for each SK instance up to $n = 11$ qubits, using the optimal hopping rate $\gamma_{\text{opt}}^{(h)}$ for each instance. To do this, I simulated the QW dynamics for a successively-doubling duration until a time at which the condition is met was reached. The fluctuation parameter $\epsilon = 0.05$ corresponds to a deviation of 5%. To verify that the mixing-time $\tau_{\text{mix}}^{(0.05)}$ correctly captures the relevant dynamical timescale, I have also numerically estimated it for the search problem at each system size from $n = 5$ to $n = 30$ qubits. QW on the unstructured search problem can be mapped to the symmetric subspace, allowing larger sizes to be analysed. The mixing-time $\tau_{\text{mix}}^{(0.05)}$ for search exhibits the expected exponential timescale: the solid green line of best fit in Fig. 3.7(a) has the expected scaling with problem size N of $\tau_{\text{mix}}^{(0.05)} = \tilde{O}(N^{1/2})$.

For search, the scaling is dominated by the run time, the success probability is $O(1)$. However, this behaviour only emerges clearly above $n \sim 20$. Below this, the behaviour is influenced by the finite-size effects that arise due to population of higher energy levels. This means it is not useful to analyse the behaviour of the REM time scaling; finite size effects mask the scaling behaviour for computationally tractable sizes. However, unlike search and REM, the SK behaviour is influenced by higher energy levels at all sizes, through the frustration provided by the random couplings between the spins. Hence, such finite-size effects are not evident in SK; the behaviour is already dominated by the frustration at small sizes. Fig. 3.7(b) shows a

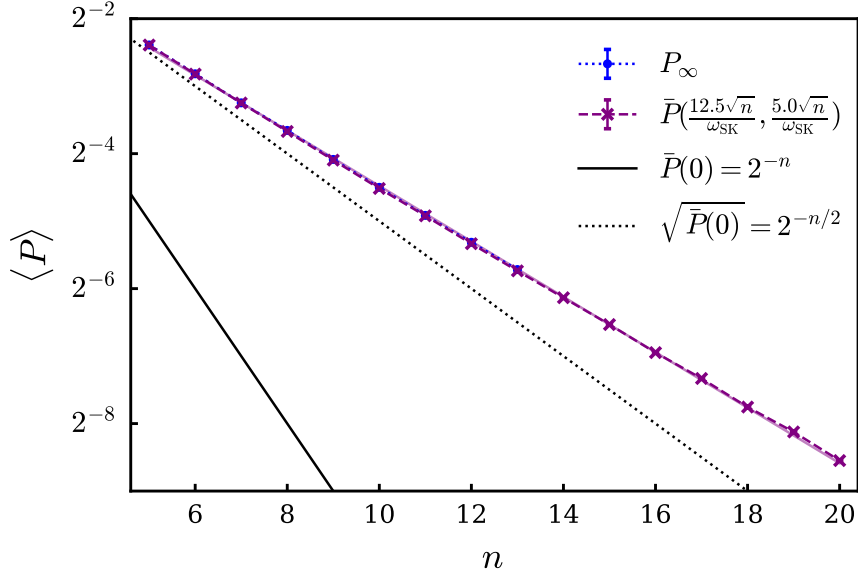


Figure 3.8: Log-plot of average success probability using $\gamma_{\text{heur}}^{(h)}$ against number of qubits for infinite-time (blue circles and dotted line) as in Fig. 3.6, and averaged over the short time window $12.5n^{\frac{1}{2}} \leq t \leq 17.5n^{\frac{1}{2}}$ (purple crosses and dashed line). The short time data are fit by $\log_2 \langle \bar{P} \rangle = (-0.410 \pm 0.002)n + (-0.37 \pm 0.02)$ (solid purple line). The 2^{-n} probability when measuring at $t = 0$, equivalent to randomly guessing (solid black line), and its square-root $2^{-n/2}$ (dotted black line) are also shown for comparison.

log-log plot of the mixing-time averaged over the ensemble $\langle \tau_{\text{mix}}^{(0.05)} \rangle$. The solid blue line of best fit has a logarithmic scaling with problem size N of

$$\langle \tau_{\text{mix}}^{(0.05)} \rangle = O(n^{0.74 \pm 0.03}) \simeq O([\log_2 N]^{0.75}). \quad (3.13)$$

Thus it contributes a logarithmic factor to the overall scaling. It is worth emphasising that, while this single-run timescale is polynomial in the number of spins n , the overall timescale is still exponential in n due to the exponential number of repeats required to achieve $O(1)$ success probability.

To confirm the subsidiary nature of the time scaling for each SK run, Fig. 3.8 shows a log-plot comparing, for the heuristic hopping rate $\gamma_{\text{heur}}^{(h)}$, the success probability P_∞ in the infinite-time case (as in Fig. 3.6) and in the case of an early, logarithmically-scaling (with respect to N) measurement window $12.5n^{\frac{1}{2}} \leq t \leq 17.5n^{\frac{1}{2}} \equiv (t_{\text{short}}, \Delta t_{\text{short}})$. This $n^{0.5}$ scaling of the window is even shorter than the fitted scaling of $n^{0.75}$, although at these sizes the difference is not significant. This finite-time probability $\bar{P}(t_{\text{short}}, \Delta t_{\text{short}})$ is similar to the infinite-time

probability P_∞ : the solid purple line of best fit in Fig. 3.8 has a scaling with problem size N of

$$\langle \bar{P}(t_{\text{short}}, \Delta t_{\text{short}}) \rangle = \tilde{O}(N^{-0.410 \pm 0.002}). \quad (3.14)$$

This should be compared with Eq. 3.11, where the value of the exponent for the average infinite time success probability P_∞ with the heuristic hopping rate $\gamma_{\text{heur}}^{(h)}$ is given by -0.417 ± 0.002 .

As the dominant factor in the total runtime comes from the required number of repeats, and because the single-run timescale contributes only a logarithmic factor, these results constitute good numerical evidence for an average total runtime which scales with problem size N as $\sim \tilde{O}(N^{0.41})$ for using QWs to find spin glass ground states, over the range of N in my data sets. This scaling is better than the best possible (quadratic) speed up achievable for QW applied to unstructured search. Moreover, it comes without the requirement for exponential precision in setting the hopping rate that renders practical use of QWs to solve unstructured search difficult for large problems. In the next section, I will present some insights into where the improvement over search comes from.

3.3 Computational mechanisms

3.3.1 Role of correlations in SK

To investigate whether the energy correlations with Hamming distance in SK play a significant role in the computational process of finding the ground state with a QW, I performed three additional sets of numerical tests.

Firstly, with the same SK instances, I simulated a similar QW but used a complete graph Hamiltonian $\hat{H}_K$, defined in Eq. (2.22), instead of the hypercube graph Hamiltonian $\hat{H}_h$. This removes the correspondence of Hamming-distance between classical states with the distance between those states on the graph: for the complete graph, every state is one unit (edge) away from every other state. In terms of the Hamiltonian, the transverse Ising term is replaced by sums of products of up to n Pauli- X operators that flip up to n qubits at the same time, in all possible combinations. For each SK instance up to $n = 11$, I estimated the optimal hopping rate $\gamma_{\text{opt}}^{(K)}$ for the complete graph, and then used it to calculate the infinite-time probability P_∞ .

Secondly, I constructed ‘scrambled SK’ instances, denoted sSK, by randomizing which state corresponds to which energy in the SK instances. The Hamiltonians this produces have identical energy spectra to the SK instances, but without the correlations between energy difference and Hamming distance on the hypercube graph. This approach has similarities with previous work (Farhi et al., 2008, 2011; Hen, 2014). For each sSK instance, I estimated the optimal hopping rate $\gamma_{\text{opt}}^{(h)}$, which is different from that used for the ordinary SK versions. This hopping rate was then used to calculate P_{∞} .

Thirdly, I sorted the eigenenergies of each REM instance in increasing size and assigned them to the computational basis states in the order of a binary-reflected Gray code on their bitstrings, to arrive at a problem denoted REMGC. This added some amount of Hamming-distance structure to the problem Hamiltonians by ensuring that the closest energies are assigned to states that differ by only a single bit-flip. For each REMGC instance, I estimated an optimal hopping rate $\gamma_{\text{opt}}^{(h)}$, which is different from that used for the ordinary REM problem. This was used to calculate the infinite-time probability P_{∞} . While REMGC is not a hard problem as defined, it provides a useful example to compare with how the QW finds the ground state of a SK spin glass.

These three variants provide separate tests of the influence of the graph structure (choice of walk Hamiltonian) and problem structure (pairwise correlations in SK). Fig. 3.9 shows how the infinite-time probability P_{∞} varies with the number of qubits n for these three variants, alongside SK and REM on a hypercube graph from Fig. 3.6, while Fig. 3.10 shows how the average *fractional* tolerance range $\langle \Delta\gamma_{\text{opt}}^{(h)}/\gamma_{\text{opt}}^{(h)} \rangle$ varies with the number of qubits n , alongside SK and REM on a hypercube graph from Fig. 3.4(b)

In both cases, the five variants are clearly split into two groups: those with REM- and search-like behaviour and those with SK-like behaviour. Removing the correlations from SK by scrambling the energies (sSK) results in behaviour like REM and search. Moreover, removing the correspondence between distance and Hamming weight by using the complete graph instead of the hypercube also changes the SK problem behaviour to be like REM and search. In the opposite direction, inserting pairwise correlations into REM via a Gray code (REMGc) results in problems that are much more like SK than like the REM problems on a hypercube graph.

These results imply that that the problem structure (in this case the pairwise correlations in SK) needs to be matched by a compatible driver Hamiltonian (in this case the hypercube/transverse

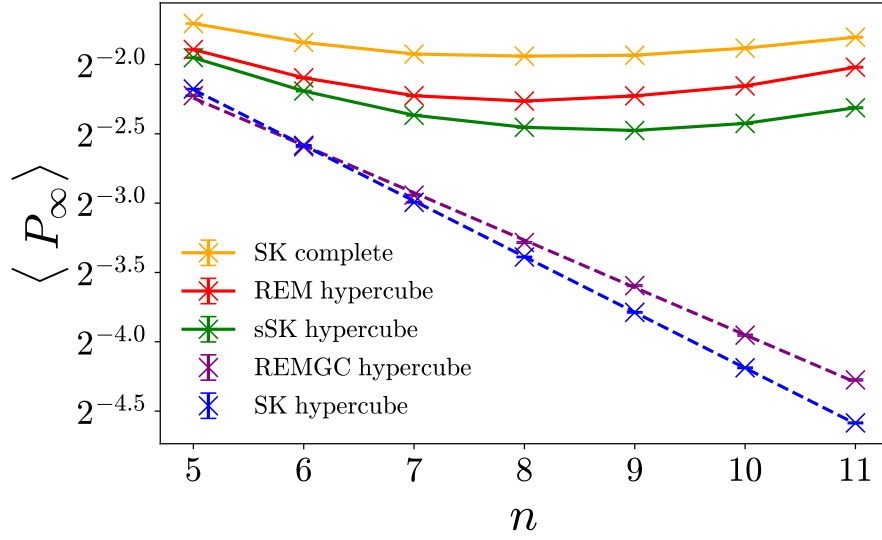


Figure 3.9: Log-linear plot showing the dependence on number of qubits n of the average success probability P_∞ for SK on hypercube (blue), REM on hypercube (red), sSK on hypercube (green), SK on complete-graph (orange) and REMGC on hypercube (purple). Dashed lines are linear fits where appropriate, while solid curves join the points together as a visual aid. The optimal hopping rates $\gamma_{\text{opt}}^{(h)}$ are used in all cases.

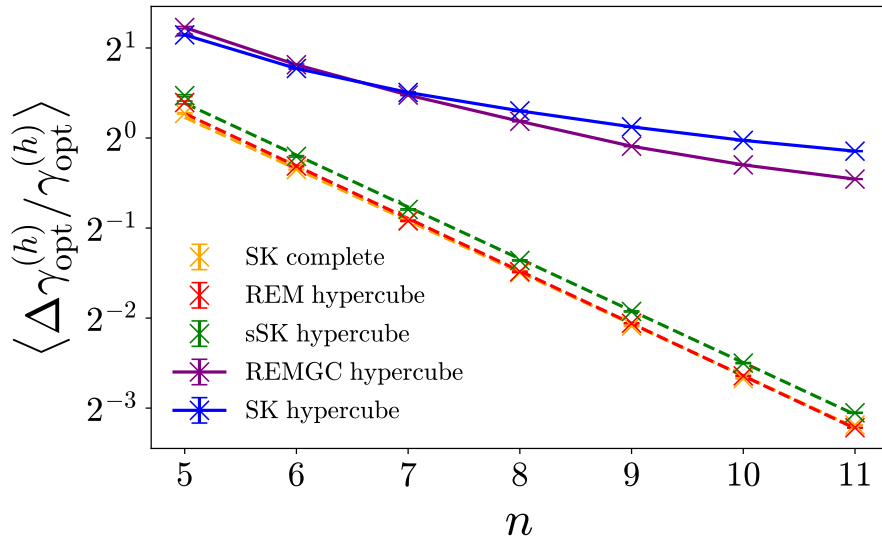


Figure 3.10: Log-linear plot showing the dependence on number of qubits n of the average fractional tolerance range $\langle \Delta \gamma_{\text{opt}}^{(h)} / \gamma_{\text{opt}}^{(h)} \rangle$ for SK on hypercube (blue), REM on hypercube (red), sSK on hypercube (green), SK on complete-graph (orange) and REMGC on hypercube (purple). Dashed lines are linear fits where appropriate, while solid curves join the points together as a visual aid. The optimal hopping rates $\gamma_{\text{opt}}^{(h)}$ are used in all cases.

Ising) to obtain better than quadratic scaling. This type of local structure in the solution space is exploited in many classical algorithms. For example, classical Monte Carlo optimizations that use a single bit flip update rule are naturally using this hypercube structure. Using a complete graph instead would correspond to flipping a random number of bits, which is equivalent to guessing at each step.

3.3.2 Energy conservation dynamics

QW time evolution is unitary, and there is no time dependence in the Hamiltonian that can lead to energy gain or loss by the system. Hence, it is important to consider how it can find a lower energy state than it starts in (with respect to $\hat{H}_P$) with any better-than-guessing probability. For the search problem, this happens through an analog of Rabi flopping (see Fig. 2.5), cycling between the initial and solution states. However, the dominant avoided level crossing structure is not present in the spin glasses to provide this mechanism.

As I will discuss here, there is in fact a very generic mechanism (also described independently by [Hastings \(2019\)](#)) that relies on starting in the ground state of the walk part of Hamiltonian $\hat{H}_G$. Let $\langle \hat{O} \rangle_{\psi(t)}$ for operator $\hat{O}$ be defined by $\langle \psi(t) | \hat{O} | \psi(t) \rangle = \langle \hat{O} \rangle_{\psi(t)}$. Then, by linearity, and the definition of $\hat{H}$ in Eq. (2.21), the energy expectation at time t is

$$\langle \hat{H} \rangle_{\psi(t)} = \langle \gamma \hat{H}_G \rangle_{\psi(t)} + \langle \hat{H}_P \rangle_{\psi(t)}. \quad (3.15)$$

Due to the unitarity of the evolution under a time-independent Hamiltonian, this expectation energy will not change over time, giving

$$\langle \hat{H} \rangle_{\psi(t)} = \langle \hat{H} \rangle_{\psi(0)}. \quad (3.16)$$

which yields

$$\gamma \left(\langle \hat{H}_G \rangle_{\psi(t)} - \langle \hat{H}_G \rangle_{\psi(0)} \right) = \langle \hat{H}_P \rangle_{\psi(0)} - \langle \hat{H}_P \rangle_{\psi(t)}. \quad (3.17)$$

As $|\psi(0)\rangle$ is chosen to be the ground state of $\hat{H}_G$, the LHS must be non-negative. Furthermore, as $|\psi(0)\rangle$ is not an eigenstate of $\hat{H}$, some dynamics are guaranteed to occur and so the LHS must become positive at early times. Therefore, the RHS must also be non-negative always

and positive at early times. Thus, taking any final time t_f gives the inequality

$$\frac{1}{t_f} \int_{t=0}^{t_f} dt \langle \hat{H}_P \rangle_{\psi(t)} < \langle \hat{H}_P \rangle_{\psi(0)}. \quad (3.18)$$

Eq. (3.18) shows that performing time evolution under the computational QW Hamiltonian from the initial state $|\psi(0)\rangle$ is guaranteed to lower the energy of the system with respect to $\hat{H}_P$ (the expectation value $\langle \hat{H}_P \rangle_{\psi(t)}$). This implies that the overlap with low energy eigenstates of $\hat{H}_P$ will increase, at least for short times. A measurement in the computational basis will thus be on average more likely than a random guess to produce a low energy state.

Starting in a low energy state is thus important for the success of the QW algorithm. This can be verified by examining the performance of the QW with higher energy starting states. I have done this by calculating the success probability of short time (as in subsection 3.2.5) QWs on SK problems with initial states $|\psi(0)\rangle$ chosen to be eigenstates of the hypercube graph Hamiltonian $\hat{H}_h$ of the form $|\pm\rangle \otimes |\pm\rangle \otimes \dots$ (in this form, the usual uniform superposition state is $|+\rangle \otimes |+\rangle \otimes \dots$). These states $|\pm\pm\dots\rangle$ have initial hypercube graph Hamiltonian expectation values $E_{\text{init}} \equiv \langle \pm\pm\dots | \hat{H}_h | \pm\pm\dots \rangle = 2\gamma_{\text{heur}}^{(h)} \text{Count}_{(-)}(|\pm\pm\dots\rangle)$ where $\text{Count}_{(-)}$ is the function that counts the number of single-qubit $|-\rangle$ states in its argument. As these states have zero problem-Hamiltonian expectation values $\langle \pm\pm\dots | \hat{H}_{\text{SK}} | \pm\pm\dots \rangle = 0$, E_{init} is also the actual expectation energy with respect to the full Hamiltonian $\langle \pm\pm\dots | \hat{H} | \pm\pm\dots \rangle$. A typical $n = 11$ qubit example is shown in Fig. 3.11, where the data points and error bars describe means and standard errors taken over samples of states from the different energy manifolds of the hypercube graph Hamiltonian $\hat{H}_h$. It can be seen that success probability drops off rapidly as the energy increases. The linear fit in the logarithmic plot (Fig. 3.11(b)) suggests an exponential decay of the form $\bar{P}(t_{\text{short}}, \Delta t_{\text{short}}) \propto 2^{(-0.45 \pm 0.01)E_{\text{init}}}$.

The importance of starting in a low energy state implies that encoding prior information into the initial state will help, provided this is given in the form of a lower energy state than the uniform superposition state. This could be the final state from a previous run, for example. It is also necessary to bias the walk Hamiltonian so that its ground state matches this biased initial state. Since this starting state is a known computational basis state, it is possible to do this biasing for suitably designed hardware.

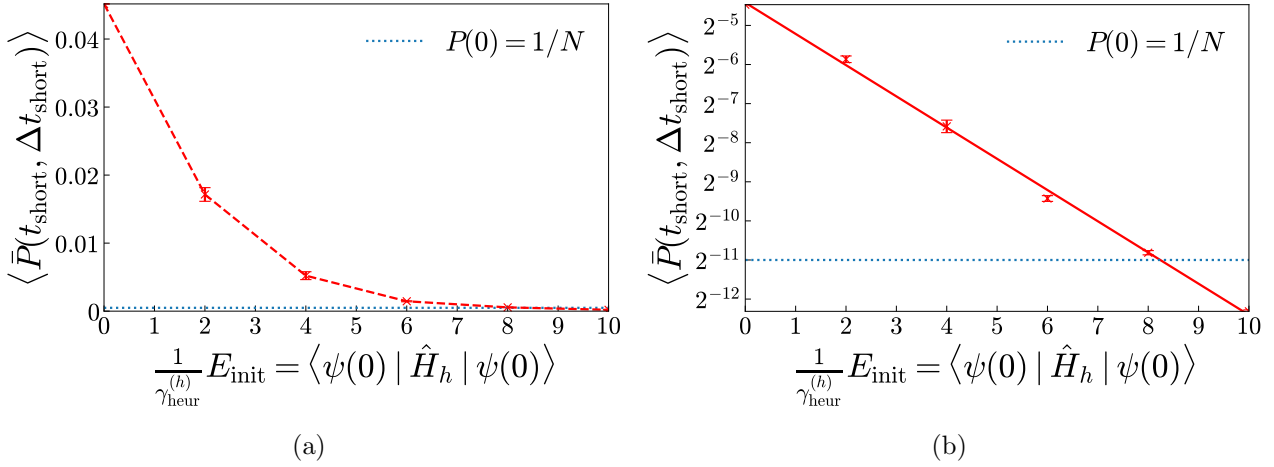


Figure 3.11: (a) Success probability $\bar{P}(t_{\text{short}}, \Delta t_{\text{short}})$ of a short time QW applied to a typical $n = 11$ qubit SK instance against the (rescaled) energy E_{init} of the initial state, which are chosen to be eigenstates of the hypercube graph Hamiltonian $\hat{H}_h$ of the form $|\pm\rangle \otimes |\pm\rangle \otimes \dots$. The data points and error bars describe means and standard errors taken over samples of states from the different energy manifolds of the hypercube graph Hamiltonian $\hat{H}_h$. The dotted blue line indicates random guessing probability. It can be seen that success probability drops off rapidly as the energy increases. (b) The same quantities but with probability plotted on a logarithmic axis. The solid line is a linear fit, with fit parameters suggesting an exponential decay of the form $\bar{P}(t_{\text{short}}, \Delta t_{\text{short}}) \propto 2^{(-0.45 \pm 0.01)E_{\text{init}}}$.

For many optimization problem applications, it is helpful to find a low energy state, even if it is not actually the true ground state. From this point of view, that QWs necessarily lower the expectation energy with respect to the problem Hamiltonian is very appealing as a computational mechanism. This argument by itself does not provide a guaranteed scaling or quantum speed up, but it does explain how the QW dynamics work in this setting, where there is no way to lose (or gain) energy. It is possible to generalise these arguments beyond time-independent Hamiltonians (see Chapter 5) to include monotonic functions $A(t)$ and $B(t)$ in Eq. (2.7).

To illustrate the energy redistribution mechanism, the plots in Fig. 3.12 show how the expectation value $\langle \hat{H}_h \rangle_{\psi(t)}$ of the walk Hamiltonian (green solid-line) and the expectation value $\langle \hat{H}_P \rangle_{\psi(t)}$ of the problem Hamiltonian (red solid-line) vary during a QW. The instantaneous success probability $P(t)$ (faint grey) is included to show that the timescale used is long enough for significant dynamics to take place. A typical 10-qubit SK example is shown in Fig. 3.12(a) and a typical 10-qubit REM example is shown in Fig. 3.12(b), both on the hypercube using their respective optimal hopping rates $\gamma_{\text{opt}}^{(h)}$. Also shown is the ground state eigenvalue $\langle \hat{H}_P \rangle_{E_0^{(P)}}$ of the problem Hamiltonian (red, dash-dotted line) and the ground state eigenvalue $\langle \hat{H}_h \rangle_{\psi(0)}$ of

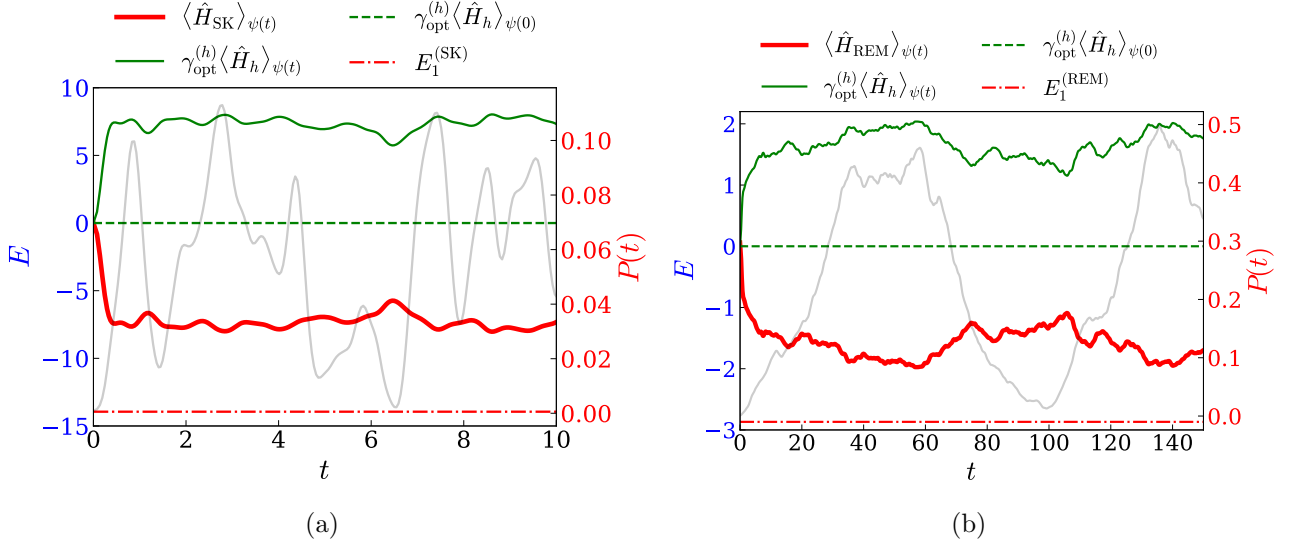


Figure 3.12: The (scaled) expectation value $\gamma_{\text{opt}}^{(h)} \langle \hat{H}_h \rangle_{\psi(t)}$ of the walk Hamiltonian (green, thin solid line) and the expectation value $\langle \hat{H}_P \rangle_{\psi(t)}$ of the problem Hamiltonian (red, thick solid line) for a typical 10 qubit (a) SK and (b) REM instance. The ground state energy eigenvalues of the walk Hamiltonian (green, dashed line) and problem Hamiltonians (red, dash-dotted line) are also shown. To illustrate that significant dynamics take place over the timescales used, the instantaneous probabilities $P(t)$ are also shown (grey, faint line). The energy values are on the left axes, while probability values are on the right axes.

the walk Hamiltonian (green, dashed line). In both SK and REM, the initial evolution takes the state away from the $\hat{H}_h$ ground state, raising the $\hat{H}_h$ expectation value, and thereby lowering the $\hat{H}_P$ expectation value to a point around which it fluctuates for the duration simulated. This clearly shows the energy redistribution mechanism at work, and the short time scale over which it appears.

3.4 Chapter summary

In this chapter, I have shown numerically that computation by continuous-time quantum walk (QW) is a viable computational method for finding ground states of hard spin glass problems. I have presented strong numerical evidence for a better-than-search polynomial quantum speed up over random guessing, with a scaling of the average single run success probability $\sim \tilde{O}(N^{-0.41})$, using data sets of size $5 \leq n \leq 20$ spins ($32 \leq N \leq 1,048,576$). Moreover, and importantly, this is obtained without the need to set parameters exponentially precisely, as is required when applying QWs to the unstructured search problem. The hopping rate γ , that determines the relative strengths of the walk and problem Hamiltonians, can be estimated from

the overall energy scales, which are determined by the hardware and encoding of the problem.

To explain why QWs are able to do better than quantum searching in this case, I have compared variants on the spin glass problems that remove or add pairwise correlations, and compared the hypercube graph walk Hamiltonian with the complete graph walk Hamiltonian. This showed that the combination of pairwise correlations in the encoding of the problem and a matching single spin flip walk Hamiltonian is required to exploit the correlations. The single spin-flips driven by the transverse field terms $\hat{X}_j$ in the hypercube quantum walk Hamiltonian are the correct operators for the pairwise interaction terms $\hat{Z}_j\hat{Z}_k$ in the spin glass Hamiltonian. A single spin flip on either qubit j or k changes the energy for that term from high to low, or vice versa. Since the way the problem is encoded into the Hamiltonians is a choice made by the algorithm designer, and there are known methods to convert higher order terms to pairwise terms (Bremner et al., 2002; Dattani, 2019), this mechanism can be used both for its computational advantages and practicality for hardware implementation as the transverse Ising Hamiltonian.

To explain how QWs are able to find low energy states when the closed quantum dynamics have no mechanism for losing energy, I detailed how starting in the ground state of the walk part of the Hamiltonian guarantees dynamics that decrease the expectation value of the energy with respect to the problem Hamiltonian. This also ensures that prior information can be provided by starting in lower energy states, from which improved solutions can be found. Exploiting this process will allow an optimal quantum algorithm to be built from multiple QW runs that use the information gained from prior runs. Performing multiple QW runs in early, noisy quantum hardware is a more viable approach than maintaining coherence for sufficiently accurate adiabatic algorithms. QWs may also be simpler to implement since they do not require time dependent controls. The work in this chapter thus provides a significant advance in understanding how to exploit QWs in practical hardware for optimization problems.

It is likely that further insights into the computational effectiveness of QWs in this transverse Ising Hamiltonian setting are to be found in current knowledge of spin glass phases in the presence of transverse fields. The spin glass transition itself is not fully understood, in neither the quantum nor classical case (see, e.g., Parisi, 1980; Fisher and Huse, 1987, 1988; Thirumalai et al., 1989; Larson et al., 2013; Young, 2017; Magalhaes et al., 2017). However, the phases of interest for computation are not the spin glass phases themselves, but the phases where

transitions between states are still occurring at a rapid enough rate to find solution states. Extremely long equilibration timescales are a defining property of all glass phases, including spin glasses (Bouchaud et al., 1998; Cugliandolo, 2002). Since the equilibration (mixing) times $\tau_{\text{mix}}^{(\epsilon)}$ found in subsection 3.2.5 for the SK spin glass only scale polynomially with the number of spins, it is most likely, that at the optimal hopping rates $\gamma_{\text{opt}}^{(h)}$, the QWs are not in a finite size precursor to a spin glass phase, but rather in a precursor to a paramagnetic phase, for which equilibration times can be fast. Given that the system should localize more in lower energy states for smaller transverse fields, it is reasonable that the optimal hopping rates $\gamma_{\text{opt}}^{(h)}$ occur near the edge of the precursor to the spin glass phase. Furthermore, the mild scaling of the width $\Delta\gamma_{\text{opt}}^{(h)}$ of the peak around the optimal hopping rate $\gamma_{\text{opt}}^{(h)}$ suggests that the regime where QWs perform well may correspond to the second paramagnetic phase observed in Magalhaes et al. (2017). Polynomial gaps have been found around the spin glass–paramagnetic phase transition in a related model in Knysh (2016).

As the work in this chapter is mostly numerical, the question of the asymptotic scaling of the algorithm is necessarily left open. In particular, I observed a range of hardness in the SK data sets and future work could investigate what fraction of the instances are actually hard for classical algorithms. In Chapter 4, I will apply similar techniques to MAX2SAT, characterise the hardness of small random instances in more detail, and establish QWs as a tool for hard optimization problems more generally. While general methods are known to speed up the best classical algorithms (Hartwig et al., 1984) for this type of problem (Montanaro, 2018, 2020), further work is required to determine whether an optimal QW algorithm can be devised that fully leverages the advantage from the correlations. Nonetheless, the work in this chapter represents a significant advance in developing QW computation for hard optimization problems, and provides key insights into the computational mechanisms that can be exploited over short timescales, well-suited to the limited coherence times of noisy, intermediate scale quantum hardware.

Chapter 4

MAX2SAT

In this chapter, I will consider an optimization problem defined quite differently than the Sherrington-Kirkpatrick (SK) spin-glass problem studied in detail in Chapter 3. The problem I will consider is the Maximum 2-Satisfiability (MAX2SAT) problem defined in subsection 2.1.4, using the Ising model mapping described in subsection 2.2.2. While the problem is defined very differently, the resulting problem Hamiltonian $\hat{H}_{\text{M2S}}$ when mapped to the Ising model looks quite similar. The main difference is that the coefficients J_{jk}, h_j takes values in $-1, 0, 1$ rather than being drawn from a normal distribution. Much of the structure of SK that was identified as being important to the dynamics of the continuous-time quantum walk computation (QW) and performance in Chapter 3 should also be present in MAX2SAT, and therefore similar behaviour is to be expected. This is one of the questions I will address in this chapter.

In (Crosson et al., 2014), a large number of MAX2SAT instances with $n = 20$ binary variables and $m = 60$ clauses were generated, mapped to the Ising model, and numerically subjected to a quantum annealing (QA) protocol of the form

$$H(t) = \left(1 - \frac{t}{T}\right) \frac{1}{2} \hat{H}_h + \frac{t}{T} \hat{H}_{\text{M2S}}, \quad (4.1)$$

where $\hat{H}_h$ is the hypercube/transverse field driver Hamiltonian from Eq. (2.23), $\hat{H}_{\text{M2S}}$ is the MAX2SAT instance problem Hamiltonian and T is the total duration, chosen to be $T = 100$. Of the $> 200\text{k}$ instances generated, 137 ‘hard’ instances were selected on the basis that they produced a success probability $P(100) < 10^{-4}$, and the authors have provided me with these instances. It should be noted that, while Crosson et al. (2014) describe the instances as

being hard for adiabatic quantum computing (AQC), the protocol is by definition far from the adiabatic limit for these instances, and so I will describe them as being hard for QA. To facilitate a comparison between the instances from [Crosson et al. \(2014\)](#) and some typical MAX2SAT instances, I have generated 10,000 instances at each size $5 \leq n \leq 20$ according to the same prescription: generating $m = 3n$ unique clauses and post-selecting only those instances with unique optimal assignments. MAX2SAT has also been studied in an experimental QA setting ([Santra et al., 2014](#)).

In section 4.1, I will demonstrate that the behaviour of QWs on SK spin-glass problems seen in Chapter 3 generalises to other problems by presenting the results of applying to MAX2SAT many of the same numerical analyses. I will also point out the differences where they exist. In section 4.2, I will apply AQC to the MAX2SAT instances and analyse the results, and then compare the AQC results to the QW results in section 4.3. In section 4.4, I will explore a powerful classical algorithm for MAX2SAT, illustrating the need for moving beyond QW and AQC to develop more powerful continuous-time quantum computing (CTQC) algorithms, an effort which will be a major focus of Chapter 5. Finally, in section 4.5, I will summarise the chapter and conclude.

4.1 Quantum walks on MAX2SAT and similarities with SK

In this section, I will analyse the performance and behaviour of QW on MAX2SAT problems in order to demonstrate that the results for QWs on SK spin-glass problems seen in Chapter 3 generalise to other problems. I will also point out the differences where they exist.

I will first consider the how the average success probability for QW on MAX2SAT depends on the number of variables (equivalently, number of qubits) n . Fig. 4.1 shows a log-linear plot of the average (over all 10,000 instances at each size) success probability $\langle \bar{P}(0, 100) \rangle$ against number of qubits n , both with the optimal hopping rates $\gamma_{\text{opt}}^{(h)}$, calculated for each instance by a brute-force search, and with the heuristic hopping rates $\gamma_{\text{heur}}^{(h)}$ for each n , based on the same ‘energy-spread’ calculation used for SK in Eq. (3.9). Lines of best fit are also shown, and their

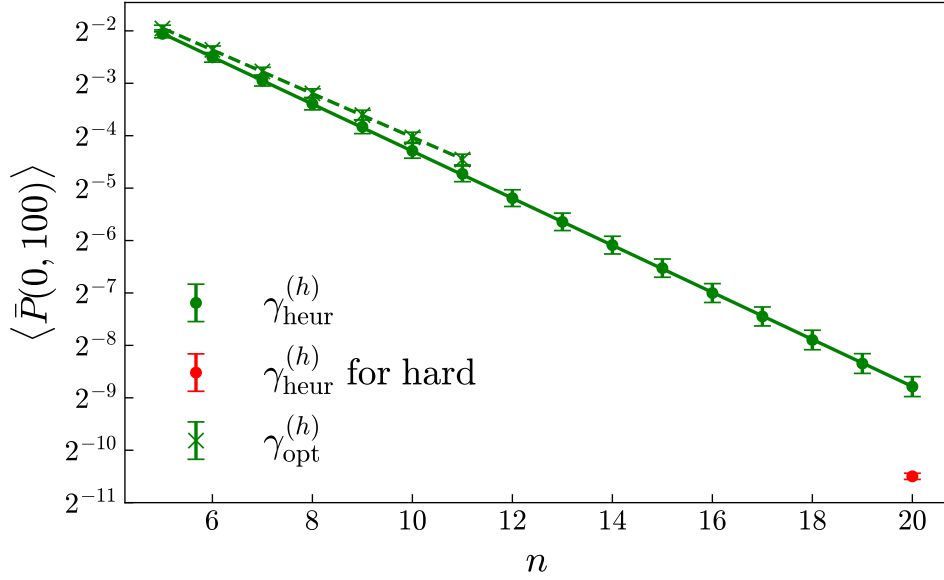


Figure 4.1: Log-linear plot of average (over all 10,000 instances at each size) success probability $\langle \bar{P}(0, 100) \rangle$, averaged over the time window $0 \leq t \leq 100$, using $\gamma_{\text{opt}}^{(h)}$ (green crosses and dashed line of best fit) and $\gamma_{\text{heur}}^{(h)}$ (green circles and solid line of best fit) against number of qubits n . The gradients of the fit lines imply scalings of $\langle \bar{P}(0, 100) \rangle \sim O(N^{-0.416 \pm 0.001})$ and $\langle \bar{P}(0, 100) \rangle \sim O(N^{-0.4490 \pm 0.0005})$ for $\gamma_{\text{opt}}^{(h)}$ and $\gamma_{\text{heur}}^{(h)}$ respectively. The red data point is an average over the $n = 20$ instances from Crosson et al. (2014).

gradients imply the scalings

$$\langle \bar{P}(0, 100) \rangle = \begin{cases} \tilde{O}(N^{-0.416 \pm 0.001}) & \text{with } \gamma_{\text{opt}}^{(h)} \\ \tilde{O}(N^{-0.449 \pm 0.005}) & \text{with } \gamma_{\text{heur}}^{(h)}. \end{cases} \quad (4.2)$$

As with the similar results for SK (although with the probability averaged over all time in that case) in Fig. 3.6 (scalings in Eq. (3.11)), the success probability with the heuristic hopping rate behaves similarly to the success probability with the optimal hopping rate, although the difference in scaling exponents is slightly larger for MAX2SAT than it is for SK.

The red data point is an average over the $n = 20$ instances from Crosson et al. (2014), and it can be seen that this point is at a significantly lower probability than the typical $n = 20$ instances. As these instances have been chosen specifically because they are hard for QA, this suggests that the same problem features that make an instance hard for QA tend to also make them hard for QW. Further work is required to determine exactly what these features are.

Having seen that the same energy-spread-based heuristic hopping rate $\gamma_{\text{heur}}^{(h)}$ as for SK works well for MAX2SAT, it is informative to examine whether this is for the same reasons, by examining

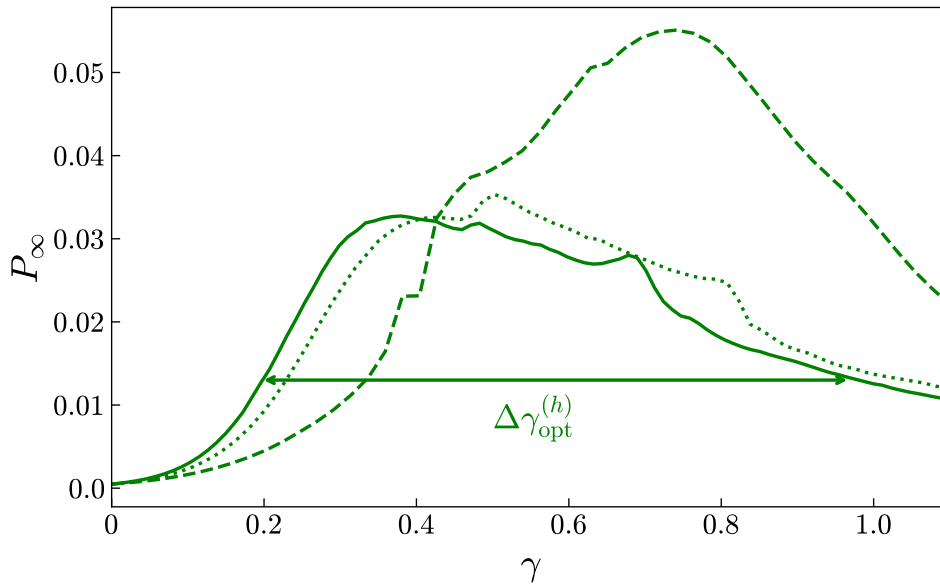


Figure 4.2: Infinite-time success probability P_∞ against hopping rate γ for 3 typical 11-qubit examples of MAX2SAT. Also indicated (for one example) is the width $\Delta\gamma_{\text{opt}}^{(h)}$ of the peak.

the tolerance of the success probability to deviations of the hopping rate from the optimal $\gamma_{\text{opt}}^{(h)}$. Fig. 4.2 shows the infinite-time success probability P_∞ against hopping rate γ for 3 typical 11-qubit examples of MAX2SAT. Also indicated (for one example) is the width $\Delta\gamma_{\text{opt}}^{(h)}$ of the peak. Comparing to Fig. 3.2, it can be seen that these low, broad, and bumpy peaks for MAX2SAT resemble the SK peaks much more closely than the random energy model (REM) peaks.

On a similar note, Fig. 4.3 shows an approximate probability density histogram $p(\gamma_{\text{opt}}^{(h)})$ of the numerically-found optimal hopping rate $\gamma_{\text{opt}}^{(h)}$ for $n = 11$ MAX2SAT, inferred based on 10,000 instances of this size. Also shown as a dashed line is the heuristic hopping rate $\gamma_{\text{heur}}^{(h)}$, calculated according to Eq. (3.9). While the $\gamma_{\text{opt}}^{(h)}$ is similarly broad as for SK, the heuristic falls slightly further from the peak of the distribution, indicating that there is room for improvement in finding, better more general heuristics - this will be discussed further in Chapter 5. However, as the curves of success probability as a function of hopping rate, like those in Fig. 4.2, are so broad, using this heuristic still works reasonably well, as can be seen in in Fig. 4.1.

Examining the tolerance of the success probability to deviations of the hopping rate from the optimal $\gamma_{\text{opt}}^{(h)}$ more broadly than just for a few instances (as in Fig. 4.2), Fig. 4.4 shows both the mean optimal hopping rate $\langle\gamma_{\text{opt}}^{(h)}\rangle$ (circles, solid line) and the heuristic hopping rate $\gamma_{\text{heur}}^{(h)}$ (squares, dashed line) against number n of qubits for MAX2SAT. The shaded regions indicate

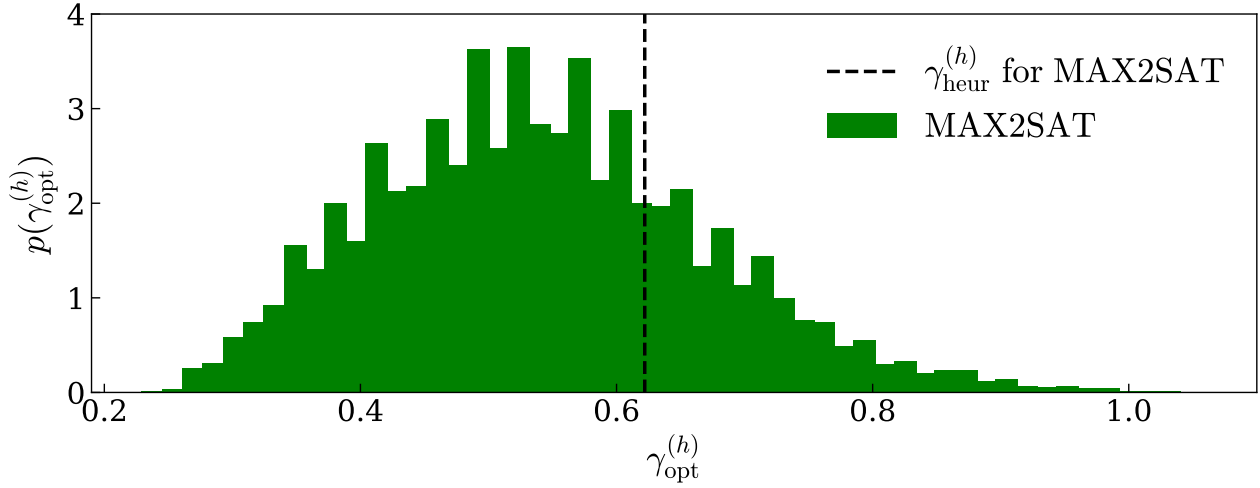


Figure 4.3: Histogram (showing the approximate inferred probability density $p(\gamma_{\text{opt}}^{(h)})$) of the numerically-found optimal hopping rates $\gamma_{\text{opt}}^{(h)}$ for the 10,000 11-qubit instances of MAX2SAT. The dashed line shows the heuristic hopping rate $\gamma_{\text{heur}}^{(h)}$, calculated according to Eq. (3.9).

the average tolerance range $\langle \Delta \gamma_{\text{opt}}^{(h)} \rangle$ to non-optimal hopping rates. This tolerance is defined as the full width at half maximum (FWHM) of the probability peak surrounding $\gamma_{\text{opt}}^{(h)}$, as illustrated in Fig. 4.2. Comparing this plot to Fig. 3.4(a), it can again be seen that MAX2SAT behaves much more like SK than REM. The only noticeable difference as compared to the SK plot is the greater deviation of $\gamma_{\text{heur}}^{(h)}$ from $\langle \gamma_{\text{opt}}^{(h)} \rangle$ for MAX2SAT, consistent with Fig. 4.3.

The dependence on n of the tolerance $\langle \Delta \gamma_{\text{opt}}^{(h)} \rangle$ is quantified more thoroughly in Fig. 4.5. This figure shows (on log-linear axes at the top and on log-log axes at the bottom) the average *fractional* tolerance $\langle \Delta \gamma_{\text{opt}}^{(h)} / \gamma_{\text{opt}}^{(h)} \rangle$ against number of qubits n for MAX2SAT. The fact that the data appear much more linear in the log-log plot implies that the tolerance decreases only polynomial in n , as is the case with SK (but not REM) in Fig. 3.4(b). However, the gradient of the line of best fit on the log-log plot implies a scaling of

$$\langle \Delta \gamma_{\text{opt}}^{(h)} / \gamma_{\text{opt}}^{(h)} \rangle \sim O(n^{-0.737 \pm 0.007}) = O([\log_2 N]^{-0.737 \pm 0.007}), \quad (4.3)$$

which is quite different than the $O(n^{-1.09 \pm 0.04})$ found for SK.

In subsection 3.3.1, I demonstrated the importance of the structure (in particular, the correlation between Hamming distance and problem energy difference between two computational basis states) for QWs on SK. I did this by making the comparison, among others, between the original SK problem and a version I called ‘scrambled SK’, denoted sSK, by randomizing which

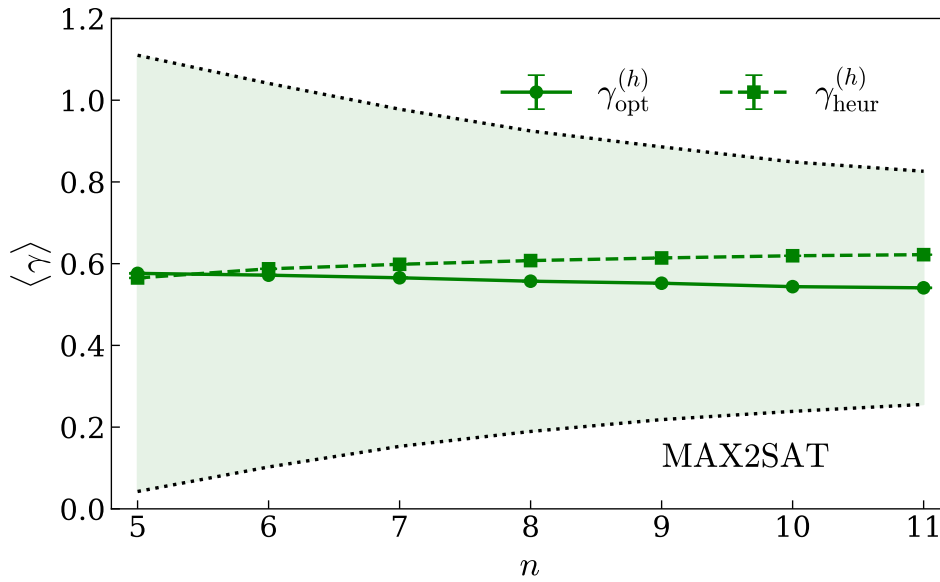


Figure 4.4: Average optimal (circles, solid line) and heuristic (squares, dashed line) hopping rates, $\langle \gamma_{\text{opt}}^{(h)} \rangle$ and $\gamma_{\text{heur}}^{(h)}$, against number n of qubits for MAX2SAT. The shaded regions bordered by dotted lines indicate the average tolerance range $\langle \Delta \gamma_{\text{opt}}^{(h)} \rangle$ to non-optimal hopping rates, defined as full width at half maximum (FWHM) of the probability peak surrounding $\gamma_{\text{opt}}^{(h)}$, as illustrated in Fig. 4.2.

state corresponds to which energy in the SK instances, completely removing the correlation. I have done the same to the MAX2SAT problem Hamiltonians, creating ‘scrambled MAX2SAT’ (sMAX2SAT). Fig. 4.6 shows (on log-linear axes) the dependence on the number of qubits n of the average success probability $\langle P_{\infty} \rangle$ for original MAX2SAT on hypercube (green) and sMAX2SAT on hypercube (blue). In both cases, the optimal hopping rates $\gamma_{\text{opt}}^{(h)}$ are used (a new set of optimal rates were calculated for sMAX2SAT as they are not in general the same as for the raw MAXSAT instances). Comparing to Fig. 3.9, it can be seen that removing the structure for MAX2SAT instances results in a problem type much more like REM than like raw MAX2SAT or SK. This indicates that there is structure in MAX2SAT instances playing a similar role to that in SK.

Turning to the importance of starting in a low energy state that was seen for SK in subsection 3.3.2, Fig. 4.7 shows the dependence of success probability $\bar{P}(0, 100)$ on the energy expectation value of the initial state $|\psi(0)\rangle$ of the QW. The method used is the same as for SK in Fig. 3.11: QWs were performed using initial states of the form $|\pm\rangle \otimes |\pm\rangle \otimes \dots$ (which are eigenstates of $\hat{H}_h$) and the success probabilities were calculated. Fig. 4.7(a) and Fig. 4.7(b) show the same data, but the latter is on log-linear axes, showing that the data fit well to a linear model,

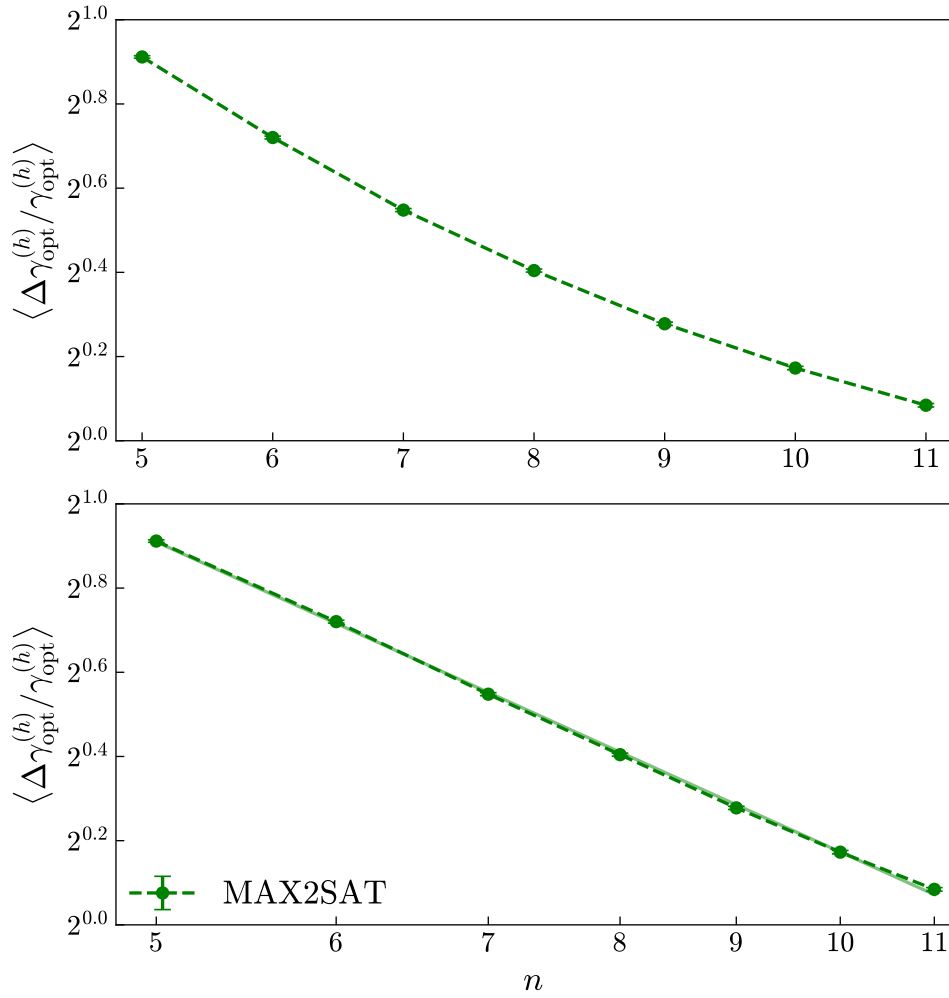


Figure 4.5: Log-linear plot (top) and log-log plot (bottom) of average *fractional* tolerance range $\langle \Delta \gamma_{\text{opt}}^{(h)} / \gamma_{\text{opt}}^{(h)} \rangle$ for MAX2SAT. The data appear much more linear in the log-log plot, implying a polynomial decrease, and the gradient of the line of best fit implies a scaling of $\langle \Delta \gamma_{\text{opt}}^{(h)} / \gamma_{\text{opt}}^{(h)} \rangle \sim O(n^{-0.737 \pm 0.007}) = O([\log_2 N]^{-0.737 \pm 0.007})$.

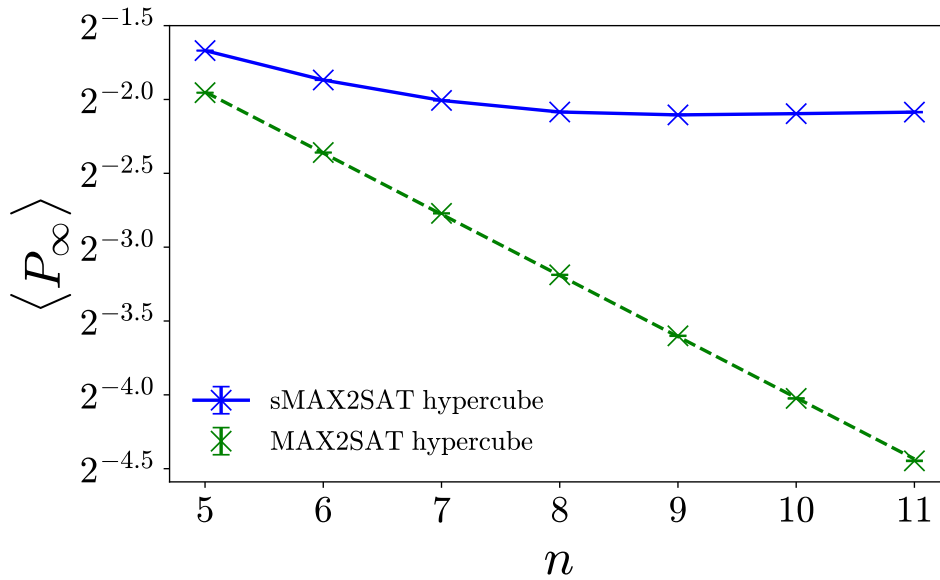


Figure 4.6: Log-linear plot showing the dependence on number of qubits n of the average success probability P_∞ for MAX2SAT on hypercube (green) and sMAX2SAT on hypercube (blue). The green dashed line is a line of best fit for MAX2SAT, while the solid curve joins the points for sMAX2SAT together as a visual aid. The optimal hopping rates $\gamma_{\text{opt}}^{(h)}$ are used in both cases.

indicating an exponential dependence, as was also found for SK. The gradient of the line of best fit implies an exponential decay of the form $\bar{P}(0, 100) \propto 2^{(-1.14 \pm 0.04)E_{\text{init}}}$.

4.2 Adiabatic quantum computing on MAX2SAT

In this section, I will analyse the performance of the AQC algorithm on the MAX2SAT instances. The measure of performance I will consider is the minimum duration $T_{0.99}$ of a linear AQC schedule (i.e a schedule of the form in Eq. (4.1)) that produces a success probability of at least 99%. This will enable comparisons with the QW results from section 4.1 which will be the focus of section 4.3.

Fig. 4.8 shows plots (on log-linear axes at the top and on log-log axes at the bottom) plots of the mean duration $\langle T_{0.99} \rangle$ at each number $n \leq 11$ of qubits. While simulations were performed for $n > 11$, a non-trivial proportion of instances at these larger sizes took prohibitively long and were not completed, and so these sizes have not been included in the plot. Both plots also show lines of best fit, and it can be seen that the linear fit is more appropriate for the top plot, suggesting an exponentially growing duration, but the fit is far from perfect, meaning

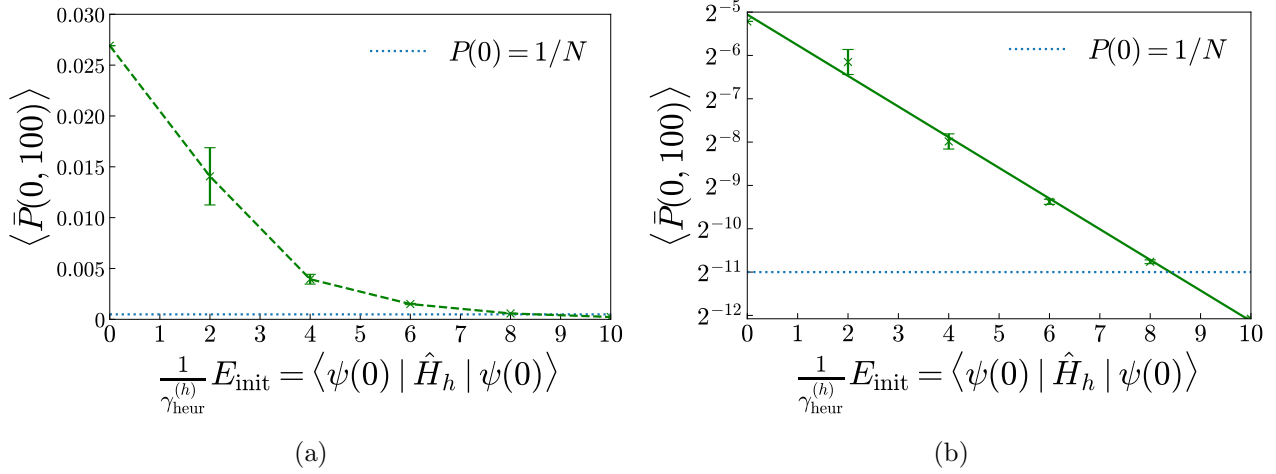


Figure 4.7: (a) Success probability $\bar{P}(0, 100)$ of a QW applied to a typical $n = 11$ qubit MAX2SAT instance against the (rescaled) energy E_{init} of the initial state, which are chosen to be eigenstates of the hypercube graph Hamiltonian $\hat{H}_h$ of the form $|\pm\rangle \otimes |\pm\rangle \otimes \dots$. The data points and error bars describe means and standard errors taken over samples of states from the different energy manifolds of the hypercube graph Hamiltonian $\hat{H}_h$. The dotted blue line indicates random guessing probability. It can be seen that success probability drops off rapidly as the energy increases. (b) The same quantities but with probability plotted on a logarithmic axis. The solid line is a linear fit, with fit parameters suggesting an exponential decay of the form $\bar{P}(0, 100) \propto 2^{(-1.14 \pm 0.04) E_{\text{init}}}$.

that strong conclusions cannot be reasonably drawn. As is, the gradient of the line of best fit implies a scaling of

$$\langle T_{0.99} \rangle \sim O(N^{0.274 \pm 0.005}). \quad (4.4)$$

Fig. 4.9 is similar to Fig. 4.8 but instead shows the *median* minimum duration $\text{Median}(T_{0.99})$ rather than the mean. Unlike Fig. 4.8, the linear fit for the median is most appropriate on the bottom (log-log) plot, suggesting a median duration that only grows polynomially. Specifically, the gradient of the line of best fit implies a scaling of

$$\text{Median}(T_{0.99}) \sim O(n^{0.84 \pm 0.01}) = O([\log_2 N]^{0.84 \pm 0.01}). \quad (4.5)$$

If the very different scalings of the mean and median AQC duration $T_{0.99}$ persist to larger sizes, they suggest a heavily skewed distribution (becoming increasingly skewed with increasing size) with a majority of instances requiring only moderate duration (accounting for the mildly scaling median) and a small number of instances requiring much longer (skewing the mean toward larger durations).

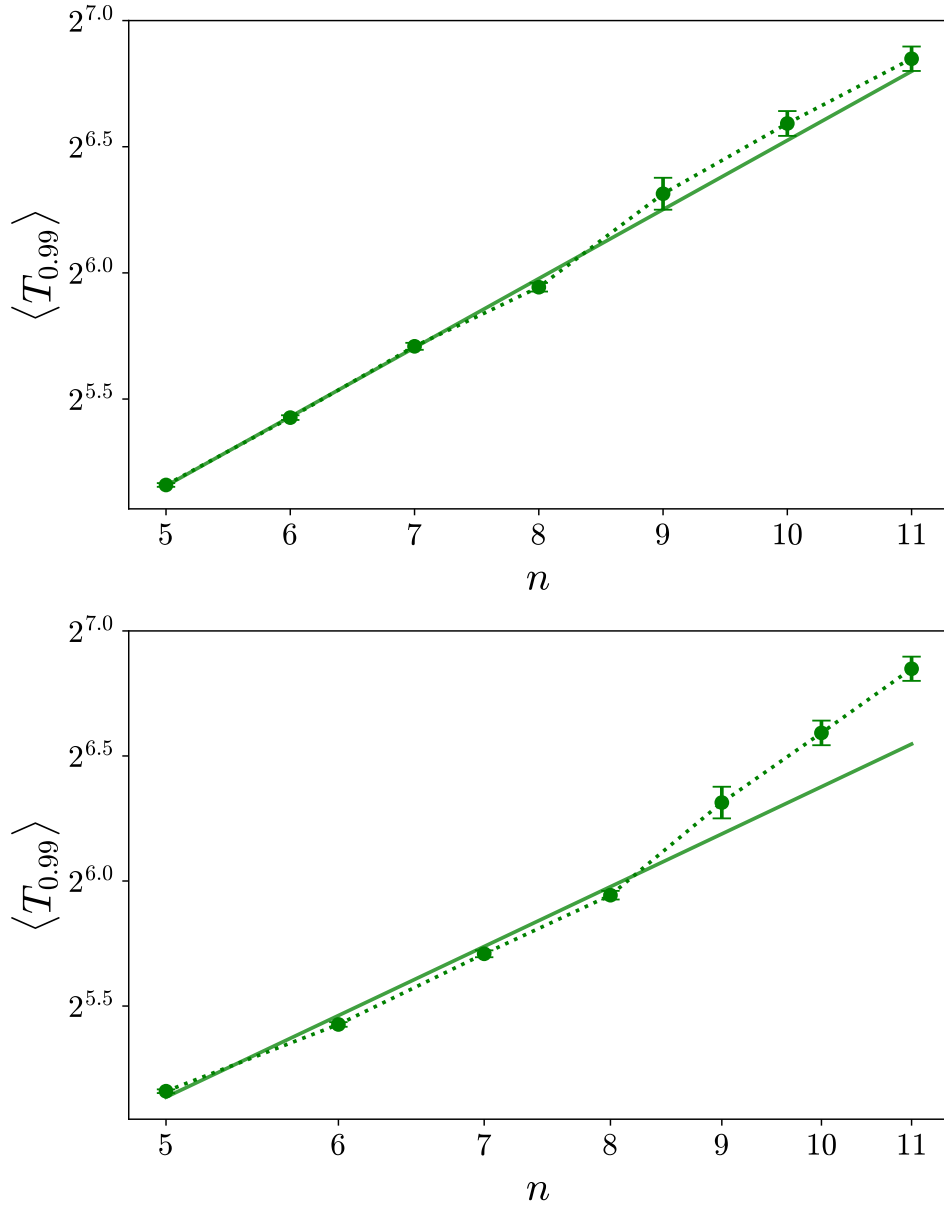


Figure 4.8: Log-linear (top) and log-log (bottom) plots of the *mean* minimum duration $\langle T_{0.99} \rangle$ of a linear AQC schedule that produces a success probability of at least 99% at each number n of qubits. While simulations were performed for $n > 11$, a non-trivial proportion of instances at these larger sizes took prohibitively long and were not completed. Solid lines of best fit are shown for both plots, and the fit appears to be best for the top plot, suggesting an exponentially growing duration. While the fit is far from perfect, meaning that strong conclusions cannot be reasonably drawn, the gradient of the line of best fit implies a scaling of $\langle T_{0.99} \rangle \sim O(N^{0.274 \pm 0.005})$.

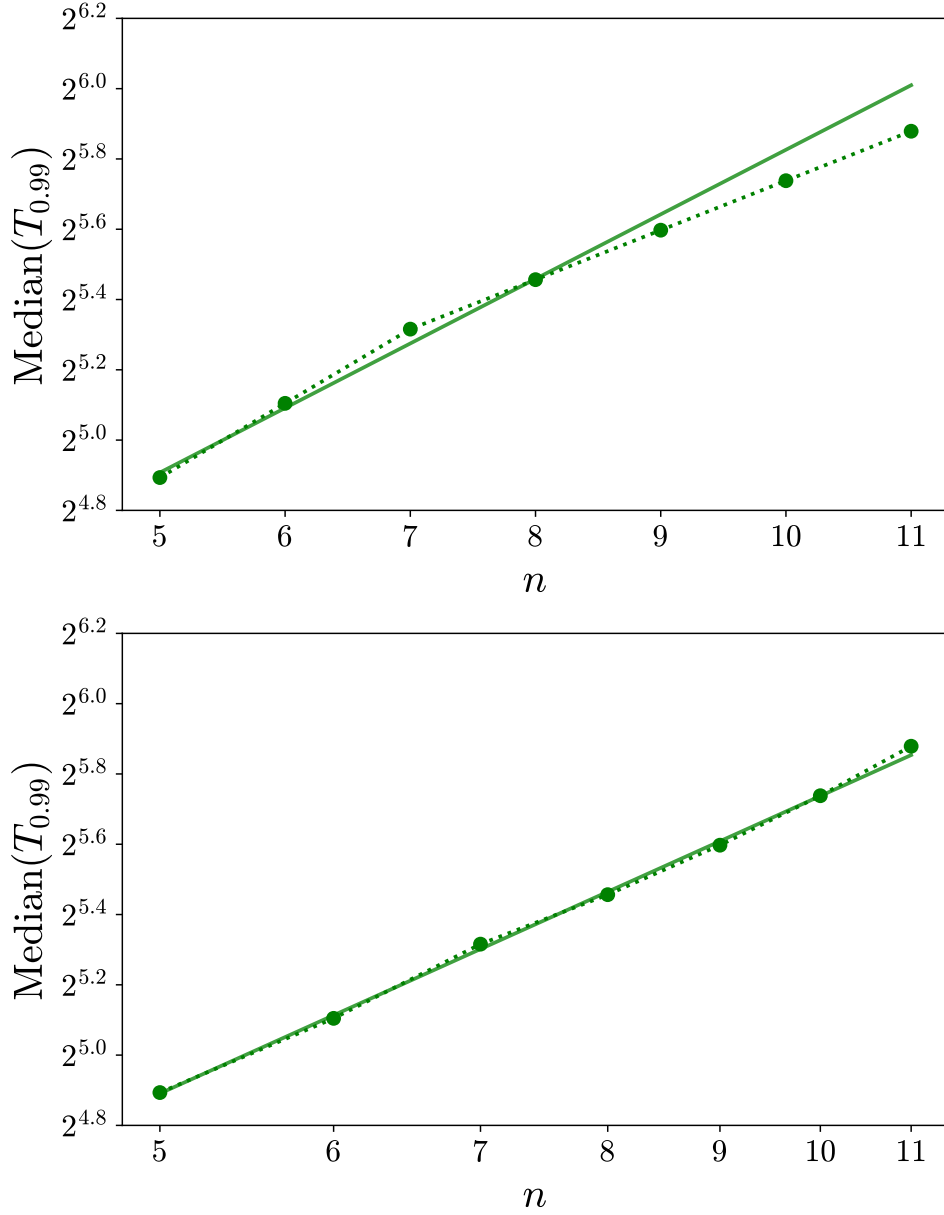


Figure 4.9: Log-linear (top) and log-log (bottom) plots of the *median* minimum duration $\text{Median}(T_{0.99})$ of a linear AQC schedule that produces a success probability of at least 99% at each number n of qubits. While simulations were performed for $n > 11$, a non-trivial proportion of instances at these larger sizes took prohibitively long and were not completed. Solid lines of best fit are shown for both plots, and the fit appears to be best for the bottom plot, suggesting a duration that grows only polynomially in n . The gradient of the line of best fit implies a scaling of $\text{Median}(T_{0.99}) \sim O(n^{0.84 \pm 0.01}) = O([\log_2 N]^{0.84 \pm 0.01})$.

4.3 Comparing QW and AQC

In section 4.1 it was seen that the average success probability for QW appears to scale as $\langle \bar{P}(0, 100) \rangle \sim O(N^{-0.449 \pm 0.005})$, while in section 4.2 it was seen that the average AQC duration appears to scale as $\langle T_{0.99} \rangle \sim O(N^{0.274 \pm 0.005})$. This indicates that, on average, AQC performs much better than QW on MAX2SAT instances; however, this does not rule out the possibility that there are certain kinds of instances on which QW performs better. In this section, I will compare the performance of AQC and QW on a more fine-grained level, including on a per-instance basis

For the first set of comparisons, I have separated the MAX2SAT instances at each size n into ‘hardness deciles’ based on their QW success probability $\bar{P}(0, 100)$; that is, the 10% of instances with the highest probability have been put into decile 1, then next highest 10% into decile 2, and so on, until the 10% of instances with the lowest probability have been put into decile 10. Similarly, I have separated the instances into hardness deciles based on their AQC duration $T_{0.99}$; that is, the 10% of instances with the shortest duration have been put into decile 1, then next shortest 10% into decile 2, and so on, until the 10% of instances with the longest duration have been put into decile 10.

In the first row of Fig. 4.10, the left-hand plot shows (on log-linear axes) QW success probability $\bar{P}(0, 100)$ of the instances arranged into deciles based on that same success probability (darker points/lines indicates harder deciles). A linear model fits well to the points in each decile, suggesting an exponential (in n) decay, as with the mean over the whole set of instances (see Fig. 4.1). The right-hand plot of the first row, shows the scaling exponent κ for each decile, which has been extracted from the lines of best fit in the left-hand plot. The exponents range from $\kappa = -0.390 \pm 0.002$ (easiest) to $\kappa = -0.551 \pm 0.007$ (hardest). The red data point in the left plot is an average over the instances from Crosson et al. (2014), and it can be seen that these are roughly as hard as the hardest 10% of the typical instances. As these instances were chosen in Crosson et al. (2014) as the hardest 137 instances for QA out of around 200k instances, the position of this point on the plot suggests that these instances are not as *extremely* hard for QW as they are for QA.

The second row of Fig. 4.10 is similar to the first, but for AQC duration $T_{0.99}$; that is, the left-hand plot shows (on log-linear axes) the AQC duration of the instances arranged into deciles

based on that same duration (darker points/lines indicates harder deciles). In the right-hand plot of the second row, the exponents κ range from $\kappa \approx 0.13$ (easiest) to $\kappa = 0.48 \pm 0.02$ (hardest). It is noteworthy that a big jump occurs between the exponent at decile 9 and at decile 10. This is consistent with the previous observation (from Fig. 4.8 and Fig. 4.9) that only a small portion of the instances scale harshly. However, that observation also suggests that the easier deciles should not scale exponentially, yet they appear to fit reasonably well to the lines in the left-hand plot. Further work is required to resolve this apparent contradiction.

The left-hand plot of the third row of Fig. 4.10 shows QW success probability $\bar{P}(0, 100)$ of the instances, but arranged into hardness deciles based on the AQC duration $T_{0.99}$. The darker points/lines indicates harder deciles in terms of AQC duration, but it can be seen that those deciles are typically also harder in terms of QW probability. The scaling exponents κ extracted from the linear fits and plotted in the right-hand plot confirm this, ranging from $\kappa = -0.384 \pm 0.002$ (easiest) to $\kappa = -0.575 \pm 0.004$ (hardest). This indicates that instances which are harder for AQC also tend to be harder for QW.

The fourth row of Fig. 4.10 makes the opposite comparison to the third row: the left-hand plot shows AQC duration $T_{0.99}$ of the instances, but arranged into hardness deciles based on the QW probability $\bar{P}(0, 100)$. The darker points/lines indicates harder deciles in terms of QW probability, but it can be seen that those deciles are typically also harder in terms of AQC duration. Linear models have been fitted to the deciles, but as the points deviate from these lines more than in the other plots, dashed lines connecting the data points are also included as a guide for the eye. The scaling exponents κ extracted from the linear fits and plotted in the right-hand plot confirm that hardness for AQC also increases as the QW deciles get harder. The exponents range from $\kappa = 0.094 \pm 0.005$ (easiest) to $\kappa = 0.55 \pm 0.02$ (hardest). This indicates that instances which are harder for QW also tend to be harder for AQC.

To compare directly on a per-instance basis, Fig. 4.11 shows a heatmap of the joint distribution of AQC duration $T_{0.99}$ and QW success probability $\bar{P}(0, 100)$ at $n = 15$, using the 9868 instances for which $T_{0.99}$ was successfully calculated. Visually, the heatmap indicates a reasonably high level of correlation between the two quantities, and numerical measures of correlation confirm this; for example the Spearman's rank correlation coefficient is ≈ -0.71 . While the level of correlation is high, it is not perfect. This means that there could be some kinds of instances that are easy for QW but hard for AQC and vice versa. Further work is required to determine

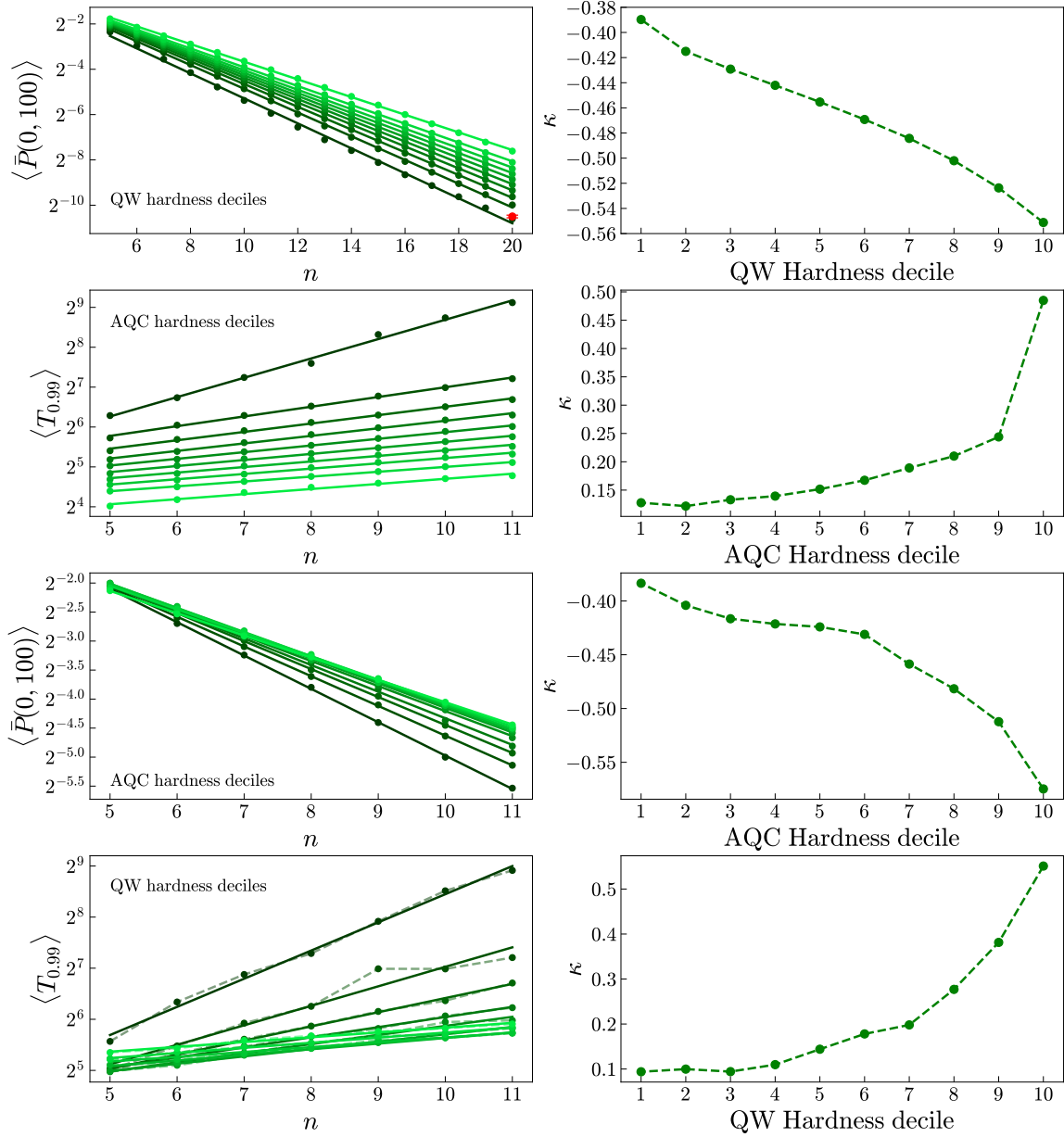


Figure 4.10: The MAX2SAT instances have been separated into ‘hardness deciles’ for both QW success probability $\bar{P}(0, 100)$ (lowest and highest probabilities in deciles 10 and 1 respectively) and AQC duration $T_{0.99}$ (longest and shortest durations in deciles 10 and 1 respectively) at each number of qubits n . The plots in the left column show (on log-linear axes): (1st row) QW success probability $\bar{P}(0, 100)$ of the instances arranged into deciles based on QW success probability. (2nd row) AQC duration $T_{0.99}$ of the instances arranged into deciles based on AQC duration. (3rd row) QW success probability $\bar{P}(0, 100)$ of the instances arranged into deciles based on AQC duration $T_{0.99}$. (4th row) AQC duration $T_{0.99}$ of the instances arranged into deciles based on QW success probability $\bar{P}(0, 100)$. A linear model has been fit (solid line) to each decile on each plot (dashed lines connecting the points are also shown in the 4th row plot, as there is more deviation from the fit lines than in other plots). Darker points/lines correspond to harder deciles. A scaling exponent κ has been inferred from the gradient of each linear fit. Each plot in the right column shows the scaling exponent κ for each decile in the plot to its left.

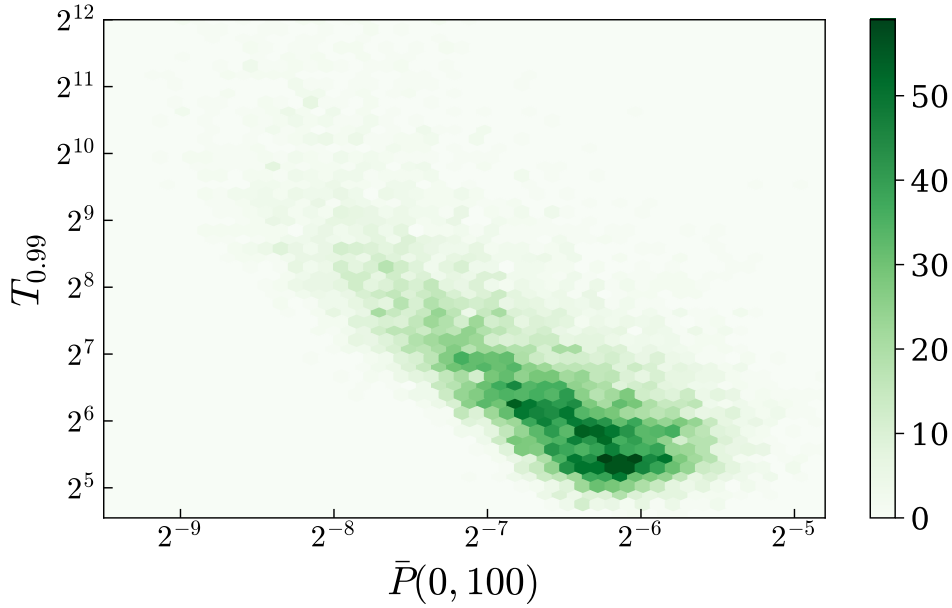


Figure 4.11: A heatmap showing the joint distribution of AQC duration $T_{0.99}$ and QW success probability $\bar{P}(0, 100)$ at $n = 15$, using the 9868 instances for which $T_{0.99}$ was successfully calculated. Visually, the heatmap indicates a reasonably high level of correlation between the two quantities, and numerical measures of correlation confirm this; for example the Spearman’s rank correlation coefficient is ≈ -0.71 .

what kinds of instances fall into these categories.

4.4 Classical algorithm for MAX2SAT

Sections 4.1, 4.2, and 4.3 considered the QW and AQC continuous-time quantum algorithms for the MAX2SAT problem, but it is useful to see how well classical algorithms can perform on this problem.

A powerful algorithm known as MIXSAT was developed by [Wang and Kolter \(2018\)](#) for solving general MAXSAT problems (defined as I have defined MAX2SAT but with clauses not limited to containing only two literals). The MIXSAT algorithm follows the branch-and-bound structure described in subsection 2.1.1. When applied to MAX2SAT problems from a 2016 MAXSAT solver competition, MIXSAT was found to significantly outperform even the best solvers that were used in the competition. MIXSAT incorporates a number of heuristic tricks to speed up the solve. When applied to the $n = 5$ to $n = 20$ instances that I have been using in this chapter, MIXSAT solved the problems so quickly that the runtimes were completely dominated by overhead, making it impossible to extract any useful information about problem hardness.

In order to analyse my MAX2SAT instances properly, I have implemented a branch-and-bound algorithm in the Python language (Van Rossum and Drake, 2003) that incorporates some of the core aspects of MIXSAT. In particular, the **Bound** component is based on a semidefinite programming method known as the ‘Mixing method’ (Wang et al., 2018), which uses a step-free iteration to hone in on a fixed point. The **Branch** component simply produces two new partial solutions by fixing the next variable (in index order) to ‘TRUE’ and ‘FALSE’. The **Cost** component checks the candidate solution against each clause and counts the number that are not satisfied. A major modification compared to the basic structure described in subsection 2.1.1 is that the output of the Mixing method in **Bound** is used to produce a ‘guess’ solution that is likely to have a low **Cost**. This allows ‘best_cost’ to be updated much more regularly, leading to much more pruning, significantly speeding up the solve. I will refer to this implementation as ‘MixBandB’.

I have applied MixBandB to all of the MAX2SAT instances I have generated, from $n = 5$ to $n = 20$. For each instance, I have calculated a quantity N_{calls} . This counts the number of ‘calls’ to the problem; that is, how many times the problem specification is accessed. This is the total number of iterations of the Mixing method (summed over each call to **Bound**) added to the total number of calls to **Cost**. N_{calls} serves as a measure of classical hardness.

The plot in Fig. 4.12 shows (on log-linear axes) the mean, taken over all 10,000 instances at each problem size n , $\langle N_{\text{calls}} \rangle$ of the number of problem calls for a run of the MixBandB algorithm at each problem size n . A line of best-fit has been fitted to $n \geq 12$, where it fits well (the line is solid in this region, and dotted at lower n), indicating an exponential (in n) increase. The gradient of the line of the best fit implies a scaling of

$$\langle N_{\text{calls}} \rangle = O(N^{0.278 \pm 0.002}). \quad (4.6)$$

This scaling is much better than the scaling for QW and possibly comparable to the scaling for AQC. It should be expected that the scaling for the actual MIXSAT algorithm would be even better (I check this in Fig. 4.15). The red point is a mean over the instances from Crosson et al. (2014), and it can be seen that this point is at a significantly higher number of calls than the typical $n = 20$ instances. As these instances have been chosen specifically because they are hard for QA, this suggests that the same problem features that make an instance hard for QA tend to also make them hard for the classical algorithm. Further work is required to determine

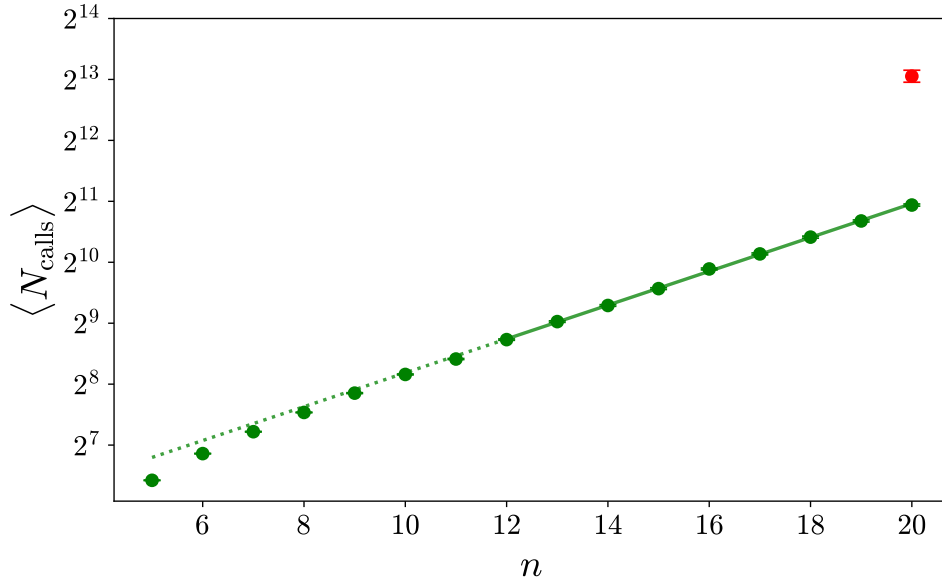


Figure 4.12: Log-linear plot of the mean number of problem calls $\langle N_{\text{calls}} \rangle$ at each problem size n for a run of the MIXBandB algorithm. The means are taken over all 10,000 instances at each size. A line of best-fit has been fitted to $n \geq 12$, where it fits well (the line is solid in this region, and dotted at lower n), indicating an exponential (in n) increase. The gradient of the line of the best fit implies a scaling of $\langle N_{\text{calls}} \rangle = O(N^{0.278 \pm 0.002})$. The red point is a mean over the instances from Crosson et al. (2014).

exactly what these features are.

To get a sense of how the number N_{calls} of problem calls is distributed, the top plot in Fig. 4.13 is a histogram of N_{calls} at $n = 20$, showing the approximate probability density $p(N_{\text{calls}})$ inferred from all the instances at this size. The 10,000 typical instances form the green histogram, while the instances from Crosson et al. (2014) form the red histogram. It can be seen that a majority of typical instances require a relatively small numbers of calls, but a small number of instances form a heavy tail that goes to much larger numbers of calls; this does not appear to be true for the (Crosson et al., 2014) instances. The bottom plot shows the same data, but binned and plotted on a logarithmic scale. Here, it can be seen more clearly that the typical instances form a bimodal distribution, suggesting that the instances separate into distinct ‘hard’ and ‘easy’ varieties. Moreover, it can be seen that the instances from (Crosson et al., 2014) contain a much higher proportion of ‘hard’ instances. Further work is required to establish what features differ between the two kinds of instances.

To understand how classical hardness (according to N_{calls}) correlates with quantum hardness (at least for QW and AQC), Fig. 4.14 shows two heatmaps. The top heatmap shows, on log-

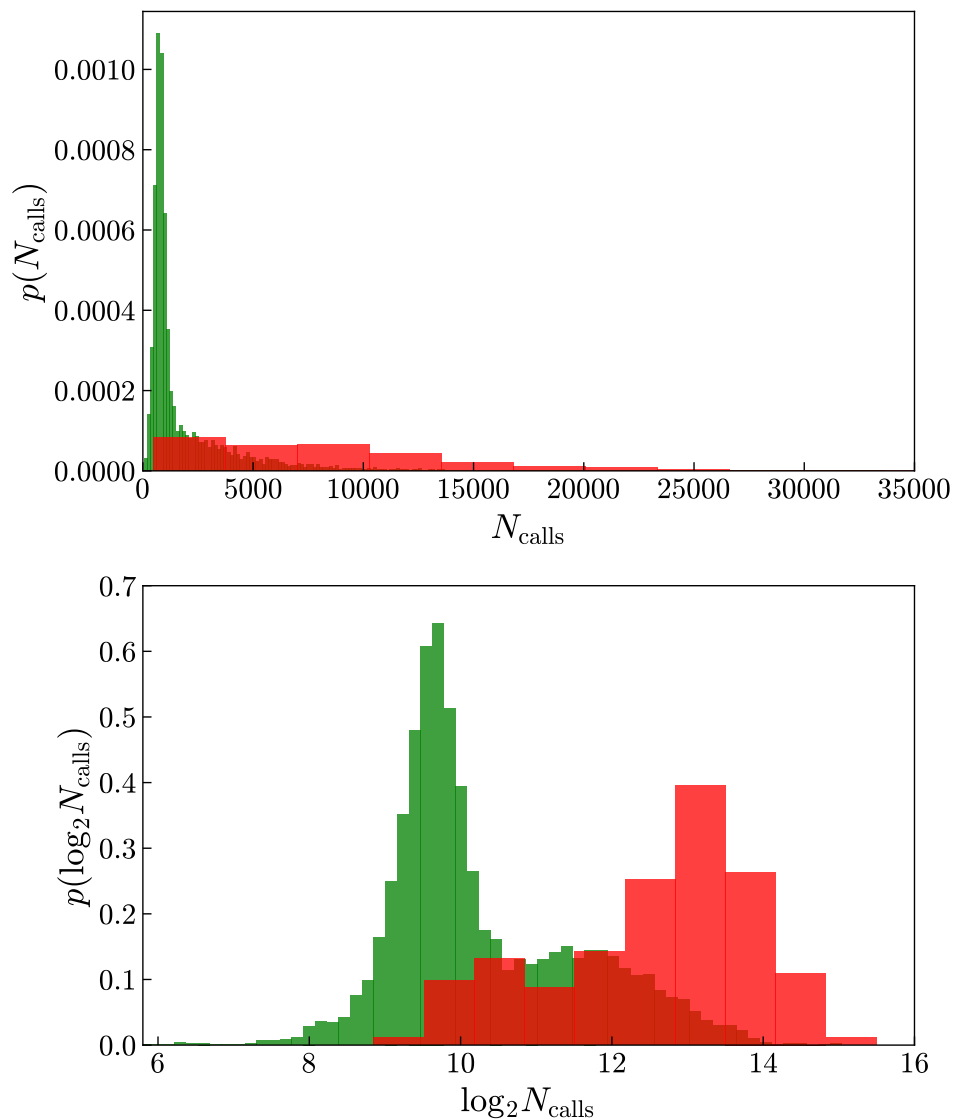


Figure 4.13: (top) Histogram for $n = 20$ (showing approximate probability density $p(N_{\text{calls}})$ inferred from all the instances at this size) of the number N_{calls} of problem calls, for the typical instances (green) and the instances from [Crosson et al. \(2014\)](#) (red). (bottom) The same data, but binned and plotted on a logarithmic scale.

log axes, the joint distribution of the number N_{calls} of MixBnB problem calls and QW success probability $\bar{P}(0, 100)$ at $n = 15$. The bottom heatmap shows, also on log-log axes, the joint distribution of N_{calls} and AQC duration $T_{0.99}$, plotted using the 9868 instances for which $T_{0.99}$ was successfully calculated) at $n = 15$. In both cases, some correlation is apparent, but it is difficult to see how well-correlated they are. Turning to numerical measures of correlation, it turns out that the quantities are moderately well-correlated; for example the Spearman's rank correlation coefficient is ≈ -0.47 and ≈ 0.52 for the top and bottom plots respectively. These results, along with the correlation between QW and AQC shown in Fig. 4.11, suggest that typically the same problem features make a problem instance easier or harder for all three methods. Further work is required to determine what these features are, and to find features which only make an instance hard for one of the methods (that the correlations are not perfect suggests there will be some).

Finally, to see how well the actual state-of-the-art MIXSAT algorithm scales on MAX2SAT, I have generated 10,000 instances at much large sizes $50 \leq n \leq 160$ (at these sizes, it was no longer possible to ensure that each instance has a unique solution). For each instance, I have run the actual MIXSAT implementation of Wang and Kolter (2018) and recorded the runtime t_{run} (it was not possible to extract N_{calls} from this implementation). Fig. 4.15 shows a log-linear plot of the mean runtime $\langle t_{\text{run}} \rangle$ of at each problem size n . A linear model has been fitted to $n \geq 125$, where it fits well (the line is solid in this region, and dotted at lower n), indicating an exponential (in n) increase. The gradient of the line implies a scaling of

$$\langle t_{\text{run}} \rangle = O(N^{0.1143 \pm 0.0007}). \quad (4.7)$$

As expected, this is much better performance than the MixBnB algorithm, and is also much better performance than the QW and AQC quantum algorithms.

4.5 Chapter summary

In this chapter, I have demonstrated that the results found for computation by continuous-time quantum walk (QW) on Sherrington-Kirkpatrick (SK) spin-glass problems in Chapter 3 generalize to combinatorial optimization problems other than SK by performing similar simu-

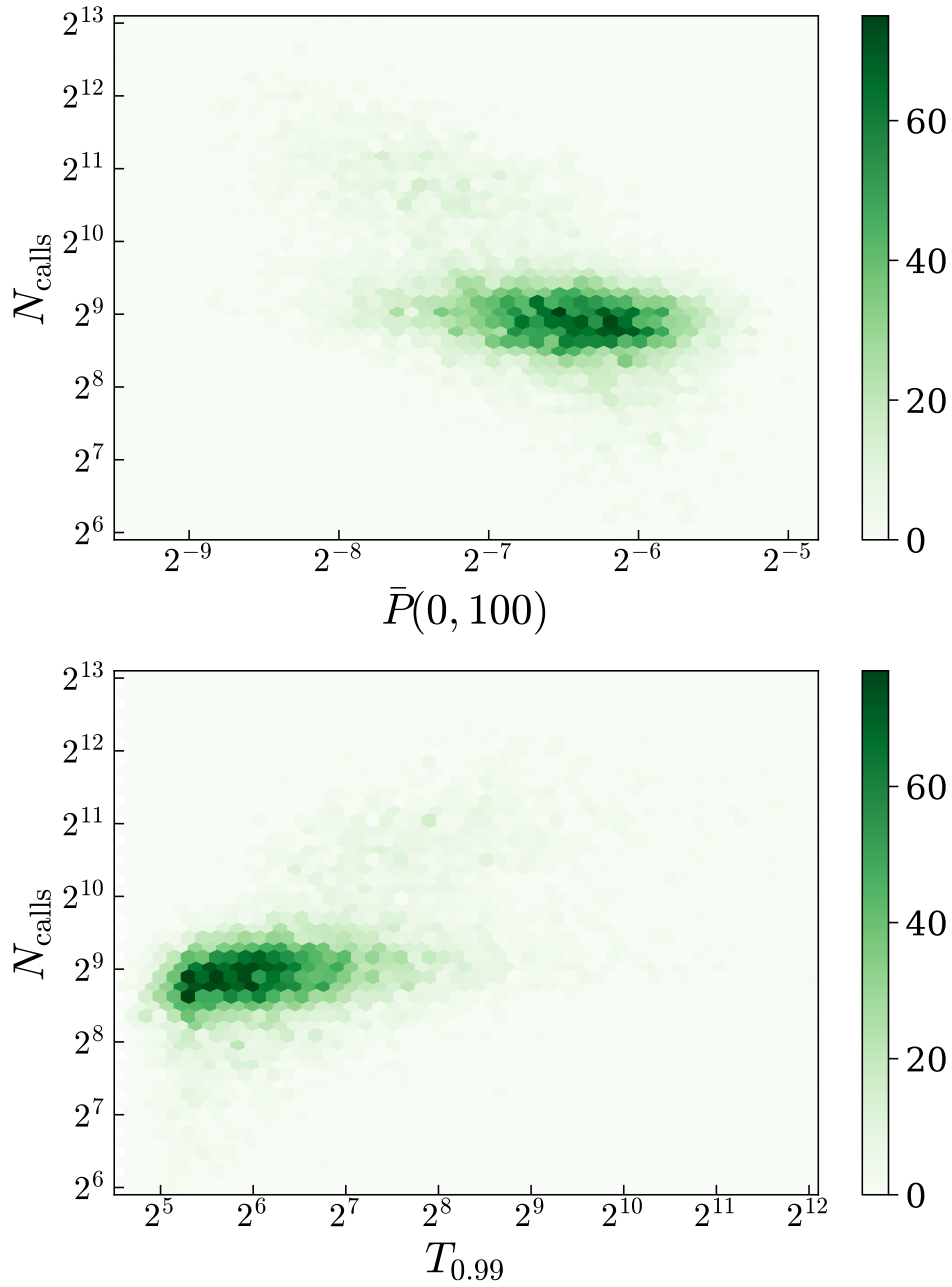


Figure 4.14: Heatmaps (on log-log axes) showing the joint distribution of the number N_{calls} of problem calls and QW success probability $\bar{P}(0, 100)$ (top) and AQC duration $T_{0.99}$ (bottom, using the 9868 instances for which $T_{0.99}$ was successfully calculated) at $n = 15$. In both cases, some correlation is apparent, and numerical measures of correlation show that the quantities are moderately well-correlated; for example the Spearman's rank correlation coefficient is ≈ -0.47 and ≈ 0.52 for the top and bottom plots respectively.

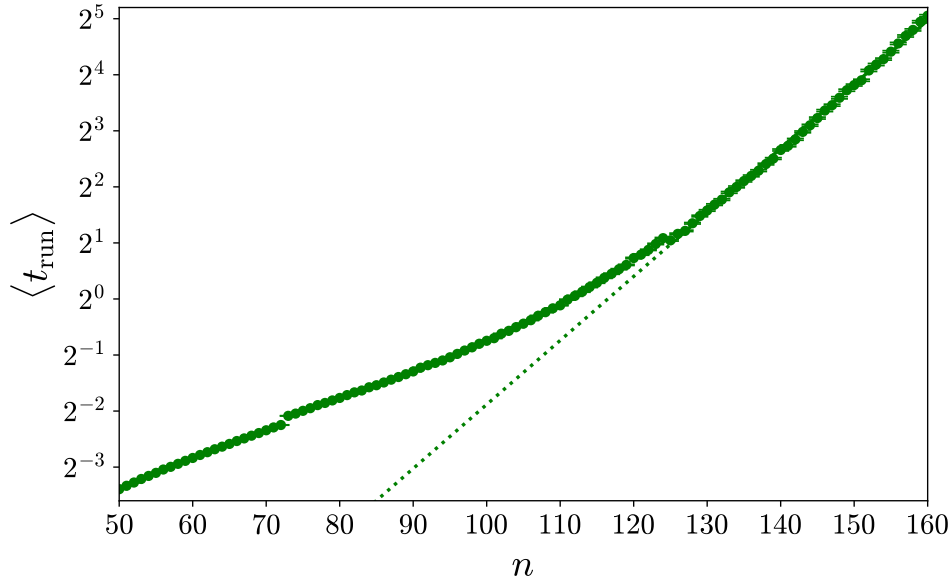


Figure 4.15: Log-linear plot of the mean runtime $\langle t_{\text{run}} \rangle$ of at each problem size n for a run of the MIXSAT algorithm from Wang and Kolter (2018). The means are taken over 10,000 newly generated instances at each size (at these sizes, it was no longer possible to ensure that each instance has a unique solution). A line of best-fit has been fitted to $n \geq 125$, where it fits well (the line is solid in this region, and dotted at lower n), indicating an exponential (in n) increase. The gradient of the line of the best fit implies a scaling of $\langle t_{\text{run}} \rangle = O(N^{0.1143 \pm 0.0007})$.

lations on the Maximum 2-Satisfiability (MAX2SAT) problem. While the behaviour of QWs is qualitatively very similar between the two problems, I have also pointed out the quantitative differences where they have arisen. The behaviour is very similar because the problems result in similar problem Hamiltonians when mapped to the Ising model (and a transverse field is used as the driver Hamiltonian in both cases). While the two problems have different kinds of coupling and field strengths, normally-distributed for SK and integer valued for MAX2SAT, Fig. 3.9 and Fig. 4.6 show that the correlation between Hamming distance and energy difference in the spectrum are important in the same way in both cases.

I also moved beyond the QW setting to numerically test the performance of adiabatic quantum computing (AQC) on MAX2SAT. While it was difficult to find conclusive results for AQC, it appears that AQC significantly outperforms QW on MAX2SAT, achieving better runtime scaling. Furthermore, I compared QW and AQC on MAX2SAT at a much more fine-grained level, including on a per-instance basis, and found that hardness for each method is quite well-correlated. However, the correlation is not perfect, suggesting that there will be classes of instances for which one method is much more suitable than the other.

I also considered a state-of-the-art classical algorithm for MAX2SAT, known as MIXSAT, including implementing my own solver that incorporates many of its features in order to assess the performance on the same instances I used for QW and AQC. The original MIXSAT algorithm outperforms both QW and AQC considerably, and my own related solver showed that hardness for both QW and AQC is moderately well-correlated with classical hardness - although the correlations were far from perfect, suggesting there will be classes of instances for which the quantum methods could be more suitable than the classical algorithms.

Throughout, I have compared my typical MAX2SAT instances with the $n = 20$ -variable MAX2SAT instances from [Crosson et al. \(2014\)](#) that were data-mined for their hardness for quantum annealing (QA). These instances have been consistently harder than the typical instances for QW and the classical algorithm (I have not been able to compare the AQC performance), suggesting that the same features tend to make an instance hard for all three approaches. That said, the instances from [Crosson et al. \(2014\)](#) do not appear to be as *extremely* hard for QW and classical as they were for QA.

Further work is required to back-up many of the results in this chapter. In particular, it would be useful to see whether the AQC scaling results from Fig. 4.8 and Fig. 4.9 continue beyond the sizes simulated; ideally, that would take the form of an analytic scaling result, but simply simulating at larger sizes would be useful, although computationally demanding. This would also enable comparison of AQC on typical instances with the hard-for-QA instances from [Crosson et al. \(2014\)](#). In addition, while the comparison of classical hardness with AQC/QW hardness in section 4.4 is illuminating, it would be useful to confirm that my solver implementation really finds the same instances hard as the full MIXSAT implementation. This could be done, for example, by re-implementing the whole of MIXSAT in a way more amenable to extracting a measure of hardness comparable to N_{calls} .

The results in this chapter have shown that classical algorithms can sometimes be very powerful, highlighting the need to move beyond simple approaches like QW and AQC to find competitive continuous-time quantum computing (CTQC) algorithms. That AQC seems to outperform QW considerably indicates that time-dependence in the schedule can be very useful, but fully-fledged AQC often requires long coherence times that can be impractical on near-term devices. As such, Chapter 5 will look for advantages in the rapid-quench regime in QA: time-dependent protocols (although the time-independent QWs can be seen as one limit of them) that proceed

much faster than AQC. Through numerical and analytical work, I will discuss how to design effective rapid quench protocols.

Chapter 5

Rapid quenches in quantum annealing

5.1 Introduction

The continuous-time quantum walks (QWs) that were the focus of Chapter 3 can be thought of as a time-independent limit of quantum annealing (QA). In this chapter, I will move beyond this limit, into time-dependent QA schedules. The initial framework that I will describe will include the QW and adiabatic quantum computing (AQC) (see section 2.5) as limits, but the chapter will focus on time-dependent schedules that proceed much faster than can be treated by the same analyses used for AQC. Such protocols can be referred to as *rapid quenches*, a term I take to include the QW (as in [Hastings \(2019\)](#)).

Rapid quenches can operate in an open-systems context, but this chapter will discuss the coherent regime of operation, for which the effects of thermal dissipation and decoherence can be neglected. Such a regime could be experimentally reached either by reducing noise, implementing quantum error correction ([Jordan et al., 2006](#); [Sarovar and Milburn, 2005](#); [Young et al., 2013](#); [Sarovar and Young, 2013](#); [Freeman et al., 2018](#); [Atalaya et al., 2020](#); [Pudenz et al., 2014](#); [Bookatz et al., 2015](#); [Lidar, 2008](#)), or quenching on a timescale that is much faster than the decoherence time. A complementary approach to reduce noise is to implement dynamics which reduce or eliminate the interaction between the system and its environment through quantum interference effects, known as dynamical decoupling ([Lidar, 2008](#); [Paz-Silva et al., 2012](#); [Quiroz and Lidar, 2012](#)). Although current superconducting QA hardware operates in a dissipative regime ([Dickson et al., 2013](#)), QA has been implemented in atomic settings where coherence is

easier to maintain than in superconducting circuits (Bernien et al., 2017b), and progress has been made to reduce noise in superconducting circuit settings (D-Wave Systems Inc., 2019a). Simple kinds of error correction have also been implemented experimentally for QA (Pudenz et al., 2014, 2015; Vinci et al., 2015, 2016; Vinci and Lidar, 2018), and there have been efforts to sidestep limitations on quench rates in superconducting systems (Lanting, 2018).

In the fully coherent regime, the dynamics are straightforward to model theoretically, since they can be described by a set of qubits (two state quantum systems) under the action of a Hamiltonian, evolving according to the Schrödinger equation. Conventionally, the Hamiltonian for this evolution is written as the sum of a *problem* Hamiltonian $\hat{H}_P$, which is diagonal in the computational basis, and encodes the classical problem being solved, and a *driver* Hamiltonian $\hat{H}_{\text{drive}}$ which implements quantum dynamics to explore the solution space. In this chapter, two equivalent forms for the total Hamiltonian are used. First,

$$\hat{H}_{\text{AB}}(t) = A(t) \hat{H}_{\text{drive}} + B(t) \hat{H}_P, \quad (5.1)$$

where $A(t)$ and $B(t)$ are positive, time-dependent control functions. The total Hamiltonian $\hat{H}_{\text{AB}}(t)$ typically takes the form of the transverse Ising Hamiltonian $\hat{H}_{\text{TI}}(t)$ defined in Eq. (2.7), but is left in its more general form for the purposes of this discussion. While Eq. (5.1) is a perfectly valid parametrization, typically the crucial feature is what happens to the ratio of driver to problem strength $A(t)/B(t)$ as the algorithm progresses. As such, a useful alternative parametrization of the Hamiltonian, equivalent to $\hat{H}_{\text{AB}}(t)$ up to an overall (time-dependent) scaling factor $B(t)$, is

$$\hat{H}_{\Gamma}(t) = \Gamma(t) H_{\text{drive}} + H_P, \quad (5.2)$$

where there is a single control function $\Gamma(t) > 0$ for the ratio $A(t)/B(t)$. Since Eq. (5.1) and Eq. (5.2) are equivalent, up to a rescaling of the time parameter, results for one form of Hamiltonian will generalize to results for the other. Both forms will be used in this chapter, with the most convenient form chosen for the specific problem or example.

Hamiltonians of the form Eqs. (5.1) and (5.2), which begin with $A(t) > 0$ and $B(t) = 0$ and end with $A(t) = 0$ and $B(t) > 0$, or equivalently, begin with $\Gamma(t) \gg 1$ and end with $\Gamma(t) = 0$, are used for most types of continuous-time quantum computing (CTQC). The term *rapid quench* will refer to protocols that are run on a much shorter timescale than required for AQC.

The QW, described in section 2.4 and applied to spin glass optimization problems in Chapter 3, is the simplest form of CTQC in the coherent regime. In terms of the general Hamiltonian in Eq. (5.2), for QW the control functions are time-independent and set so that $\Gamma(t) = \gamma$ where γ is a constant hopping rate. This can be viewed as the limit of an infinitely fast quench, in which $B(0)$ jumps from zero to $A(0)/\gamma$ at $t = 0$ and $A(t_f)$ drops to zero at the final time t_f . The other pure state CTQC which is commonly considered is AQC, described in section 2.5, for which the control functions $A(t)$ and $B(t)$ are varied slowly from $A(0) = 1$ and $B(0) = 0$ to $A(t_f) = 0$ and $B(t_f) = 1$. As shown in subsection 2.5.1, the adiabatic theorem of quantum mechanics means that AQC achieves a success probability (probability of finding the ground state of the problem Hamiltonian $\hat{H}_P$) which approaches 1 as $t_f \rightarrow \infty$.

The fully coherent regime has provable quantum speedups in the case of both AQC and QW. As explained in subsections 2.4.5 and 2.5.2, the unstructured search problem can yield the same speedup in the AQC (Roland and Cerf, 2002) and QW (Childs and Goldstone, 2004) settings as the gate based counterpart via Grover’s algorithm. It is possible to interpolate between these two techniques while preserving the speedup (Morley et al., 2019). For problems which are closer to real world optimisation, theoretical studies have mostly focused on AQC (Albash and Lidar, 2018a), likely because the adiabatic theorem provides a general way to show that such algorithms could in principle succeed with high probability. While theoretically tractable, the adiabatic regime is difficult to reach experimentally, and contains some counter-intuitive effects in the deep adiabatic regime (Wiebe and Babcock, 2012; Campos Venuti and Lidar, 2018; Passos et al., 2020). Assuming $P \neq NP$, solving NP-hard problems adiabatically will obtain at best a polynomial speed up. AQC requires the system to remain coherent throughout, meaning that the exponentially long runtimes for NP-hard problems require an exponentially long coherence time, which is experimentally challenging for near-term quantum computing (QC).

On the other hand, if a CTQC protocol has a short enough runtime to fit within a constant or mildly scaling coherence time, the achievable success probabilities will necessarily be exponentially low, requiring exponentially many repeats to succeed with high probability. This approach is a valid one for problems other than search; the QW approach to solving spin-glass optimization problems presented in Chapter 3 is an example of this. It has also been numerically demonstrated that rapid quenches can be superior to long quenches for AQC-like algorithms (Crosson et al., 2014). Recently, Crosson and Lidar (2020) made an important contribution

to the theory of QA outside of the adiabatic limit by introducing *diabatic* quantum annealing (DQA), which formalizes ideas described in [Muthukrishnan et al. \(2016\)](#); [Katsuda and Nishimori \(2013\)](#). DQA relies on a generalisation of the adiabatic theorem from [Jansen et al. \(2007\)](#), and describes a class of QA algorithms in which amplitude is restricted to a low-energy part of the Hamiltonian spectrum.

Finally, for algorithms for NP-hard problems that achieve a high success probability with a single shot, achieving even a polynomial speedup typically requires setting, with exponential precision, the control functions to values which lead to exponentially small gaps in the Hamiltonian spectrum. This was shown to be necessary for unstructured search in [Childs et al. \(2002\)](#); [Roland and Cerf \(2002\)](#); [Morley et al. \(2019\)](#) and for the random energy model in Chapter 3. This requirement is problematic, as there are no general methods for determining where these gaps occur and because such precise control settings can be difficult to achieve in real hardware. Recent work by [Chakraborty et al. \(2018\)](#) demonstrates that some of the fine tuning requirements in unstructured search can be avoided by formulating the Hamiltonian differently, but it is unclear whether this approach would extend to the random energy model.

Methods which can succeed with limited coherence time will be important for near term QC. In this chapter, mathematical tools will be developed to improve understanding of how computation is achieved in both the rapid quench regime including in QW protocols. These tools are important not only for theoretical understanding of when adiabatic algorithms and rapid quenches will be effective, but also for choosing parameters for the Hamiltonians used. While some theoretical arguments, such as those presented in Chapter 3 (and also in [Hastings \(2019\)](#)), can be made for why QW with short runtimes seems to perform well, a theoretical understanding of rapid quenches with time-dependent Hamiltonians far from the adiabatic regime is essentially missing.

It has been recently shown numerically ([Brady et al., 2020](#)) that the optimal protocol for solving problems often involves an annealing step where the driver and problem Hamiltonians are active simultaneously, as opposed to bang-bang controls that alternate between the driver and problem Hamiltonians. Previous theoretical work based on the Pontryagin's minimum principle showed that optimal control patterns would always take the bang-bang form ([Yang et al., 2017](#)), but that these controls would sometimes require un-physically switching between the driver and problem an infinite number of times in a finite time span. The work done in

Brady et al. (2020) restricted to cases with a finite number of “bangs” and found that, in this more realistic setting, protocols involving annealing may be superior.

This chapter is structured as follows: section 5.2 will show some numerical examples to illustrate the performance gains that can be obtained from well-chosen rapid quenches in QA. This provides motivation to understand why rapid quenches work, and how to exploit the effects more systematically. Next, section 5.3 will provide a general set of conditions (essentially requiring that quenches be monotonic) under which rapid quenches will preferentially seek out high quality solutions. In section 5.4, this analysis is augmented by studying the transitions between different computational basis states, from which the level of dynamics that will occur can be deduced. These tools are then applied to different problem settings, including a discussion of the conditions for general optimisation problems to yield a significant level of dynamics. Then, in section 5.5, it is shown that the tools that have been developed here can be used to construct heuristics for setting the parameters for QWs and rapid quenches. Finally, the chapter is summarised in 5.6.

5.2 Rapid quench examples

In this section, three illustrative examples showing the power of rapid quenches to solve problems will be presented to motivate the theoretical tools that will be discussed later. For simplicity and concreteness, this section will focus on *monotonic* quenches; that is, quenches for which the control parameter satisfies $\Gamma(t') \leq \Gamma(t) \forall t' > t$.

5.2.1 Two-stage quantum walk

The two-stage QW is a minimal modification to the time-independent QW. It consists of two time-independent stages of evolution separated by an infinitely fast quench. I refer to it as a *two-stage* QW because each time-independent stage is effectively a QW. As before, a simple transverse field driver Hamiltonian defined in Eq. (2.6) will be used. However, the usual

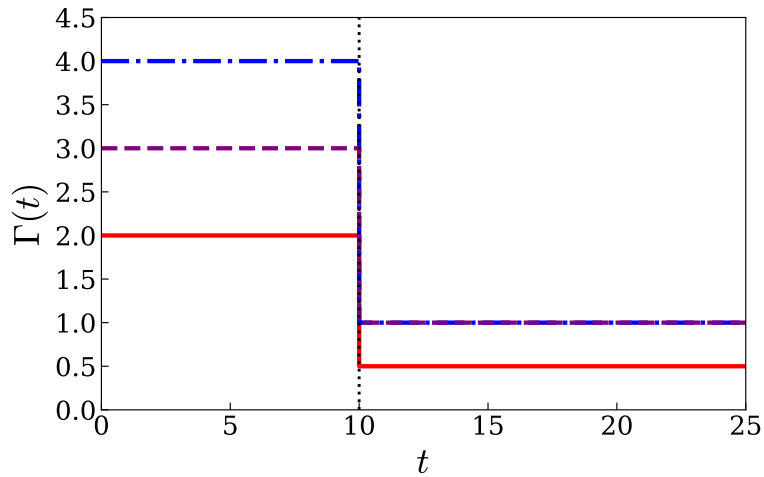


Figure 5.1: The annealing schedule for the two stage QWs in Fig. 5.2 (red, solid line), Fig. 5.3 (blue, dot dashed line), and Fig. 5.4 (purple, dashed line). In all cases, the step occurs at $t = 10$ (dotted line).

constant control function ($\Gamma(t) = \gamma$) is replaced with the time-dependent schedule

$$\Gamma(t) = \begin{cases} \gamma_1 & 0 < t < t_1 \\ \gamma_2 & t_1 < t < (t_1 + t_2) \end{cases}, \quad (5.3)$$

which consists of two consecutive evolution stages with two different time independent Hamiltonians. Although each of these stages is effectively a QW, the second stage uses non-standard starting conditions, as its initial state is the final state of the first stage. The standard initial state is the equal superposition of all basis states, $|\psi_0\rangle = 2^{-n/2} \sum_j |j\rangle$, chosen because it is the ground state of the driver Hamiltonian, and also represents complete ignorance of which basis state is the solution to the problem. The schedules used for the two stage QW are shown in Fig. 5.1 for each of the three examples that will be discussed below.

As discussed in Chapter 3, a QW can be understood from an energetic perspective according to a mechanism referred to there as the *energy conservation mechanism*. Being time-independent, QWs conserve the total energy of the system. To show the effect of changing the hopping rate γ part way through the walk, thus disrupting the energy conservation, consider the example of a simple two qubit problem Hamiltonian

$$\hat{H}_{2Q} = -\hat{Z}_1 \hat{Z}_2 - \frac{1}{2} \hat{Z}_1, \quad (5.4)$$

where $\hat{Z}_j$ is the Pauli $\hat{Z}$ operator acting on qubit j . The initial state at $t = 0$ is $|\psi_0\rangle =$

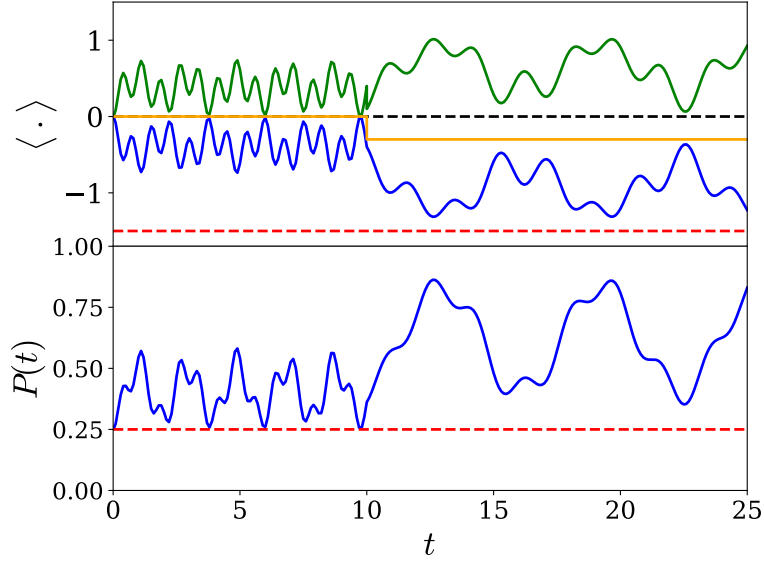


Figure 5.2: Two stage QW using Hamiltonian in Eq. (5.4) with $\gamma_1 = 2$ and $\gamma_2 = \frac{1}{2}$. The instantaneous quench occurs at time $t_1 = 10$. Top: energy expectation values $E_\Gamma = \Gamma(t) \langle \hat{H}_{\text{drive}} \rangle + \langle \hat{H}_P \rangle$ (gold), $\Gamma(t) \langle \hat{H}_{\text{drive}} \rangle$ (green), $\langle \hat{H}_P \rangle$ (blue). Also shown (black, dashed) is a guide to the eye at 0, and the minimum eigenvalue of $\hat{H}_P$ (red, dashed). Bottom: probability $P(t)$ of being in the ground state of $\hat{H}_P$ at time t (blue), probability of random guessing (red, dashed).

$\frac{1}{2}(|00\rangle + |01\rangle + |10\rangle + |11\rangle)$, the two qubit ground state of the driver Hamiltonian $\hat{H}_{\text{drive}}$ in Eq. (2.6). Let $\langle \hat{H}_P \rangle_{\psi(t)} \equiv \langle \psi(t) | \hat{H}_P | \psi(t) \rangle$ be the instantaneous expectation value of the problem Hamiltonian with respect to the state $\psi(t)$ at time t . Likewise, let $\langle \hat{H}_{\text{drive}} \rangle_{\psi(t)} \equiv \langle \psi(t) | \hat{H}_{\text{drive}} | \psi(t) \rangle$ be the corresponding quantity for the driver Hamiltonian. Let $E_\Gamma(t) = \Gamma(t) \langle \hat{H}_{\text{drive}} \rangle + \langle \hat{H}_P \rangle$ be the total instantaneous energy expectation value.

Fig. 5.2 (top) shows that the expectation value $\langle \hat{H}_{\text{drive}} \rangle$ for the transverse field is zero initially ($t = 0$). As in subsection 3.3.2, the energy conservation mechanism then decreases the expectation value of the problem Hamiltonian at the expense of increasing the expectation value of the driver Hamiltonian. When the instantaneous quench is performed, the problem Hamiltonian expectation value is unchanged, but the driver Hamiltonian expectation value (and therefore the total energy expectation value $E_\Gamma(t)$) is reduced. As the minimum eigenvalue of $\hat{H}_{\text{drive}}$ is zero, the total energy expectation value $E_\Gamma(t)$ acts as an effective upper bound on $\langle \hat{H}_P \rangle_{\psi(t)}$. The net effect is that, even if all of the energy stored in the transverse field were returned to the problem Hamiltonian, its expectation value would still be less than it was at the beginning of the algorithm. What actually happens, however, is that the transverse field is able to capture even more of the energy, thereby reducing the problem Hamiltonian expectation value further,

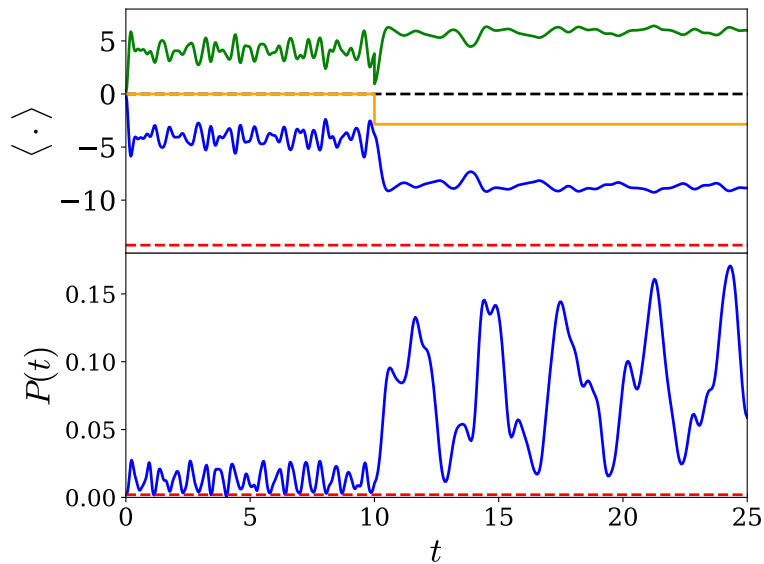


Figure 5.3: Two stage QW on a 9 qubit SK spin glass, ID code **ovcjhw bhtcpcvwicox-pdpvjzqojril** from the public repository in [Chancellor et al. \(2019\)](#) with $\gamma_1 = 4$ and $\gamma_2 = 1$. The instantaneous quench occurs at time $t_1 = 10$, (vertical dotted line). Top: energy expectations $E_\Gamma = \Gamma(t) \langle \hat{H}_{\text{drive}} \rangle_{\psi(t)} + \langle \hat{H}_P \rangle_{\psi(t)}$ (gold), $\Gamma(t) \langle \hat{H}_{\text{drive}} \rangle_{\psi(t)}$ (green), $\langle \hat{H}_P \rangle_{\psi(t)}$ (blue). Also shown (black, dashed) is a guide to the eye at 0, and the minimum eigenvalue of $\hat{H}_P$ (red, dashed). Bottom: probability $P(t)$ of being in the ground state of $\hat{H}_P$ at time t (blue).

and increasing the average probability of finding the ground state, as in Fig. 5.2 (bottom).

A more realistic problem is the Sherrington-Kirkpatrick (SK) spin-glass ([Sherrington and Kirkpatrick, 1975](#)) ground-state problem investigated in Chapter 3, with the problem Hamiltonian defined in Eq. (2.8). Fig. 5.3 shows a two-stage QW performed on a nine qubit SK Hamiltonian¹ from the public repository ([Chancellor et al., 2019](#)). In the setting of this larger problem, the fluctuations after each stage of the QW are smaller relative to the dynamical range than in the two qubit case, a very early sign of the approach to the thermodynamic limit. Apart from this, the behaviour is qualitatively similar to the two qubit toy model $\hat{H}_{2Q}$ of Eq. (5.4) shown in Fig. 5.2, and produces a significant increase in the probability of finding the ground state.

5.2.2 Biased two-stage quantum walk

In this subsection, a biased driver Hamiltonian is introduced, similar to the ones used in [Duan et al. \(2013\)](#); [Graß \(2019\)](#). The biased driver Hamiltonian used here is formulated slightly

¹I chose instance **ovcjhw bhtcpcvwicox-pdpvjzqojril** and used it throughout this chapter for all single SK problem examples in this chapter.

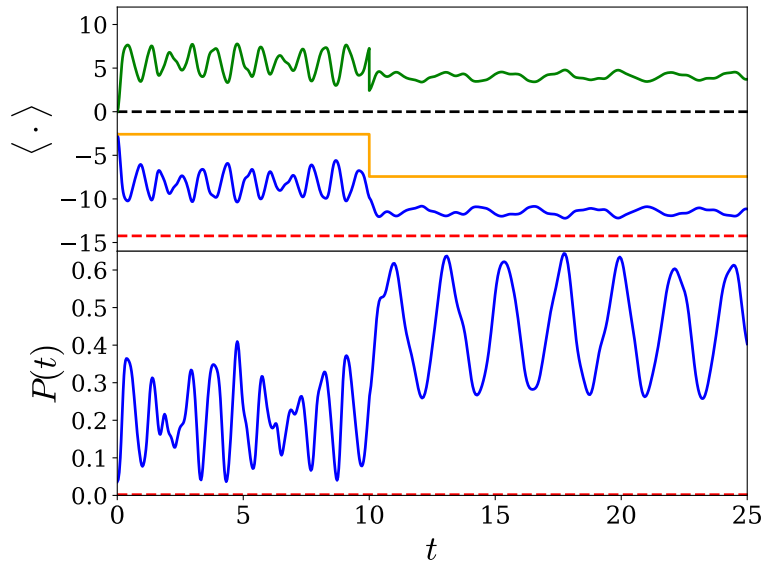


Figure 5.4: Biased two-stage QW on a 9 qubit SK spin glass, ID code **ovcjhwbtcpvcwicox-pdpvjzqojril** from the public repository in [Chancellor et al. \(2019\)](#) with $\gamma_1 = 3$ and $\gamma_2 = 1$, using a biased driver (Eq. (5.5)), biased towards the optimal solution of $\hat{H}_P$ using $\theta = \frac{\pi}{8}$. The instantaneous quench occurs at time $t_1 = 10$. Top: energy expectations $E_\Gamma = \Gamma(t) \langle \hat{H}_{\text{drive}} \rangle_{\psi(t)} + \langle \hat{H}_P \rangle_{\psi(t)}$ (gold), $\Gamma \langle \hat{H}_{\text{drive}} \rangle_{\psi(t)}$ (green), $\langle \hat{H}_P \rangle_{\psi(t)}$ (blue). Also shown (black, dashed) is a guide to the eye at 0, and the minimum eigenvalue of $\hat{H}_P$ (red, dashed). Bottom: probability $P(t)$ of being in the ground state of $\hat{H}_P$ at time t . (blue).

differently as

$$\hat{H}_{\text{bias}}(g, \theta) = n \mathbb{1} - \sum_{i=1}^n \left(\cos(\theta) \hat{X}_i + g_i \sin(\theta) \hat{Z}_i \right), \quad (5.5)$$

where $g_i \in \{-1, 1\}$ is a candidate (or guess) solution, and takes the value 1 if the i th bit of the guess solution is 0, and -1 if it is 1. The certainty of the guess is parametrized by $0 \leq \theta \leq \frac{\pi}{2}$; if $\theta = 0$ the guess goes unused and the driver reduces to a transverse field of Eq. (2.6). In the other extreme, if $\theta = \frac{\pi}{2}$, then the ground state of $\hat{H}_{\text{bias}}(g, \theta = \frac{\pi}{2})$ is the candidate solution and there are no dynamics. The ground state of the biased driver Hamiltonian has zero energy for all allowed values of θ and g , and is a tensor product of spin states which are each anti-parallel to the fields in Eq. (5.5); this state is used as the initial state. For simplicity, in this example biasing is only performed toward the most optimal solution (i.e., correct guesses), and the same nine-qubit SK spin glass as in the previous subsection is used.

As Fig. 5.4 shows, the effect of biasing toward the optimal solution is to lower the initial values of E_Γ and $\langle \hat{H}_P \rangle_{\psi(t)}$; biasing toward a well chosen guess effectively gives the algorithm a ‘head start’ with respect to energy expectation values. This is qualitatively similar to what happens at the beginning of the second stage of the two stage QW, except that the driver

energy $\langle \hat{H}_{\text{bias}}(g, \theta) \rangle_{\psi(t)}$ starts at exactly zero, rather than having some initial energy left over from a previous stage. The bias improves the initial stage success probability by a factor of ten compared with the unbiased QW in Fig. 5.3, while the second stage again provides a (further) factor of three improvement. This biased two-stage QW example provides proof-of-concept that the rapid quench mechanism described here can be leveraged on top of a biased search.

5.2.3 Pre-annealed quantum walk

The final example is again in two stages, but this time the first stage is a quantum anneal, and the second stage is a QW that starts from the point where the anneal stops. The motivating intuition is that the initial time-dependent annealing stage will prepare an initial state for the QW that has a lower average problem energy $\langle \hat{H}_P \rangle_{\psi(t)}$ than the usual uniform superposition state. If performed too slowly, such a quench will put the system into its instantaneous ground state according to the adiabatic theorem, and there will be no QW dynamics. If performed too rapidly, the state will not evolve much during the anneal stage and the resulting QW will be similar to one without a pre-annealing stage. However, if the anneal is performed at an intermediate rate, it leads to significant QW dynamics, starting from a lower problem Hamiltonian expectation value $\langle \hat{H}_P \rangle_{\psi(t)}$.

Using the $\hat{H}_{AB}$ parametrization defined in Eq. (5.1), the protocol consists of pre-annealing with a quadratic schedule for a time t_1 , followed by a time-independent QW; specifically, the schedule is defined as

$$A(t) = \begin{cases} \gamma[1 + (\frac{t}{t_1} - 1)^2] & 0 \leq t \leq t_1 \\ \gamma & t_1 < t < (t_1 + t_2) \end{cases}, \quad (5.6)$$

$$B(t) = \begin{cases} [1 - (\frac{t}{t_1} - 1)^2] & 0 \leq t \leq t_1 \\ 1 & t_1 < t < (t_1 + t_2) \end{cases} \quad (5.7)$$

which is plotted in Fig. 5.5.

Using the same nine-qubit SK problem as before, with its optimal γ value of approximately 1.004, the results for three different values of t_1 are shown in Fig. 5.6. Pre-annealing both decreases the average problem expectation value $\langle \hat{H}_P \rangle_{\psi(t)}$ and increases the success probability, but causes the peak values to be reached more slowly. In the longest pre-anneal with $t_1 = 4$, the

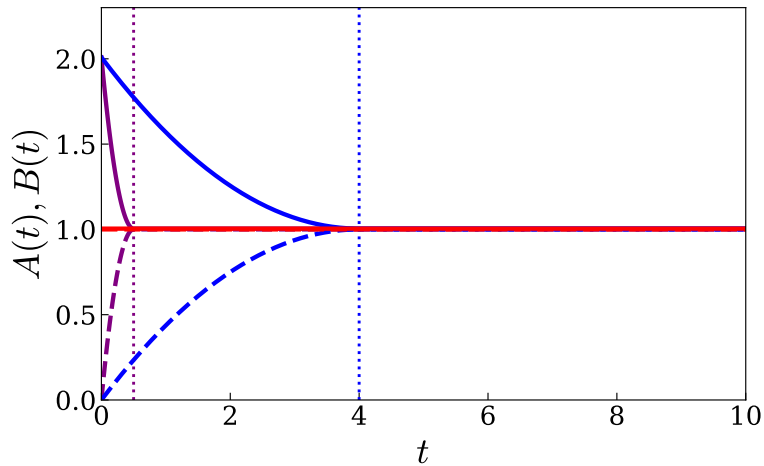


Figure 5.5: Schedule $A(t)$ (solid) and $B(t)$ (dashed) of a pre-annealed QW using $\gamma \approx 1.004$ and $t_1 = 4$ (blue, vertical dotted line), $t_1 = 0.5$ (magenta, vertical dotted line), and $t_1 = 0$, (red, pure QW).

success probability undergoes small amplitude, approximately sinusoidal, oscillations suggesting that the dynamics are dominated by a two level subspace. For $t_1 = 0.5$ and $t_1 = 0$, the oscillations are less structured, indicating that more than two energy levels are playing a non-trivial role in the dynamics.

The increases in the success probability seen in Fig. 5.6 are relatively modest for this example. To determine the typical improvement in success probability due to pre-annealing, pre-annealed QWs were performed on 1,000 random instances (chosen independently for each data point) from the set of 10,000 SK instances at each size from $n = 5$ to $n = 11$ from [Chancellor et al. \(2019\)](#) to compare the success probability averaged over the QW stage using 20 different linearly spaced pre-annealing times up to $t_1 = 4$. In Fig. 5.7 (top), it can be seen that the success probability increases with pre-anneal time, up to a plateau, and the relative effect of pre-annealing becomes larger as n increases. To quantify this effect, the scaling exponent at each pre-annealing time has been calculated by fitting a linear model on log-linear axes. The scaling exponent κ is defined such that the success probability $p_{\text{success}} \propto 2^{\kappa n}$. The fitted values of κ are plotted in Fig. 5.7 (bottom). As the inset of Fig. 5.7 (bottom) shows, the success probability is modelled well by a simple exponential function, as with the QW success probability results in Chapter 3. It can be seen that pre-annealing significantly improves the scaling from $\kappa = -0.405 \pm 0.03$ for a pure QW, in agreement with the findings in Chapter 3, to a maximum of $\kappa \approx -0.276 \pm 0.02$. It is an open question whether this scaling will continue to problem sizes which are of practical interest, but the lack of visible finite size effects in

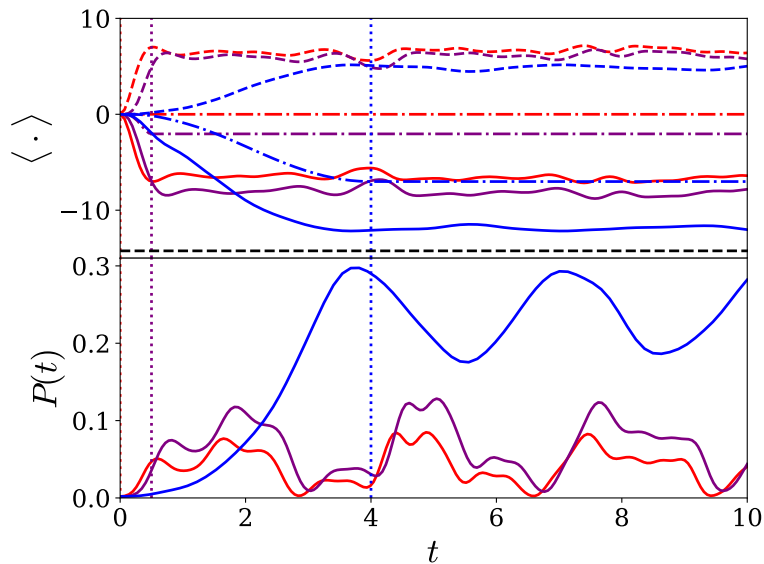


Figure 5.6: Pre-anneal performed on a nine qubit SK spin glass, ID code **ovcjhw-bhtpcvwicoxpdvpvjzqojril** from [Chancellor et al. \(2019\)](#), for pre-anneal times $t_1 = 4$ (blue), $t_1 = 0.5$ (magenta), $t_1 = 0$ (red), i.e., pure QW. Dotted lines show when the pre-anneal ends. Top: Expectation values $E_{\Gamma}(t) = \frac{A(t)}{B(t)} \langle \hat{H}_{\text{drive}} \rangle_{\psi(t)} + \langle \hat{H}_P \rangle_{\psi(t)}$ (dot-dashed), $\langle \hat{H}_P \rangle_{\psi(t)}$ (solid), $\frac{A(t)}{B(t)} \langle \hat{H}_{\text{drive}} \rangle_{\psi(t)}$ (dashed, colour). The black dashed line indicates the minimum eigenvalue of $\hat{H}_P$. Bottom: success probability $P(t)$ to be in the lowest eigenstate of $\hat{H}_P$ at time t .

Fig. 5.7 suggests that it might. Since very fast quenches can be experimentally challenging to implement (though methods are being explored ([Lanting, 2018](#))), determining the effects of quenching at a finite rate is of practical importance. The results here show that such quenches are potentially a better strategy than trying to speed up or slow down to approach QW or AQC extremes.

5.3 Energy redistribution mechanism

In all the examples in section 5.2, it can be seen that the total energy expectation value $E_{\Gamma}(t)$ never increases during a rapid quench, and that $E_{\Gamma}(t)$ serves as an upper-bound to the problem expectation value $\langle \hat{H}_P \rangle_{\psi(t)}$, assuming that the groundstate of $\hat{H}_{\text{drive}}$ is arranged to be at zero energy (the identity term in Eq. (2.6) ensures this). In this section, these observations are formalised into a mechanism that I refer to as the *energy redistribution mechanism*. The analysis extends the energy conservation mechanism described in subsection 3.3.2 to quenches where the Hamiltonian is not time invariant, and therefore total energy is not conserved.

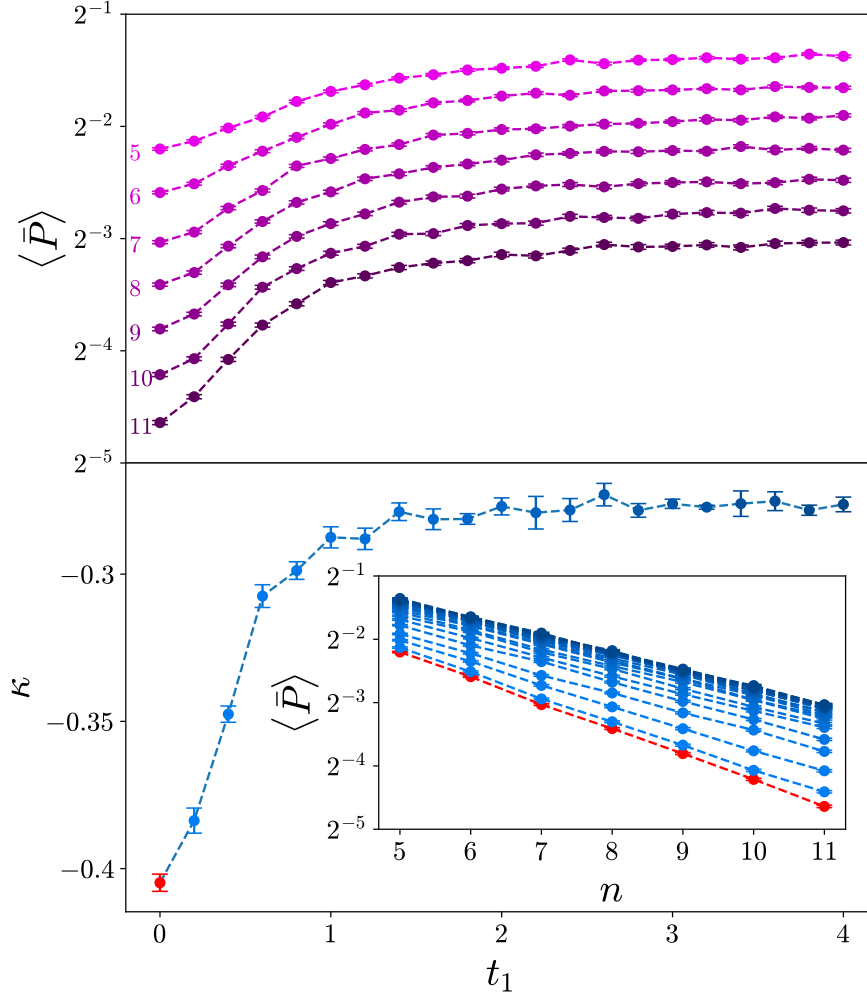


Figure 5.7: Top: success probability $\langle \bar{P} \rangle$ for $n = 5$ to $n = 11$ for 21 different linearly spaced pre-anneal times from $t_1 = 0$ to $t_1 = 4$, darker magenta colour indicates higher n . All data points are averaged over 1,000 random instances (chosen independently for each data point) from the set of 10,000 SK instances at each size from [Chancellor et al. \(2019\)](#). Bottom: Scaling exponent κ for a model where $p_{\text{success}} \propto 2^{\kappa n}$ extracted from the linear fit on log-linear axes for different pre-annealing times in the inset. Inset: Scaling of success probability versus n , for the same t_1 values, with $t_1 = 0$ in red, and $t_1 = 4$ in dark blue (same colour coding as the bottom main figure).

Consider a closed system QA schedule on a system with a Hamiltonian $H(t)$ defined by Eq. (5.2):

$$\hat{H}(t) = \Gamma(t) \hat{H}_{\text{drive}} + \hat{H}_P.$$

I will show that (for duration $t_f \geq 0$) the energy expectation value with respect to the problem Hamiltonian at the end is never higher than at the initial time $t = 0$,

$$\left\langle \psi(t_f) \mid \hat{H}_P \mid \psi(t_f) \right\rangle \leq \left\langle \psi(0) \mid \hat{H}_P \mid \psi(0) \right\rangle, \quad (5.8)$$

provided the following conditions are satisfied:

1. (*initial ground state*) the initial state $|\psi(t = 0)\rangle$ is a ground state of the driver Hamiltonian $\hat{H}_{\text{drive}}$
2. (*positivity*) the control function is non-negative: $\Gamma(t) \geq 0 \forall t$
3. (*monotonicity*) the control function is monotonically decreasing: $\Gamma(t) \geq \Gamma(t') \forall t' > t$

Condition (1) means that the system is initially prepared in the ground state of the driver Hamiltonian. This condition is necessary for AQC, and is also standard for QW. Condition (2) prevents pathological behaviour where the driver spectrum is effectively inverted by taking negative values of the control function $\Gamma(t)$. This condition is satisfied in all traditional AQC and QW settings. Condition (3) is that the quench is *monotonic*; this condition excludes methods such as reverse annealing, both the dissipatively driven form proposed in Chancellor (2017) and implemented on D-Wave devices (D-Wave Systems Inc., 2019b), and the similar coherent method proposed in Perdomo-Ortiz et al. (2011) which is sometimes also referred to as reverse annealing. The biased driver Hamiltonians proposed in Duan et al. (2013); Graß (2019) are compatible with condition (3). The results presented in this chapter do not rely on the adiabatic theorem and the control function $\Gamma(t)$ does *not* need to be a continuous function.

Without loss of generality, the driver Hamiltonian $\hat{H}_{\text{drive}}$ can be chosen such that its ground-state eigenvalue is zero. Let $E_\Gamma(t) = \left\langle \psi(t) \mid \hat{H}_\Gamma(t) \mid \psi(t) \right\rangle$ be the expectation value of the energy at time t . Then, it follows immediately from condition (1) that, at time $t = 0$,

$$E_\Gamma(0) = \left\langle \psi(0) \mid \hat{H}_P \mid \psi(0) \right\rangle. \quad (5.9)$$

Furthermore, it follows from conditions (1) and (2) that, at any later time $t > 0$,

$$E_{\Gamma}(t) \geq \left\langle \psi(t) \mid \hat{H}_P \mid \psi(t) \right\rangle, \quad (5.10)$$

since $\left\langle \psi(t) \mid \hat{H}_{\text{drive}} \mid \psi(t) \right\rangle \geq \left\langle \psi(0) \mid \hat{H}_{\text{drive}} \mid \psi(0) \right\rangle = 0$ can only increase from the ground state initial energy.

For the next part of the proof, I will show that it follows from conditions (2) and (3) that the energy expectation value $E_{\Gamma}(t)$ monotonically decreases with time t ; that is,

$$E_{\Gamma}(t') \leq E_{\Gamma}(t) \forall t, t' : t' > t \quad (5.11)$$

The first step is to introduce a discretized approximation to the evolution as

$$\left| \psi_k^{(q)} \right\rangle = \mathcal{T} \prod_{k'=k}^1 \exp(-i\hat{H}_{\Gamma}(\frac{k't_f}{q})\frac{t_f}{q}) \left| \psi(0) \right\rangle, \quad (5.12)$$

for $1 \leq k \leq q$ and where the symbol $\mathcal{T}$ is added to emphasise that the time order of the product must be preserved, since the Hamiltonians at different times are non-commuting. This discretized approximation becomes exact in the limit $q \rightarrow \infty$. The evolution of a quantum system under the time dependent Hamiltonian given in Eq. (5.1) from time $t = 0$ to time $t = t_f$ from the initial state $|\psi(0)\rangle$ is broken down as follows: The initial state is evolved under the constant Hamiltonian $\hat{H}(\frac{t_f}{q})$ for time $\frac{t_f}{q}$ to produce a state $|\psi_1^{(q)}\rangle$ which then evolves under the constant Hamiltonian $\hat{H}(2\frac{t_f}{q})$ for time $\frac{t_f}{q}$ and so on, until a final state $|\psi_q^{(q)}\rangle$ is reached. Then, in the limit, $|\psi(t_f)\rangle = \lim_{q \rightarrow \infty} |\psi_q^{(q)}\rangle$. This kind of discretization can be thought of as an extension of the Suzuki-Trotter decomposition (Trotter, 1959; Suzuki, 1993) and is therefore sometimes informally referred to as Trotterization. In the same manner, a discretized version of the energy expectation value can be defined as

$$E_{\Gamma,k}^{(q)} = \Gamma\left(\frac{k't_f}{q}\right) \left\langle \psi_k^{(q)} \mid \hat{H}_{\text{drive}} \mid \psi_k^{(q)} \right\rangle + \left\langle \psi_k^{(q)} \mid \hat{H}_P \mid \psi_k^{(q)} \right\rangle, \quad (5.13)$$

Quantum states are only defined up to a constant phase, which is equivalent to choosing an arbitrary ‘zero’ for the energy. Hence, without loss of generality, the choice $\left\langle \psi(0) \mid \hat{H}_{\text{drive}} \mid \psi(0) \right\rangle = 0$ can be made; in other words, the condition *semidefiniteness* is imposed on $\hat{H}_{\text{drive}}$ by defining

its ground state $|\psi(0)\rangle$ to have eigenvalue 0.

During each time-independent evolution step, the energy expectation value $E_{\Gamma,k}^{(q)}$ is conserved. Furthermore, since by definition $\hat{H}_{\text{drive}}$ is positive semidefinite and $\Gamma(\frac{(k+1)t_f}{q}) \leq \Gamma(\frac{kt_f}{q})$, it follows that

$$E_{\Gamma,k+1}^{(q)} \leq E_{\Gamma,k}^{(q)}. \quad (5.14)$$

Repeated application of this inequality results in the more useful inequality

$$E_{\Gamma,q}^{(q)} \leq E_{\Gamma,1}^{(q)}. \quad (5.15)$$

Since $E_{\Gamma,1}^{(q)}$ is the energy during the whole of the first evolution step, it follows that

$$E_{\Gamma,1}^{(q)} = E_{\Gamma}(t = 0). \quad (5.16)$$

Furthermore,

$$\lim_{q \rightarrow \infty} E_{\Gamma,q}^{(q)} = E_{\Gamma}(t_f). \quad (5.17)$$

which means

$$E_{\Gamma}(t_f) \leq E_{\Gamma}(0). \quad (5.18)$$

Since this equation holds for all $t_f > 0$, it means that $E_{\Gamma}(t)$ monotonically decreases with t ; that is, Eq. (5.11) has been proven.

Taken together, the statements in Eqs. (5.9), (5.10) and (5.11) imply

$$\left\langle \psi(t=0) \mid \hat{H}_P \mid \psi(t=0) \right\rangle \geq \left\langle \psi(t=t_f) \mid \hat{H}_P \mid \psi(t=t_f) \right\rangle, \quad (5.19)$$

for final time t_f . In other words, the energy expectation with respect to the problem Hamiltonian can only decrease compared with the initial state. If the energy expectation of the problem Hamiltonian decreases, then the probability of measuring low energy states increases.

I have shown that Eq. (5.19) holds for quenches parameterized with the single control function $\Gamma(t)$, in the form of Eq. (5.2). However, since the control function $\Gamma(t)$ is identified with the ratio $A(t)/B(t)$ of control functions for quenches in the form of Eq. (5.1), the result in Eq. (5.19) follows automatically for quenches in $A(t)$, $B(t)$ form, except for when $B(0) = 0$, when $\Gamma(0)$ is not well-defined.

I will now show that the results can be extended to the case where $B(0) = 0$, with the additional condition that the driver Hamiltonian $\hat{H}_{\text{drive}}$ has a finite gap between its ground and first-excited manifolds (which is automatically true for all Hamiltonians on Hilbert spaces of finite dimension).

In the case that $B(0) = 0$, consider the modified Hamiltonians

$$\hat{H}'_{\text{drive}} = \hat{H}_{\text{drive}} - \frac{\epsilon}{A(0)} \hat{H}_P \quad (5.20)$$

$$\hat{H}'_P = \hat{H}_P \quad (5.21)$$

and the modified control functions

$$A'(t) = A(t) \quad (5.22)$$

$$\begin{aligned} B'(t) &= B(t) + \frac{A(t)}{A(0)} \epsilon \\ &= B(t) \left[1 + \Gamma(t) \frac{\epsilon}{A(0)} \right], \end{aligned} \quad (5.23)$$

where $\epsilon \ll 1$. It can be seen that that total Hamiltonian is unchanged,

$$\begin{aligned} \hat{H}'_{A,B}(t) &\equiv A'(t) \hat{H}'_{\text{drive}} + B'(t) \hat{H}'_P \\ &= A(t) \hat{H}_{\text{drive}} + B(t) \hat{H}_P, \end{aligned} \quad (5.24)$$

but it is also true that

$$B'(0) = \epsilon. \quad (5.25)$$

Define

$$\Gamma'(t) \equiv \frac{A'(t)}{B'(t)} \quad (5.26)$$

$$\Gamma'(t) = \frac{\Gamma(t)}{\left[1 + \Gamma(t) \frac{\epsilon}{A(0)} \right]}. \quad (5.27)$$

It can be immediately seen that $\Gamma'(t)$ is non-negative if $\Gamma(t)$ is non-negative, and so condition

(2) is satisfied. Furthermore,

$$\frac{d\Gamma'(t)}{d\Gamma(t)} = \frac{1}{\left[1 + \Gamma(t)\frac{\epsilon}{A(0)}\right]^2}. \quad (5.28)$$

Thus, $\Gamma'(t)$ is monotonically-decreasing if $\Gamma(t)$ is monotonically decreased, and so condition (3) is satisfied.

If the protocol were to be started in the state $|\psi'_{\text{gs}}\rangle$, a ground-state of $\hat{H}'_{\text{drive}}$, condition (1) would be satisfied and the result would be proven. However, the original protocol under consideration starts in the the state $|\psi(0)\rangle$, ground-state of $\hat{H}_{\text{drive}}$. Applying first order perturbation theory in ϵ to $\hat{H}'_{\text{drive}}$, gives that $\hat{H}'_{\text{drive}}$ has a ground-state

$$|\psi'_{\text{gs}}\rangle = |\psi(0)\rangle + O\left(\frac{\epsilon}{A(0)\Delta}\right) |\psi_{\perp}\rangle \quad (5.29)$$

where $|\psi_{\perp}\rangle$ is a normalized state vector orthogonal to $|\psi(0)\rangle$ and Δ is the energy gap between the ground and first-excited manifolds of the actual driver Hamiltonian $\hat{H}_{\text{drive}}$. Thus, assuming the driver Hamiltonian $\hat{H}_{\text{drive}}$ is not gapless (which is automatically true for all Hamiltonians on Hilbert spaces of finite dimension), the inequality in Eq. (5.19) is satisfied in the limit as $\epsilon \rightarrow 0$.

The key result shown here is that, for quenches where the control function $\Gamma(t)$ decreases monotonically, the energy expectation value of the problem Hamiltonian $\hat{H}_P$ cannot be higher than its initial value. Put another way, on average, a monotonic quench can never perform worse than random guessing. This result is important for two reasons. Firstly, although not being harmful to average solution quality is a rather weak statement, it applies very generally to a broad class of algorithms. Secondly, and more importantly, this result can be built upon to determine control functions that can provide a significant improvement, which is important for algorithm design. To do this, it is necessary to combine the result in this section with criteria for when the transfer of amplitude between computational basis states will be significant, which will be obtained in the next section.

5.4 Ensuring significant dynamics

Section 5.2 showed examples of a quantum quench giving significantly better performance than pure QWs. This section will consider theoretically how a significant improvement can occur. It was shown in section 5.3 that dynamics will never be detrimental; this means that, if dynamics occur, in general it will be beneficial. What remains is to determine the circumstances in which significant dynamics will occur.

5.4.1 Quantifying the strength of short-time dynamics

In the analytical solutions for unstructured search in a continuous-time setting (Roland and Cerf, 2002; Childs and Goldstone, 2004), the method involves analysing the dynamics in a two dimensional subspace. To obtain significant dynamics in this setting, the hopping rate γ or schedule functions $A(t)$, $B(t)$ must carefully balance the relative strengths of the driver and problem Hamiltonians, such that the off-diagonal terms in the two dimensional subspace are maximized. Motivated by this, shorter timescales can be investigated by examining local subspaces spanned by a pair of basis states. To analyse whether significant dynamics will occur, a similar analysis to characterise how strong the transitions are to locally redistribute amplitude has been performed. If these are large for most of the transitions mediated by the driver, then the system will generate a high level of dynamics on a short timescale; otherwise, it will not, although dynamics may still occur on longer timescales.

To get a measure of dynamics that can be efficiently estimated at all sizes, I have analysed individual pairs of computational basis states connected by the driver, to determine whether significant transfer occurs between them, irrespective of the state in the rest of the Hilbert space. Note that for classical problems in the setting considered here, the problem Hamiltonian is diagonal in the computational basis, hence all of its subspaces are, too. Consider two basis states $|j\rangle$ and $|k\rangle$ connected by the driver, i.e., $\langle j | \hat{H}_{\text{drive}} | k \rangle \neq 0$, and define an effective two-level system Hamiltonian

$$\hat{H}_{\Gamma}^{(jk)}(t) = \Gamma(t)\hat{H}_{\text{drive}}^{(jk)} + \hat{H}_P^{(jk)} \quad (5.30)$$

with the *local problem Hamiltonian* $\hat{H}_P^{(jk)}$ defined as

$$\hat{H}_P^{(jk)} = \begin{pmatrix} E^{(j)} & 0 \\ 0 & E^{(k)} \end{pmatrix}, \quad (5.31)$$

where $E^{(j)} = \langle j | \hat{H}_P | j \rangle$ is the energy of computational basis state $|j\rangle$ with respect to the problem Hamiltonian (similarly for k), and with the *local driver Hamiltonian* $\hat{H}_{\text{drive}}^{(jk)}$ defined as

$$\hat{H}_{\text{drive}}^{(jk)} = \begin{pmatrix} \langle j | \hat{H}_{\text{drive}} | j \rangle & \langle j | \hat{H}_{\text{drive}} | k \rangle \\ \langle k | \hat{H}_{\text{drive}} | j \rangle & \langle k | \hat{H}_{\text{drive}} | k \rangle \end{pmatrix}. \quad (5.32)$$

The extent to which the local subspace Hamiltonian $\hat{H}_\Gamma^{(jk)}(t)$ can transfer amplitude between the basis states $|j\rangle$ and $|k\rangle$ can be characterised by comparing the off-diagonal energy scale to the diagonal one. Define a local *transfer coefficient*, which takes values $0 \leq T^{(jk)} \leq 1$, as

$$T^{(jk)} = \mathcal{R} \left[\Gamma(t) \hat{H}_{\text{drive}}^{(jk)}, \hat{H}_P^{(jk)} \right] \quad (5.33)$$

$$\equiv \frac{2\Gamma(t) |\langle k | \hat{H}_{\text{drive}} | j \rangle|}{2\Gamma(t) |\langle k | \hat{H}_{\text{drive}} | j \rangle| + |\Delta_{jk}|}. \quad (5.34)$$

where

$$\Delta_{jk} = \left\{ \Gamma(t) \langle j | \hat{H}_{\text{drive}} | j \rangle + E^{(j)} \right\} - \left\{ \Gamma(t) \langle k | \hat{H}_{\text{drive}} | k \rangle + E^{(k)} \right\}$$

is the difference between the diagonal elements in the diagonal basis of the problem Hamiltonian.

Similarly, as implied by the energy redistribution mechanism described in section 5.3, transfer between driver eigenstates is also important. To capture this, a local *driver coefficient* $D^{(jk)}$ can be defined by transforming the local subspace Hamiltonian in $\hat{H}_\Gamma^{(jk)}(t)$ into the diagonal basis of the local driver Hamiltonian $\hat{H}_{\text{drive}}^{(jk)}$ and writing a similar expression to Eq. (5.34). That is,

$$D^{(jk)} = \mathcal{R} \left[U^{(jk)\dagger} \hat{H}_P U^{(jk)}, \Gamma(t) U^{(jk)\dagger} \hat{H}_{\text{drive}} U^{(jk)} \right], \quad (5.35)$$

where $U^{(jk)}$ is a unitary such that $U^{(jk)\dagger} \hat{H}_{\text{drive}}^{(jk)} U^{(jk)}$ is diagonal.

It is easily shown that, for unbiased drivers such as Eq. (2.6), the local driver coefficient $D^{(jk)}$

and local transfer coefficient $T^{(jk)}$ are related by $D^{(jk)} = 1 - T^{(jk)}$. This makes it clear there is a trade off between the two quantities to obtain significant dynamics under the combined Hamiltonian. The overall level of amplitude transfer expected can be quantified by the product of the transfer and driver coefficients $T^{(jk)}$ and $D^{(jk)}$, which is defined as the dynamic coefficient,

$$\text{Dyn}^{(jk)} = T^{(jk)} D^{(jk)}. \quad (5.36)$$

For unbiased drivers, since $D^{(jk)} = 1 - T^{(jk)}$, and $0 \leq D^{(jk)}, T^{(jk)} \leq 1$, it follows that $\text{Dyn}^{(jk)}$ satisfies $0 \leq \text{Dyn}^{(jk)} \leq 0.25$.

The dynamic coefficient $\text{Dyn}^{(jk)}$ captures the level of algorithmically useful local dynamics experienced by the system. In particular, if $\Gamma \gg 1$, then the driver Hamiltonian dominates and the problem Hamiltonian $\hat{H}_P$ will have little effect on the dynamics of the system. Since the initial state is the ground state of the driver Hamiltonian, the dynamics are driven by the much smaller problem Hamiltonian on short timescales. This limit is captured by the dynamical coefficient, as $D^{(jk)} \approx 0$, and hence $\text{Dyn}^{(jk)} \approx 0$. In the opposite extreme, if $\Gamma \ll 1$, then the problem Hamiltonian dominates, but since it is diagonal, the dynamics will consist almost entirely of phase rotations in the computational basis, and the amplitudes will change very little. This limit is captured by the transfer coefficient, as $T^{(jk)} \approx 0$, and hence $\text{Dyn}^{(jk)} \approx 0$.

To characterise the level of dynamics in the entire system, the mean value of $\text{Dyn}^{(jk)}$ can simply be taken over the values of j and k which correspond to a non-zero off-diagonal elements in $\hat{H}_{\text{drive}}$. That is, the *average* dynamic coefficient is defined as

$$\overline{\text{Dyn}} = \langle \text{Dyn}^{(jk)} \rangle_{jk} \quad (5.37)$$

where $\langle \cdot \rangle_{jk}$ represents the mean over all pairs of computational basis states j, k connected by the driver Hamilton $\hat{H}_{\text{drive}}$. Although Eq. (5.37) cannot be exactly calculated efficiently, it should in general be possible to approximate it efficiently (up to additive error) by sampling. This follows from the fact that the values of $\text{Dyn}^{(jk)}$ are bounded $0 \leq \text{Dyn}^{(jk)} \leq 0.25$, and therefore the error $\delta \overline{\text{Dyn}}$ can be reduced by the central limit theorem to the range this value can take, multiplied by the statistical noise in the sample, which scales as the square root of the number of samples, i.e.,

$$\delta \overline{\text{Dyn}} \sim \frac{0.25}{N_{\text{samples}}^{1/2}}. \quad (5.38)$$

As the average dynamic coefficient $\overline{\text{Dyn}}$ is calculated by considering only those states that are directly connected by the driver Hamiltonian H_{drive} , it naturally captures only the fastest quantum dynamics that are present in the system. For example, in the case of the transverse field driver from Eq. (2.6), $\overline{\text{Dyn}}$ only depends on transitions between states that differ by a single bit-flip, which will typically be happening much faster than those that involve two or more bit flips.

The minimum gap between the ground state and first excited state of the total Hamiltonian is often used in AQC (the adiabatic limit of QA) as an indication of the computational difficulty of different parts of the schedule. Although inspired by the analytical solution to the search problem, where balancing the driver and problem Hamiltonians corresponds to this minimum gap, there is no reason to expect the local Hamiltonians balanced by maximizing $\overline{\text{Dyn}}$ to also locate the global minimum gap, except in special cases. Fig. 5.8 shows a comparison, for two different $n = 9$ SK instances, between the average dynamic coefficient $\overline{\text{Dyn}}$ (red solid line, right axis) and the gap between the ground- and first excited-state (blue solid line, left axis). The quantities are plotted against the schedule parameter $s(t)$ in the AQC-like parameterization $A(t) = 1 - s(t), B(t) = s(t)$. The maximum of $\overline{\text{Dyn}}$ and the minimum gap are indicated by the red and blue dotted lines respectively. As in these examples, it is typical, for the SK spin glasses, for the maximum $\overline{\text{Dyn}}$ value to appear significantly before the minimum gap (i.e., closer to the driver end of the schedule). This could be related to the fact that the smallest gaps occur in a spin-glass phase in which dynamics are expected to be much slower, as described in Knysh (2016) and discussed in relation to the SK problem studied in Chapter 3. Transitioning slowly through the minimum gap is important for the long timescales of AQC, but it is not necessarily related to what is needed for maximizing the success probability for shorter run times. Away from the adiabatic limit, there are different mechanisms at play, as has been highlighted in Wong and Meyer (2016) and discussed in Chapter 3 and elsewhere (Morley et al., 2019).

5.4.2 Analytical bounds on $\overline{\text{Dyn}}$

Equipped with the definition of the average dynamic coefficient $\overline{\text{Dyn}}$, the next question to investigate is “when is it possible to find a value of $\Gamma(t)$ such that $\overline{\text{Dyn}}$ is large enough for significant short time dynamics to be generated?”. For simplicity, I will restrict the discussion to the unbiased driver case, when the local driver coefficient $D^{(jk)}$ and local transfer coefficient

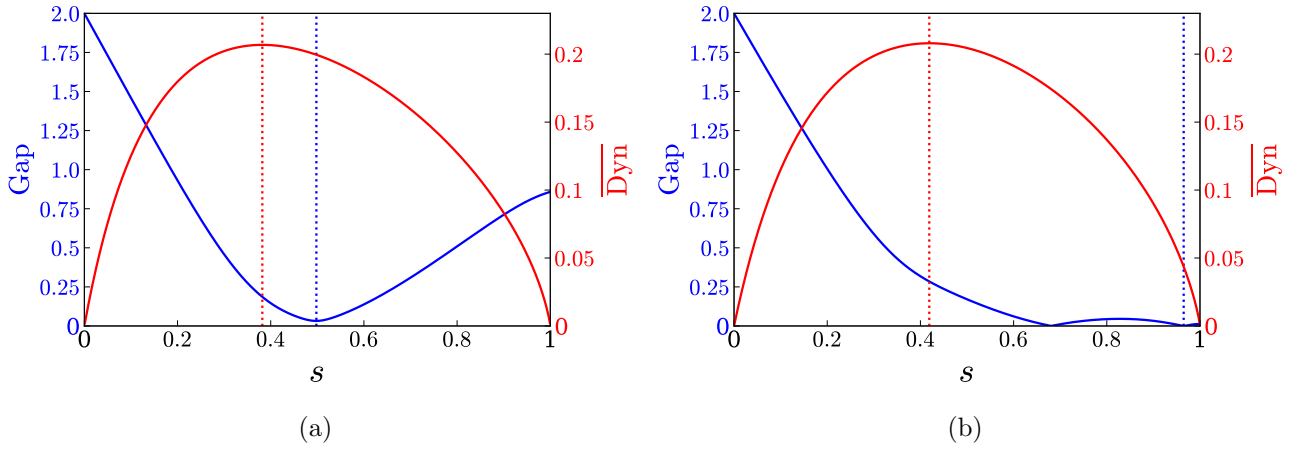


Figure 5.8: Comparison, for two different $n = 9$ SK instances, between the average dynamic coefficient $\overline{\text{Dyn}}$ (red solid line, right axis) with the gap between the ground- and first excited-state (blue solid line, left axis). The quantities are plotted against the schedule parameter s in the AQC-like parametrization $A(t) = 1 - s(t)$, $B(t) = s(t)$. The maximum of $\overline{\text{Dyn}}$ and the minimum gap are indicated by the red and blue dotted lines respectively. Left: the instance with ID code **ovcjhw bhtcpcvwicoxpdvpvjzqojril** from Chancellor et al. (2019), used in Figs. 5.3, 5.4 and 5.6. Right: the $n = 9$ instance from Chancellor et al. (2019) (ID code **cpahzppaxangdnisyqutdbbjlkqamc**) with the smallest minimum gap.

$T^{(jk)}$ are related by $D^{(jk)} = 1 - T^{(jk)}$. In this case, the local dynamic coefficient $\text{Dyn}^{(jk)}$ can be written in terms of the driver strength $\Gamma(t)$ and a single *scaled gap* parameter $\zeta_{jk} = \frac{|\Delta_{jk}|}{2|\langle k | \hat{H}_{\text{drive}} | j \rangle|}$ as

$$\text{Dyn}^{(jk)} = \frac{\zeta_{jk}/\Gamma(t)}{(1 + \zeta_{jk}/\Gamma(t))^2}. \quad (5.39)$$

Let p_ζ be the probability density function that governs the distribution of ζ_{jk} in the particular problem and driver Hamiltonians under consideration. It can be shown that the maximum value attained by the average dynamic coefficient $\overline{\text{Dyn}}$ for any choice of driver strength $\Gamma(t)$ has a lower bound which can be stated formally as

$$\max_t(\overline{\text{Dyn}}) \geq \max_{0 < c < 1} \left[\frac{1 - c}{(2 - c)^2} \left(1 - \frac{1}{c^2} \frac{\mu_2(p_\zeta)}{\mu_1^2(p_\zeta)} \right) \right], \quad (5.40)$$

where $\mu_1(p_\zeta)$ ($\mu_2(p_\zeta)$) is the first (second) central moment of the distribution governed by the probability density function p_ζ . To prove this lower bound, it is useful to first prove the following general result about distributions: assume that the distribution $p(x)$ has a finite second moment

$$\mu_2(p) = \int_{-\infty}^{\infty} dx p(x) (x - \mu_1(p))^2, \quad (5.41)$$

where

$$\mu_1(p) = \int_{-\infty}^{\infty} dx p(x)x, \quad (5.42)$$

is the first moment (mean). Then, for any choice of values $x_{\max} > x_{\min}$ such that $\mu_1(p) = \frac{1}{2}(x_{\max} + x_{\min})$, the probability for x to be in the interval $[x_{\min}, x_{\max}]$ can be lower bounded as

$$\int_{x_{\min}}^{x_{\max}} dx p(x) \geq 1 - \frac{4\mu_2(p)}{(x_{\max} - x_{\min})^2} \quad (5.43)$$

To see this, first recognise that the distribution $q(x) = \frac{1}{2}\delta(x_{\min} - \epsilon) + \frac{1}{2}\delta(x_{\max} + \epsilon)$ has the minimum possible second moment while having no support in the interval $[x_{\min}, x_{\max}]$, where δ is the Dirac delta distribution. In the limit $\epsilon \rightarrow 0$, the second moment of this distribution is $\mu_2(q) = \frac{1}{4}(x_{\max} - x_{\min})^2$. Thus, if $\mu_2(p) < \mu_2(q)$, then $p(x)$ must have some support within the range $[x_{\min}, x_{\max}]$. The second moment $\mu_2(p)$ can be lower bounded as

$$\begin{aligned} \mu_2(p) &= \int_{-\infty}^{\infty} dx p(x)(x - \mu_1(p))^2 \\ &\geq \int_{-\infty}^{x_{\min}} dx p(x)(x - \mu_1(p))^2 + \int_{x_{\max}}^{\infty} dx p(x)(x - \mu_1(p))^2 \\ &\geq \mu_2(q) \left(\int_{-\infty}^{x_{\min}} dx p(x) + \int_{x_{\max}}^{\infty} dx p(x) \right) \\ &= \mu_2(q) \left(1 - \int_{x_{\min}}^{x_{\max}} dx p(x) \right), \end{aligned}$$

from which the result in Eq. (5.43) follows directly.

Now, the result in Eq. (5.40) can be shown. Let $\eta_{jk} = \frac{\zeta_{jk}}{\Gamma}$. Furthermore, let p_{ζ} and p_{η} be probability density functions that govern the distribution of the values ζ_{jk} and η_{jk} , respectively, over a set of problem instances. Let $\mu_1(p)$ and $\mu_2(p)$ refer to the first and second moments, respectively, of a distribution governed by the probability density function p .

The dynamic coefficient is

$$\text{Dyn}^{(jk)} = \frac{\eta_{jk}}{(1 + \eta_{jk})^2}, \quad (5.44)$$

so, consider the function

$$f(x) = \frac{x}{(1 + x)^2} \quad (5.45)$$

where $x > 0$.

Let x be distributed according to the probability density function p_η . The expectation value $\langle f(x) \rangle_x$ is then

$$\begin{aligned}
 \langle f(x) \rangle_x &= \int_0^\infty dx p_\eta(x) f(x) \\
 &= \int_{x_{\min}}^{x_{\max}} dx p_\eta(x) f(x) + \int_0^{x_{\min}} dx p_\eta(x) f(x) + \\
 &\quad \int_{x_{\max}}^\infty dx p_\eta(x) f(x) \\
 &= P_\eta(x_{\min} < x < x_{\max}) \langle f(x) \rangle_{x_{\min}}^{x_{\max}} + \\
 &\quad P_\eta(x \geq x_{\min}) \langle f(x) \rangle_0^{x_{\min}} + P_\eta(x_{\max} \geq x) \langle f(x) \rangle_{x_{\max}}^\infty
 \end{aligned} \tag{5.46}$$

where $x_{\max} > x_{\min}$, $P_\eta(\dots)$ is the probability of its argument being true if η is distributed according to p_η , and $\langle f(x) \rangle_a^b$ is the expectation value of $f(x)$ if x is distributed according to a (renormalized) version of p_η with all support on $x < a$ and $x > b$ removed. As $f(x)$ is positive for all $x > 0$, this leads to the lower bound on $\langle f(x) \rangle_x$,

$$\langle f(x) \rangle_x \geq P_\eta(x_{\min} < x < x_{\max}) \langle f(x) \rangle_{x_{\min}}^{x_{\max}} \tag{5.47}$$

$$> P_\eta(x_{\min} < x < x_{\max}) \min_{x_{\min} < x < x_{\max}} [f(x)]. \tag{5.48}$$

Since $f(x)$ is also convex, it is true that

$$\min_{x_{\min} < x < x_{\max}} [f(x)] = \min [f(x_{\min}), f(x_{\max})]. \tag{5.49}$$

Now, let the interval $[x_{\min}, x_{\max}]$ be of width $2c$ (for some $c > 0$), and centred on the mean $\mu_1(p_\eta)$ (thereby also constraining $c < \mu_1(p_\eta)$). That is, $x_{\min} = \mu_1(p_\eta) - c$ and $x_{\max} = \mu_1(p_\eta) + c$, and it must be determined which of $f(\mu_1(p_\eta) - c)$ and $f(\mu_1(p_\eta) + c)$ is smaller. To do this, consider under what conditions it is true that

$$f(\mu_1(p_\eta) - c) < f(\mu_1(p_\eta) + c). \tag{5.50}$$

It can be shown that Eq. (5.50) is true when

$$c^2 > \mu_1^2(p_\eta) - 1 \quad (5.51)$$

This inequality means that, when the mean $\mu_1(p_\eta)$ is greater than 1, the truth of the inequality in Eq. (5.50) depends on the value of c , but for $\mu_1(p_\eta) \leq 1$, it is always true. Therefore, making the choice $\Gamma = \mu_1(p_\zeta)$ (the mean of the distribution of ζ rather than η) gives that $\mu_1(p_\eta) = 1$, which means the inequality $f(1 - c) < f(1 + c)$ is always true (where now $0 < c < 1$), and consequently

$$\begin{aligned} \min_{(1-c) < x < (1+c)} [f(x)] &= f(1 - c) \\ \langle f(x) \rangle_{1-c}^{1+c} &> f(1 - c) \\ \langle f(x) \rangle_x &> P_\eta(1 - c < x < 1 + c) f(1 - c). \end{aligned} \quad (5.52)$$

Now, since $\zeta_{jk} \equiv \Gamma \eta_{jk} = \mu_1(p_\zeta) \eta_{jk}$, it is true that

$$\begin{aligned} P_\eta(1 - c < x < 1 + c) &= \\ P_\zeta(\mu_1(p_\zeta)(1 - c) < x < \mu_1(p_\zeta)(1 + c)) \end{aligned} \quad (5.53)$$

where $P_\zeta(\dots)$ is the probability of its argument being true if x is distributed according to p_ζ .

Applying the result Eq. (5.43) gives

$$\begin{aligned} P_\zeta(\mu_1(p_\zeta)(1 - c) < x < \mu_1(p_\zeta)(1 + c)) &\geq \\ 1 - \frac{4\mu_2(p_\zeta)}{([\mu_1(p_\zeta)(1 + c)] - [\mu_1(p_\zeta)(1 - c)])^2} &= 1 - \frac{1}{c^2} \frac{\mu_2(p_\zeta)}{\mu_1^2(p_\zeta)}. \end{aligned} \quad (5.54)$$

Putting this all together gives

$$\max_{\Gamma}(\overline{\text{Dyn}}) \geq f(1 - c) \left[1 - \frac{1}{c^2} \frac{\mu_2(p_\zeta)}{\mu_1^2(p_\zeta)} \right] \quad (5.55)$$

$$= \frac{1 - c}{(2 - c)^2} \left(1 - \frac{1}{c^2} \frac{\mu_2(p_\zeta)}{\mu_1^2(p_\zeta)} \right). \quad (5.56)$$

While this inequality gives a valid lower bound on $\overline{\text{Dyn}}$, the greatest lower bound can be found by maximising over c , which leads directly to the lower bound in Eq. (5.40)

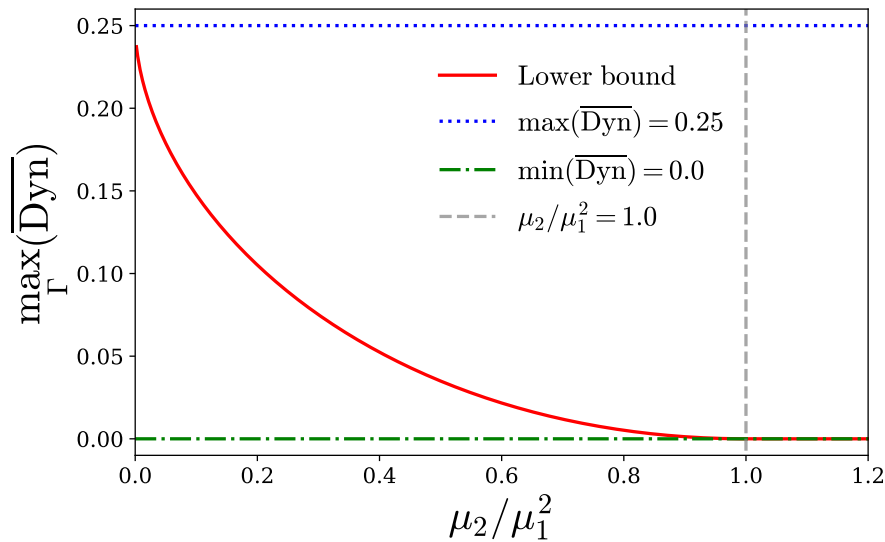


Figure 5.9: Semi-analytical lower bound (solid, red) on $\overline{\text{Dyn}}$ as a function of ratio of moments $\mu_2(p_\zeta)/\mu_1^2(p_\zeta)$ of the distribution, governed by p_ζ , of the rescaled energy gaps ζ_{jk} between computational basis states connected by the driver Hamiltonian H_{drive} . Also shown, minimum (0.0, dot-dashed, green) and maximum (0.25, dotted, blue) possible values of $\overline{\text{Dyn}}$. The lower-bound is non-trivial for $\mu_2(p_\zeta)/\mu_1^2(p_\zeta) < 1.0$ (left of grey dashed line), trivially zero otherwise.

Note that this bound has been obtained by choosing the specific driver strength $\Gamma = \mu_1(p_\zeta)$, i.e., the mean of the rescaled local gaps, which is not necessarily optimal, but serves to produce a non-trivial lower bound.

It is illuminating to look at the shape of this bound, which can be easily computed numerically for any given value of the ratio of moments $\mu_2(p_\zeta)/\mu_1^2(p_\zeta)$. The bound is plotted for the interesting range of the ratio of moments in Fig. 5.9. It can be seen that the lower-bound is non-trivial when $\mu_2(p_\zeta)/\mu_1^2(p_\zeta) < 1.0$, but is trivially zero otherwise. This shows that there is a continuous range where $\overline{\text{Dyn}}$ is bounded away from zero, and hence dynamics will definitely happen on short timescales, even for non-optimal choices of $\Gamma(t)$. This bound is in general far from tight, but still leads to some interesting examples. Next, I will illustrate the calculation of $\overline{\text{Dyn}}$ and the lower bound in Eq. (5.40) for some specific cases.

5.4.3 Example: two qubit system

As a simple example, consider the problem Hamiltonian

$$\hat{H}_P^{(2Q)} = -\hat{Z}_1\hat{Z}_2 - \frac{1}{2}\hat{Z}_1$$

as defined in Eq. (5.4), with a transverse field driver as defined in Eq. (2.6). For this problem Hamiltonian, there are four two level subspaces connected by the driver, $|00\rangle \leftrightarrow |10\rangle$, $|00\rangle \leftrightarrow |01\rangle$, $|10\rangle \leftrightarrow |11\rangle$, and $|01\rangle \leftrightarrow |11\rangle$, with $|\Delta_{jk}| = 3, 2, 2, 1$ respectively. $\overline{\text{Dyn}}$ can thus be calculated exactly as

$$\begin{aligned} \overline{\text{Dyn}} &= \frac{1}{4} \left(\frac{3/(2\Gamma)}{(1 + 3/(2\Gamma))^2} + \frac{1/(2\Gamma)}{(1 + 1/(2\Gamma))^2} + \right. \\ &\quad \left. 2 \frac{2/(2\Gamma)}{(1 + 2/(2\Gamma))^2} \right) \\ &= \frac{\Gamma}{2} \left(\frac{3}{(3 + 2\Gamma)^2} + \frac{1}{(1 + 2\Gamma)^2} + \right. \\ &\quad \left. \frac{4}{(2 + 2\Gamma)^2} \right), \end{aligned} \quad (5.57)$$

where the time dependence in $\Gamma(t)$ has been omitted for clarity. The maximum value of $\overline{\text{Dyn}}$ can be obtained by maximizing the expression in Eq. (5.57) with respect to Γ . This is easiest done numerically, giving $\max_{\Gamma}(\overline{\text{Dyn}}) \sim 0.241$ for $\Gamma \simeq 0.941$. Comparing with the bound in Eq. (5.40), the first moment of p_{ζ} is $\mu_1(p_{\zeta}) = 1$, while the second moment is $\mu_2(p_{\zeta}) = 0.125$. Using the ratio $\frac{\mu_2(p_{\zeta})}{\mu_1^2(p_{\zeta})} = 0.125$, the lower bound is $\max_{\Gamma} \overline{\text{Dyn}} \gtrsim 0.135$. This is just over half the actual value, but holds for any Hamiltonian with the same moments of the distribution.

5.4.4 Example: Sherrington-Kirkpatrick spin-glass

It is worth considering the SK spin-glass problem Hamiltonian given in Eq. (2.8). Consider the driver Hamiltonian $\hat{H}_{\text{drive}}$ to be the transverse field defined in Eq. (2.6). Due to the promising results presented in Chapter 3 for solving this problem with QWs, as well as for the more general quenches presented in section 5.2, it can be intuitively expected to be generally possible to find values of Γ for which the average dynamic coefficient $\overline{\text{Dyn}}$ takes an appreciable value.

The transverse field driver only connects pairs of states j, k that differ by a single bit flip. Thus, it can be seen from Eq. (2.8) that, for all such pairs, the energy difference can be written

$$\Delta_{jk} = - \sum_{b \neq a} s_{ab}^{(j)} J_{ab} - 2s_a^{(j)} h_a \quad (5.58)$$

where a is the index of the spin that is flipped between states $|j\rangle$ and $|k\rangle$, the sum runs over b which indexes the other spins, $s_{ab}^{(j)}$ is the eigenvalue (± 1) of the operator $Z_a Z_b$ on the state $|j\rangle$ and $s_a^{(j)}$ is the eigenvalue (± 1) of the operator Z_a on the state $|j\rangle$. The gap Δ_{jk} in Eq. (5.58) is a sum of normally distributed variables with mean 0, and so Δ_{jk} is itself a normally distributed variable with mean 0, and can be shown to have a standard deviation $\varsigma = \sqrt{2(n+1)}\sigma$, where n is the number of spins (qubits). Then, since $\langle k | \hat{H}_{\text{drive}} | j \rangle = 1$ for the unbiased transverse field driver, the scaled gap ζ_{jk} is distributed according to the *half-normal* distribution with probability density function

$$p_\zeta(\zeta) = \frac{1}{\varsigma\sqrt{2\pi}} \exp\left(-\frac{\zeta^2}{8\varsigma^2}\right), \quad \zeta \geq 0 \quad (5.59)$$

For this distribution, it can be shown that the ratio of moments is

$$\frac{\mu_2(p_\zeta)}{\mu_1^2(p_\zeta)} = \frac{1 - 2/\pi}{2/\pi} \quad (5.60)$$

$$\approx 0.571. \quad (5.61)$$

It should be emphasised that this is independent of the width ς of the distribution of the scaled gap ζ_{jk} . For this value of the ratio, the lower bound shown in Fig. 5.9 is

$$\max_{\text{f}}(\overline{\text{Dyn}}) \gtrsim 0.03. \quad (5.62)$$

While this value is small compared to the maximum possible value of $\overline{\text{Dyn}} = 0.25$, which is not unexpected for a hard problem (NP hard), it should be emphasised that it is independent of the width ς of the distribution of the scaled gap ζ_{jk} and thus does not scale with the system size. Bounding $\overline{\text{Dyn}}$ away from zero for all sizes proves that dynamics will occur over short timescales for suitable control parameters, thus providing evidence that the scaling found in Chapter 3 may continue to be useful problem sizes.

5.4.5 Example: unstructured search

It is worth considering the problem of unstructured search on n qubits as a contrasting example. In the unstructured search problem, a single computational basis state $|m\rangle$, out of the total $N = 2^n$ basis states, is marked by being given a lower energy, and its simple problem Hamiltonian

is given in Eq. (2.14). Again the driver Hamiltonian $\hat{H}_{\text{drive}}$ is taken to be the transverse field defined in Eq. (2.6). While unstructured search is a well known example with a provable quantum advantage, the algorithms which yield this advantage all involve coherent operations on time scales of order $\sqrt{N} = 2^{\frac{n}{2}}$ rather than the short-time dynamics discussed here. As such, it is to be intuitively expected that the lower bound in Eq. (5.40) would be large in this case.

Of the $n 2^{n-1}$ total off diagonal matrix element pairs in the transverse field driver, only n of these will connect a pair of computational basis states with non-zero energy difference, having energy difference $\Delta_{jk} = 1$, with the remaining $n 2^{n-1} - n$ pairs having zero energy-difference $\Delta_{jk} = 0$. Therefore, the distribution of scaled gaps ζ_{jk} can be written as

$$p_{\zeta}(\zeta) = \frac{n}{n 2^{n-1}} \delta(\zeta - \frac{1}{2}) + \left(1 - \frac{n}{n 2^{n-1}}\right) \delta(\zeta) \quad (5.63)$$

Calculating the first and second central moments of this distribution gives

$$\mu_1(p_{\zeta}) = \frac{1}{2^n} \quad (5.64)$$

$$\mu_2(p_{\zeta}) = \frac{1}{2^{n+1}} - \left(\frac{1}{2^n}\right)^2 \quad (5.65)$$

and so the relevant ratio of moments is

$$\frac{\mu_2(p_{\zeta})}{\mu_1^2(p_{\zeta})} = 2^{n-1} - 1 \quad (5.66)$$

Looking at the plot of the lower bound in Fig. 5.9, it can be seen that, for unstructured search the bound is trivially zero for all $n > 1$. The exact value can also be calculated, using Eq. (5.39). For each of the $n 2^{n-1} - n$ pairs of states j, k with $|\Delta_{jk}| = 0$, $\text{Dyn}^{(jk)} = 0 \forall \Gamma$. For the remaining n pairs of states j, k with $|\Delta_{jk}| = 1$, the choice of driver strength $\Gamma = 1/2$ will maximise $\text{Dyn}^{(jk)} = 0.25$. Thus, the average dynamic coefficient for unstructured search is

$$\begin{aligned} \overline{\text{Dyn}} &= \frac{1}{2^{n-1}} \times 0.25 \\ &= \frac{1}{2^{n+1}} \end{aligned} \quad (5.67)$$

which tends toward the lower bound of zero in the limit as $n \rightarrow \infty$.

This shows that, for search, most two-level subspaces do not exhibit dynamics and probability

enhancement of the marked state can only happen through finely tuned control. For AQC, this is achieved by slowly adjusting the Hamiltonian within a precise range so that the system can follow a very delicate path, whereas for QW this is achieved by reaching a finely tuned resonance between the marked state and the rest of a symmetric subspace of the Hilbert space. While interpolations between these two extremes are possible (Morley et al., 2019), all of the interpolated algorithms also rely on dynamics of a two level system with a gap proportional to $\sqrt{N} = 2^{\frac{n}{2}}$. In such a system, significant dynamics cannot occur in the timescales of rapid quenches, $O(1)$ or $O(\text{poly}(n))$.

5.5 Using dynamics to find heuristic quench parameters

As mentioned in section 5.4, the average dynamic coefficient $\overline{\text{Dyn}}$ can in general be efficiently estimated by sampling. In this section, I will show via two practical examples that this estimate can be used to develop heuristic methods for setting the control function $\Gamma(t)$, or equivalently, $A(t)$ and $B(t)$, for a rapid quench, in both the QW and QA settings. In both cases, the unbiased transverse field driver Hamiltonian defined in Eq. (2.6) is used. First, the QW algorithm will be considered, starting with a simplified example of a two qubit system. Then, a heuristic for the SK spin-glass problem will be developed, and it will be shown that it performs almost as well as the numerically fine-tuned heuristic described in Chapter 3, without needing any fine-tuning. Second, a simple heuristic method will be developed for defining a schedule for a time-dependent rapid quench. This will also be applied to the SK spin-glass, and it will be shown to outperform a linear ramp.

In all the examples discussed in this section, the average dynamic coefficient $\overline{\text{Dyn}}$ has been computed numerically using all non-zero j, k pairs, rather than estimated by sampling such pairs. This is computationally easy to do at these problem sizes, making it possible to separate the effectiveness of the heuristic from errors due to sampling.

5.5.1 Heuristic hopping rate for a quantum walk

For QW, the average dynamic coefficient $\overline{\text{Dyn}}$ is a function of the chosen hopping rate $\Gamma(t) = \gamma$. Informed by the result in section 5.3 that dynamics will typically be useful, it follows that by

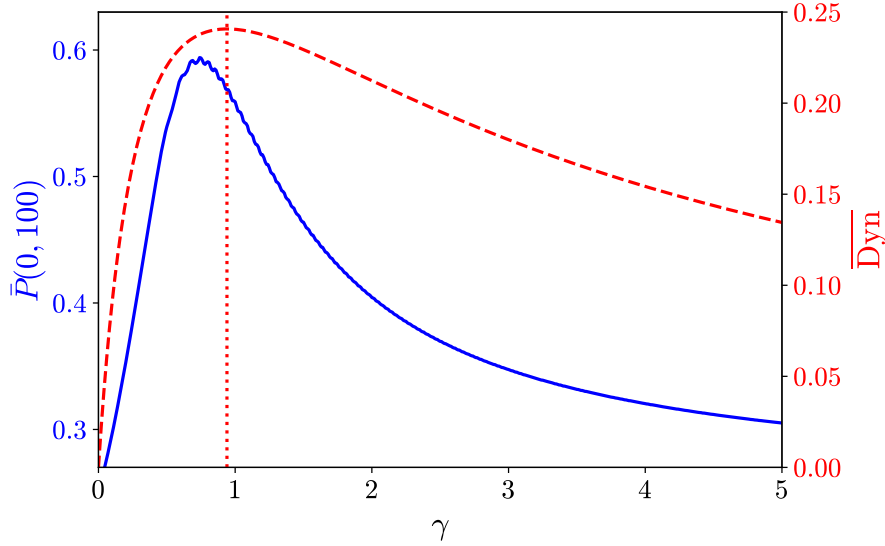


Figure 5.10: Average success probability $\bar{P}(0, 100)$ between $t = 0$ and $t = 100$ (blue, solid), calculated based on 10,000 independent random points within this range and $\overline{\text{Dyn}}$ (red, dashed) versus γ for the two qubit system given in Eq. (5.4). Dotted vertical line indicates the value of γ_{Dyn} .

maximizing $\overline{\text{Dyn}}$ a heuristic hopping rate γ_{Dyn} can be obtained, that should ensure significant dynamics occur over short timescales. For the two qubit Hamiltonian from Eq. (5.4), Fig. 5.10 shows how the average success probability within 100 dimensionless time units $\bar{P}(0, 100)$ varies with γ . For this two qubit system, $\overline{\text{Dyn}}$ can be exactly calculated, see section 5.4.3, and is shown in Fig. 5.10. The maximum value of $\overline{\text{Dyn}}$ gives a value for γ_{Dyn} which is a good quality estimate for the value of γ_{opt} . Using bisection and a numerically calculated derivative, it was found that $\gamma_{\text{Dyn}} \approx 0.864$, while the peak of $\bar{P}(0, 100)$ occurs at a slightly lower value of γ . Since the peak of $\bar{P}(0, 100)$ is quite broad, the discrepancy between γ_{Dyn} and γ_{opt} only reduces $\bar{P}(0, 100)$ by a small amount, as can be seen in Fig. 5.10.

To test how well this heuristic hopping rate works for a more realistic example, γ_{Dyn} has been numerically calculated for each instance of size $5 \leq n \leq 15$ of the SK spin glass problems used in Chapter 3. This was done by performing a bisection optimization to maximise the value of $\overline{\text{Dyn}}$ as a function of γ for each instance. Following the methods used in Chapter 3, a short-time QW was performed and the success probability

$$\bar{P}(t_{\text{short}}, \Delta t_{\text{short}}) = \int_{\frac{12.5}{\sqrt{n}}}^{\frac{17.5}{\sqrt{n}}} dt P(t), \quad (5.68)$$

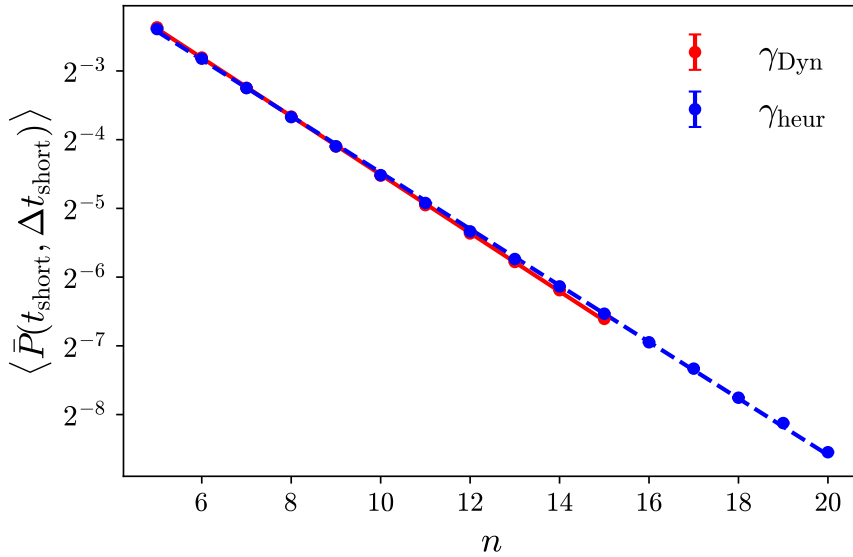


Figure 5.11: Log-linear plot of average *short time* success probability $\langle P_{\text{short}} \rangle$ against number of qubits n for QWs on the spin-glass dataset from Chapter 3, using the heuristic hopping rate γ_{Dyn} derived for each instance by optimizing the average dynamic coefficient $\overline{\text{Dyn}}$ (red). Also shown for comparison is the probability obtained using the fine-tuned heuristic hopping rate γ_{heur} (blue) described in Chapter 3.

as defined in subsection 3.2.5, was calculated. This is shown (red line) for each size in Fig. 5.11. Included for comparison (blue line) are the results from Chapter 3 using the fine-tuned heuristic γ_{heur} defined there, using properties of the eigenvalue distribution for the spin glass problem Hamiltonian. It can be seen that, despite γ_{heur} being numerically fine-tuned specifically for the SK spin-glass problem, it performs only marginally better the general method used here. Fitting the data produces $\langle \bar{P}(t_{\text{short}}, \Delta t_{\text{short}}) \rangle \sim O(N^{(-0.411 \pm 0.002)})$ for γ_{heur} compared to $\langle \bar{P}(t_{\text{short}}, \Delta t_{\text{short}}) \rangle \sim O(N^{(-0.425 \pm 0.001)})$ for γ_{Dyn} . The eigenvalue distribution used in Chapter 3 would not generally be available to calculate γ for real problems; this comparison shows that using $\overline{\text{Dyn}}$ is a viable method for determining a useful value for γ in this case.

For the small size instances used here, all the values of $\text{Dyn}^{(jk)}$ have been used to calculate the average in the definition of $\overline{\text{Dyn}}$ in Eq. (5.37). It can be shown that the error in γ_{Dyn} due to sampling a subset of $\text{Dyn}^{(jk)}$ values stays manageable for larger sizes. Consider a small error $\delta\gamma$ in γ . Doing a Taylor expansion of $\overline{\text{Dyn}}(\gamma)$ around its peak value $\overline{\text{Dyn}}_{\text{max}}$ gives

$$\delta\overline{\text{Dyn}} = \overline{\text{Dyn}}_{\text{max}} - \overline{\text{Dyn}}(\gamma_{\text{Dyn}} + \delta\gamma) = -(\delta\gamma)^2 \left. \frac{\partial^2 \overline{\text{Dyn}}(\gamma)}{\partial \gamma^2} \right|_{\gamma=\gamma_{\text{Dyn}}} + O((\delta\gamma)^3), \quad (5.69)$$

where γ_{Dyn} is the value of γ the heuristic would find² with the exact $\overline{\text{Dyn}}_{\text{max}}$. Using the sampling error in $\overline{\text{Dyn}}$ from Eq. (5.38) and rearranging yields

$$\delta\gamma \propto N_{\text{sample}}^{-\frac{1}{4}} \left(- \frac{\partial^2 \overline{\text{Dyn}}(\gamma)}{\partial \gamma^2} \Big|_{\gamma=\gamma_{\text{Dyn}}} \right)^{-\frac{1}{2}}. \quad (5.70)$$

This is a general expression that can be used for any problem Hamiltonian. For the SK spin glass, the distribution of the scaled gaps from Eq. (5.59) and the definition of $\text{Dyn}^{(j,k)}$ from Eq. (5.39) can be used together to obtain the average value of $\overline{\text{Dyn}}(\gamma)$ for SK instances, $\langle \overline{\text{Dyn}} \rangle(\gamma)$

$$\langle \overline{\text{Dyn}} \rangle(\gamma) = \frac{1}{\varsigma \sqrt{2\pi}} \int_0^\infty d\varsigma \exp \left\{ -\frac{\varsigma^2}{8\varsigma^2} \right\} \frac{\varsigma/\gamma}{(1 + \varsigma/\gamma)^2}. \quad (5.71)$$

Making the substitution $z = \varsigma/(2\sqrt{2}\varsigma)$, to remove the ς dependence in the exponential gives

$$\langle \overline{\text{Dyn}} \rangle(\gamma) = \frac{8}{\sqrt{2\pi}} \int_0^\infty dz z \exp(-z^2) \frac{\gamma/\varsigma}{(\gamma/\varsigma + 2\sqrt{2}z)^2}. \quad (5.72)$$

Differentiating with respect to γ gives

$$\frac{\partial}{\partial \gamma} \langle \overline{\text{Dyn}} \rangle(\gamma) = \frac{8}{\varsigma \sqrt{2\pi}} \int_0^\infty dz z \exp(-z^2) \frac{2\sqrt{2}z - \gamma/\varsigma}{(\gamma/\varsigma + 2\sqrt{2}z)^3} \quad (5.73)$$

and differentiating again gives

$$\frac{\partial^2}{\partial \gamma^2} \langle \overline{\text{Dyn}} \rangle(\gamma) = \frac{8}{\varsigma^2 \sqrt{2\pi}} \int_0^\infty dz 2z \exp(-z^2) \frac{\gamma/\varsigma - 4\sqrt{2}z}{(\gamma/\varsigma + 2\sqrt{2}z)^4} \quad (5.74)$$

This needs to be evaluated at $\gamma = \gamma_{\text{Dyn}}$, at the peak of $\langle \overline{\text{Dyn}} \rangle(\gamma)$. It can be seen by inspecting Eq. (5.72) that this occurs at a fixed value of γ/ς . Hence, the value of the actual integral in Eq. (5.74) evaluated at $\gamma = \gamma_{\text{Dyn}}$ is independent of n , so the scaling with n of the double derivative at $\gamma = \gamma_{\text{Dyn}}$ is determined solely by the ς^{-2} prefactor in Eq. (5.74). Using this information in Eq. (5.70) gives

$$\delta\gamma \propto N_{\text{sample}}^{-\frac{1}{4}} \varsigma. \quad (5.75)$$

²Note that the first order term is absent, and the second order term is negative, because the sampling is being done at the maximum, and it is implicitly assumed that there are no other peaks in $\overline{\text{Dyn}}$ which take similar values. If $\frac{\partial^2 \overline{\text{Dyn}}(\gamma)}{\partial \gamma^2} \Big|_{\gamma=\gamma_{\text{Dyn}}}$ vanishes, then the next non-zero even derivative should be used.

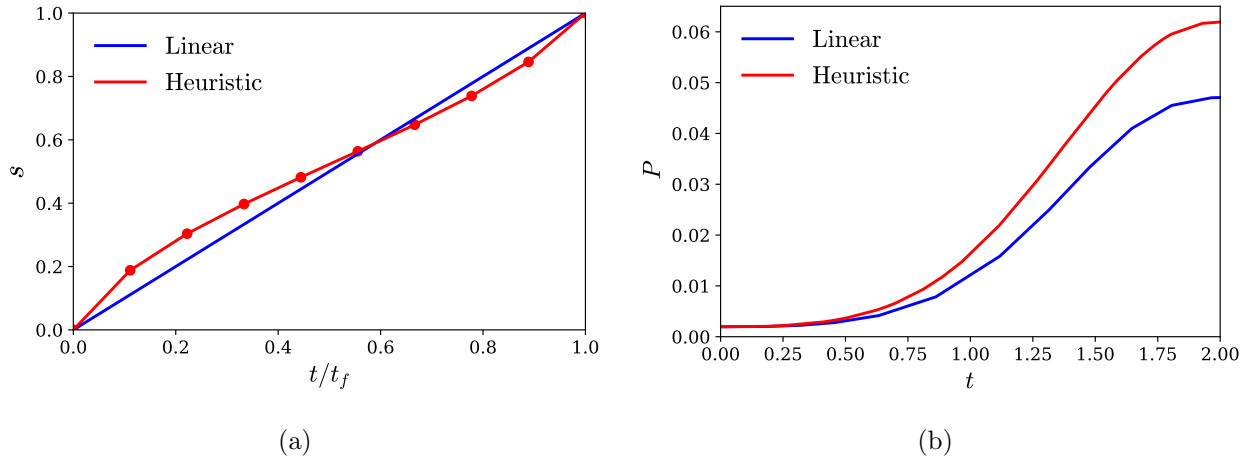


Figure 5.12: Left: A heuristic quench schedule of duration $t_f = 2.0$ (red) derived from the average dynamic coefficient $\overline{\text{Dyn}}$ for a typical nine qubit spin-glass instance from Chapter 3. For comparison, a linear schedule (also of duration $t_f = 2.0$) is also shown. Right: The instantaneous success probability $P(t)$ for measuring the problem ground-state for each time t as the quench progresses along the heuristic schedule (red) and the linear schedule (blue).

Recalling from section 5.4.4 that $\varsigma \propto \sqrt{2(n+1)}$ for these SK spin glasses then gives

$$\delta\gamma \propto N_{\text{sample}}^{-\frac{1}{4}}(n+1)^{1/2}. \quad (5.76)$$

In Chapter 3, it was determined numerically that the peak in the success probability as a function of γ is very broad for SK spin glasses, and that the width of this peak decreases as $1/n$. Combined with Eq. (5.76), this means the sampling rate to calculate $\overline{\text{Dyn}}$ needs to increase by a $\text{poly}(n)$ factor as n increases, in order to determine γ_{Dyn} to sufficient accuracy. Since n corresponds to the number of qubits, this can be done efficiently.

5.5.2 Heuristic schedule for quantum annealing

For a time-dependent rapid quench of the form $\hat{H}_{AB}(t)$ defined in Eq. (5.1) and total duration t_f , a common choice of control functions, inspired by the adiabatic algorithm, is $A(t) = 1 - s(t)$ and $B(t) = s(t)$, where $s(t)$ is a schedule function with boundary conditions $s(0) = 0$ and $s(t_f) = 1$. In the absence of any knowledge about where along the schedule useful computation can happen, the schedule function is often set to be the linear function $s(t) = t/t_f$. The average dynamic coefficient $\overline{\text{Dyn}}$ provides a measure of the level of dynamics at each point along the schedule. Intuition gained from section 5.3 suggests that the linear schedule can

in general be improved by spending less time in regions where $\overline{\text{Dyn}}$ is small and more time in regions where $\overline{\text{Dyn}}$ is large. A straightforward way to do this is to choose $\frac{ds}{dt} \propto \frac{1}{\overline{\text{Dyn}}}$ (the constant of proportionality is set by the boundary conditions $s(0) = 0$ and $s(T) = 1$). I have approximated such a schedule for a typical nine qubit SK spin-glass instance, as shown in Fig. 5.12(a) (red line). This was done by fixing the value of the points marked by circles according to $\Delta s \propto \frac{\Delta t}{\overline{\text{Dyn}}}$, subject to the boundary conditions, and then linearly interpolating between them. A linear schedule $s(t) = t/t_f$ (blue line) is also shown for comparison. Fig. 5.12(b) shows the instantaneous success probability $P(t)$ for measuring the problem ground-state as the quench progresses along the heuristic schedule (red line) and the linear schedule (blue line) for quench duration of $t_f = 2$. It can be seen that the simple heuristic used here has resulted in a significant improvement in success probability at the end of the schedule. I have checked sufficiently many of the instances to determine that this level of improvement is typical for this size of problem and total time duration $t_f = 2$. Further improvements may be available by varying t_f or choosing a different function of $\overline{\text{Dyn}}$ for $\frac{ds}{dt}$.

5.6 Chapter summary

In this chapter, I have generalised and extended the work that began in Chapter 3 to time-varying quantum annealing (QA) schedules. In Chapter 3, I provided numerical evidence for the ability of the continuous-time quantum walk (QW) to solve NP hard problems using many repeats of short runs. This strategy scales better than quantum search, by exploiting the correlations in the problem Hamiltonian. The energy conservation mechanism identified in Chapter 3 explains how energy conserving QWs can find lower energy states with better than guessing probability. In section 5.3, I generalised the energy conservation mechanism to an *energy redistribution mechanism* that holds for all monotonic quenches which start in the ground state of the driver Hamiltonian and have non-negative control functions. This includes a wide range of QA protocols used in both theoretical and experimental work. The improvements gained by using time-varying rapid quench schedules can be considerable, as illustrated in section 5.2.

To generate significant energy redistribution, there needs to be significant dynamics driving the system away from the initial state. To characterise the dynamics, in section 5.4 I defined

the average local dynamic coefficient that balances the contributions from both the driver and problem Hamiltonians. This allows the control functions in the Hamiltonian to be optimised for fast dynamics, and provides a very general way to estimate good values to use for specific problems. For the Sherrington-Kirkpatrick (SK) spin glass data ([Chancellor et al., 2019](#)), I showed in Fig. 5.11 that such estimates are almost as good as the numerically optimised values used in Chapter 3. I also verified in subsection 5.4.5 that the average local dynamic coefficient correctly predicts that the search problem will not have significant dynamics on short timescales. The average dynamic coefficient have defined in this chapter is just one way to capture the local dynamics in a QA Hamiltonian system; there are likely other formulations that would serve equally well. In the transverse Ising setting, it focuses on single spin flips, which intuitively are likely to provide the fastest dynamics. Settings with driver Hamiltonians applying multiple spin flips (e.g., ([Roland and Cerf, 2002](#); [Childs and Goldstone, 2004](#))) may prove less favorable for obtaining fast dynamics, a worthwhile direction for future investigations.

Taken together, the *energy redistribution mechanism* and the *average dynamic coefficient* are powerful tools for understanding, designing, and optimally controlling rapid quench QA algorithms. In subsection 5.5.2, I provide a simple example of how to do this to good effect for annealing schedules, and in subsection 5.5.1 I verify that it is both efficient and effective for estimating hopping rates for QWs on spin glasses. While adiabatic quantum computing (AQC) and QWs for unstructured search have long had theoretical underpinnings, this represents a significant step in understanding how to exploit QA schedules run for short times. For current state-of-the-art noisy quantum computers, short run times are a big advantage over the long coherence times required for AQC, or QW on unstructured search.

I have shown that these tools apply to the biased drivers proposed in [Duan et al. \(2013\)](#); [Graß \(2019\)](#), which provide a method of incorporating prior information into annealing schedules. This can produce significant improvements, as illustrated in subsection 5.2.2. On the other hand, reverse annealing schedules, both as proposed by [Perdomo-Ortiz et al. \(2011\)](#); [Ohkuwa et al. \(2018\)](#); [Yamashiro et al. \(2019\)](#) and discussed in [Chancellor \(2017\)](#), and as implemented in the latest D-Wave Systems ([D-Wave Systems Inc., 2019b](#)), are by definition not monotonic, so the tools and mechanisms identified here cannot be applied. Since reverse annealing is a powerful tool, extending these results to non-monotonic cases is a worthwhile direction for further research.

Chapter 6

Applying quantum annealing to the shortest vector problem

6.1 Introduction

As quantum computing (QC) hardware develops and becomes useable on a larger scale, classical cryptography will be a victim of the exponential speed-up for integer factorisation and discrete logarithm computation provided by Shor’s algorithm. This is because the security of public key cryptography requires the existence of a problem that is intractable without a particular piece of information (known as a *key*), but can be solved efficiently by a user who possesses the key. An example of a problem like this is the factorisation of a large integer $n = pq$ with only two non-trivial prime factors p and q . If one of these factors is known, the integer n can be efficiently divided by that factor to find the other, but if neither factor is known, then a much more computationally difficult approach is necessary, such as applying a number field sieve ([Lenstra and Hendrik Jr, 1993](#)), requiring super-polynomial time at best.

The hardness of integer factorization is core to the security of the RSA cryptosystem ([Rivest et al., 1978](#)), while related problems (which have also turned out to be efficiently solvable in the QC paradigm) provide the security for other schemes such as Diffie-Hellman key exchange, and ElGamal ([Diffie and Hellman, 1976](#); [Bernstein et al., 2009](#)). Thus, new cryptographic techniques will be required to keep communication and information storage secure as useful quantum computing develops. This new field is known as post-quantum cryptography (PQC).

In (Bernstein et al., 2009), five families of problems are outlined that form the focus of the search for quantum-safe cryptographic schemes. Of these families, a particularly promising one is lattice-based cryptography (LBC). Twelve of the 26 candidate systems being analysed in round two of the NIST Post-Quantum Cryptography Standardization process (NIST, 2017) are based on hard lattice-based problems.

Well known examples of LBC schemes include learning with errors (LWE) problem (Regev, 2005), and its variants. LWE has also been used as the basis for a fully homomorphic encryption scheme (FHE) (Gentry, 2009). Other important LBC schemes include the N th-degree Truncated-polynomial Ring Units (NTRU) scheme (Jeffrey et al., 1998) and the Goldreich-Goldwasser-Halevi (GGH) scheme (Goldreich et al., 1997). More broadly, the problems in LBC typically revolve around minimisation of distances in high-dimensional spaces.

Two common classical approaches to solving lattice problems are enumeration and sieving. In sieving, vectors are drawn from some distribution over the lattice and iteratively combined to produce short vectors non-deterministically (Ajtai et al., 2001). In enumeration, all the vectors within a ball of a certain radius around the origin are computed, which, with the right parameters, can guarantee an optimal solution (Fincke and Pohst, 1985; Kannan, 1983). Recently (Gama et al., 2010) significant speed-ups have been found by using non-deterministic forms of enumeration, using a technique known as ‘extreme pruning’.

To date, quantum algorithms for lattice problems have tended to combine pre-existing gate model quantum algorithms and classical lattice algorithms to provide a quantum speedup. Grover’s algorithm in the gate model has been combined with classical sieving approaches to achieve speedups in which the exponent is reduced by around 25% (Laarhoven et al., 2013). A quadratic speed-up has been obtained (Aono et al., 2018) by applying quantum tree algorithms (Montanaro, 2020, 2018) to classical enumeration approaches.

This chapter will focus on an LBC problem known as the Shortest Vector Problem (SVP) which involves searching for the shortest vector in an integer lattice for which a ‘bad’ basis (long vectors that are far from being orthogonal) is given as the input. The precise definition for this problem is given in subsection 2.1.5. A new approach to solving the SVP problem was introduced in Joseph et al. (2020). In that approach, short vectors are found by encoding SVP into a Hamiltonian of a system of ultra-cold bosons trapped in a potential landscape, and performing rapid quenches in quantum annealing (QA). In this chapter, a similar rapid quench

QA approach is considered in the Ising model setting set out in section 2.2.

The chapter is structured as follows: in section 6.2, two different mappings of SVP into the Ising model are given, along with proofs of their asymptotic space requirements. In section 6.3, QA on SVP using these two different problem mappings is studied, both numerically in the ideal closed-system setting and experimentally on the D-Wave quantum annealer (McGeoch et al., 2019), providing a fully quantum analysis of lattice problems in up to 7-dimensional instances utilising 56 logical qubits (over 1000 physical qubits). Finally, the chapter is summarised in section 6.4.

6.2 Ising model mappings for SVP

In this section, two encodings of SVP into the Ising coefficients J_{ij}, h_i from Eq. (2.4) are presented. The coefficients are derived directly from the input basis $\mathbf{B}$, following a similar process as for mapping of SVP to the Bose-Hubbard model presented in Joseph et al. (2020).

Any lattice point can be written as the vector

$$\mathbf{v} = \mathbf{x} \cdot \mathbf{B} = x_1 \mathbf{b}_1 + \dots + x_d \mathbf{b}_d. \quad (6.1)$$

The square of the length of the vector $\mathbf{v}$ can then be written

$$\begin{aligned} \|\mathbf{v}\|^2 &= \mathbf{v} \cdot \mathbf{v} \\ &= \sum_{j,k} x_j x_k \mathbf{b}_j \cdot \mathbf{b}_k. \end{aligned} \quad (6.2)$$

The aim is to find the integer coefficients $x_1, x_2, \dots, x_d$ that minimise $\|\mathbf{v}\|^2$ (for nonzero $\mathbf{v}$, given the fixed scalar values $\mathbf{b}_i \cdot \mathbf{b}_j$ which are the problem input).

To see how SVP can be mapped to the Ising model, first consider an Ising-like device consisting of qudits rather than qubits; that is, instead of a computational basis consisting of the eigenstates of Pauli Z operators, the computational basis consists of eigenstates of qudit operators $\hat{Q}$ with eigenvalues $-q, -q+1, \dots, q-1, q$ (how large q needs to be to ensure the shortest vector is represented in the Hilbert space will be dealt with shortly). With this device, SVP could be

mapped to the problem Hamiltonian

$$\hat{H}_{\text{SVP}} = \sum_{j,k}^d \hat{Q}^{(j)} \hat{Q}^{(k)} \mathbf{G}_{jk}, \quad (6.3)$$

where $\mathbf{G}_{jk} = \mathbf{b}_j \cdot \mathbf{b}_k$ is the (j, k) th element of the Gram matrix for the input basis. The problem Hamiltonian $\hat{H}_{\text{SVP}}$ has eigenstates which correspond to vectors in the lattice and eigenvalues which correspond to the squared lengths of those vectors. The ground-state thus corresponds to the trivial zero-vector $\mathbf{0}$, while the shortest non-zero vector $\mathbf{v}_{\text{shortest}}$ is represented by the first-excited manifold. This manifold will usually be at least two-fold, due to the fact that reflecting any vector about the origin produces another vector of identical length. Atypically for continuous-time quantum computing (CTQC) algorithms, it will be this first-excited manifold that will be targeted.

6.2.1 Size of qudit operators

In this subsection, I will address the question of how big the qudit operators $\hat{Q}$ need to be to ensure that at least one shortest eigenvector is expressed by eigenstates of the problem Hamiltonian $\hat{H}_{\text{SVP}}$. This will in general depend on the nature of the input basis. Here, I will consider a specific type of basis known as Hermite Normal Form (HNF). HNF bases are of particular interest because of three reasons:

1. They can be described (and therefore communicated) efficiently;
2. Each lattice has a unique HNF;
3. The HNF of a lattice can be efficiently computed from any provided basis.

A basis for a full-rank lattice is in HNF if, when arranged such that the j th basis vector is the j th column of a matrix $\mathbf{H}$, the matrix $\mathbf{H}$ has the following properties:

- $\mathbf{H}_{jk} = 0$ for $j > k$;
- The first nonzero term from the left (the pivot) is positive and strictly to the right of the first nonzero term of the row above;

- Elements directly below the pivot are zero and those directly above are reduced modulo the pivot.

The HNF basis of a lattice is described as *optimal* if there exists only one non-unit pivot. This means (for a row basis HNF) that there is only one column of non-zero values. Such a basis looks as follows:

$$\begin{bmatrix} \mathbf{b}_1 \\ \mathbf{b}_2 \\ \vdots \\ \mathbf{b}_d \end{bmatrix} = \begin{bmatrix} 1 & & b_1 \\ & \ddots & \vdots \\ & & 1 & b_{d-1} \\ & & & D \end{bmatrix}. \quad (6.4)$$

While not every lattice has an HNF that is optimal, it has been shown ([Rose et al., 2011](#)) that a randomly generated lattice has a probability ~ 0.435 of having an optimal HNF. Furthermore, optimal HNF bases are considerably more efficient to communicate than general bases and even general HNF bases because they can be communicated by sharing a single vector of length d . For these two reasons, it is reasonable to expect SVP-based cryptographic protocols to be based on the optimal HNF, and there is precedent for this ([Plantard et al., 2008](#)). For the analysis in this section, optimal HNF will be assumed, but it may be possible to perform similar analysis on other well-specified basis forms.

The form of a general lattice vector $\mathbf{v}$, written as a linear combination of the basis vectors, is

$$\begin{aligned} \mathbf{v} &= \begin{bmatrix} x_1 & \dots & x_d \end{bmatrix} \cdot \begin{bmatrix} 1 & & b_1 \\ & \ddots & \vdots \\ & & 1 & b_{d-1} \\ & & & D \end{bmatrix} = \\ &\quad \begin{bmatrix} x_1, & \dots, & x_{N-1}, & (x_1 b_1 + \dots + x_d D) \end{bmatrix}. \end{aligned} \quad (6.5)$$

Minkowski's theorem ([Minkowski, 2016](#)) provides the bound

$$||\mathbf{v}_{\text{shortest}}|| \leq \sqrt{d} \cdot D^{1/d} \quad (6.6)$$

on the length of the shortest vector $\mathbf{v}_{\text{shortest}}$.

Consider the coefficients on the right hand side of Eq (6.5). Minkowski's bound prescribes a

sphere about the origin, with radius equal to the bound in Eq. (6.6), in which to search for the shortest vector. Relaxing this constraint slightly, and applying this weaker version to the coefficients on the right hand side of Eq. (6.5), it can be asserted that every coordinate x_i of a must be less than or equal to $\sqrt{d} \cdot D^{1/d}$, which now prescribes a (larger) d -cube around the origin in which to search.

It follows that

$$|x_j|, \dots, |x_{d-1}| \leq \sqrt{d} \cdot D^{1/d}. \quad (6.7)$$

For the final component, it is the case that

$$|x_1 b_1 + \dots + x_d D| \leq \sqrt{d} \cdot D^{1/d}. \quad (6.8)$$

Since it must be true that

$$|x_d D| \leq \|\mathbf{v}_{\text{shortest}}\| + |x_1 b_1 + x_2 b_2 + \dots + x_d D|, \quad (6.9)$$

it follows that

$$\begin{aligned} |x_d D| &\leq \sqrt{d} \cdot D^{1/d} + \\ &\quad |x_1 b_1 + \dots + x_{d-1} b_{d-1}|. \end{aligned} \quad (6.10)$$

Applying the triangle inequality gives

$$\begin{aligned} |x_d D| &\leq |x_1 b_1| + \dots + |x_{d-1} b_{d-1}| + \\ &\quad \sqrt{d} \cdot D^{1/d}, \end{aligned} \quad (6.11)$$

and because the b_i are reduced modulo D ,

$$\begin{aligned} |x_d D| &\leq |x_1 D| + \dots + |x_{d-1} D| + \\ &\quad |\sqrt{d} \cdot D^{1/d}|. \end{aligned} \quad (6.12)$$

Then by applying Eq. (6.7)

$$|x_d D| \leq (d-1)\sqrt{d} \cdot D^{1/d} \cdot D +$$

$$\begin{aligned} & \sqrt{d} \cdot D^{1/d} \\ & \leq d^{3/2} \cdot D^{1+1/d}. \end{aligned} \quad (6.13)$$

Thus, the size of the final component x_d of the shortest vector $\mathbf{v}_{\text{shortest}}$ when expressed in the optimal HNF basis is bounded as

$$|x_d| \leq d^{3/2} \cdot D^{1/d}. \quad (6.14)$$

If the qudit operators $\hat{Q}_1, \dots, \hat{Q}_d$ are set up such that their minimum and maximum eigenvalues satisfy Eq. (6.7) and Eq. (6.14), then it is guaranteed that at least one shortest vector is expressed by the eigenstates of the problem Hamiltonian $\hat{H}_{\text{SVP}}$.

In the following two subsections, two different ways to encode the qudit operators $\hat{Q}$ into the Ising model $\hat{Z}$ operators will be described. The results in Eq. (6.7) and Eq. (6.14) will be used to calculate bounds on how many physical qubits will be needed to ensure that at least one shortest vector is expressed by the eigenstates of the problem Hamiltonian $\hat{H}_{\text{SVP}}$.

6.2.2 Hamming-weight-encoded Qudits

In this encoding, the qudit operators are defined by a simple sum of qubit operators

$$\hat{Q}_{\text{Ham}}^{(j)} = \sum_{p=1}^{M_j} \frac{\hat{Z}_{pj}}{2}. \quad (6.15)$$

Each computational basis state is assigned an eigenvalue that counts the number of qubits in the the +1 state, its *Hamming weight*, shifted so that the possible values are symmetric about zero.

This qudit mapping is extremely simple and is not optimal in terms of space. This is because it leads to redundancies, with multiple spin configurations corresponding to the same qudit value in $[-M/2, M/2]$. Looking to Eq. (6.7), each qudit operator $\hat{Q}^{(j)}$ for $j = 1, \dots, d-1$ requires $M_j \leq \sqrt{d} \cdot D^{1/d}$ qubits, while by Eq. (6.14) the final qudit operator $\hat{Q}^{(d)}$ requires $M_d \leq d^{3/2} \cdot D^{1/d}$ qubits. Thus, the total number of required qubits satisfies the bound

$$n \leq 2d^{3/2} \cdot D^{1/d} \quad (6.16)$$

For the rest of the chapter, the term ‘Ham’ will refer the Hamming-weight encoding, or the QA SVP algorithm using this encoding.

6.2.3 Binary-encoded qudits

In this encoding, the qudit operators are defined such that their eigenstates are indexed by the integers expressed in binary by the states of the constituent qubits. An (unsigned) qudit operator (acting on the j^{th} qudit) then takes the form

$$\hat{Q}_{\text{unsigned}}^{(j)} = \sum_{p=0}^{k-1} 2^p \left(\frac{\mathbb{1} - \hat{Z}_{pj}}{2} \right), \quad (6.17)$$

the eigenvalues of which are positive integers integer in the range $[0, 2^k - 1]$. However, for the purposes of solving SVP, a range that is approximately symmetric about 0 is needed, for example, $[-2^k, 2^k - 1]$, and so a signed integer is required. To achieve this, introduce an extra qubit (doubling the range of eigenvalues) and shift downwards by 2^k :

$$\begin{aligned} \hat{Q}_{\text{Bin}}^{(j)} &= \sum_{p=0}^k 2^p \left(\frac{\mathbb{1} - \hat{Z}_{pj}}{2} \right) - 2^k \mathbb{1} \\ &= - \sum_{p=0}^k 2^{p-1} \hat{Z}_{pj} - \frac{1}{2} \mathbb{1}. \end{aligned} \quad (6.18)$$

This encoding is maximally efficient in space as each Ising spin configuration corresponds to a distinct vector in the lattice. This efficiency, however, is at the cost of requiring high quality qubits, as will be discussed further in section 6.3.

Looking to Eq. (6.7), each qudit operator $\hat{Q}^{(j)}$ for $j = 1, \dots, d-1$ requires $2^k \leq \sqrt{d} \cdot D^{1/d} + 1$, or

$$k \leq \log_2 \left(\sqrt{d} \cdot D^{1/d} + 1 \right) \quad (6.19)$$

qubits, while by a similar calculation based on Eq. (6.14), the final qudit operator $\hat{Q}^{(d)}$ requires

$$k \leq \log_2 \left(d^{3/2} \cdot D^{1/d} + 1 \right). \quad (6.20)$$

Thus, the total number of qubits required satisfies the bound

$$\begin{aligned} n &\leq (d-1) \log_2 \left(\sqrt{d} \cdot D^{1/d} + 1 \right) + \log_2 \left(d^{3/2} \cdot D^{1/d} + 1 \right) + d \\ &= O \left(d \log_2 \left(\sqrt{d} \cdot D^{1/d} \right) \right) \end{aligned} \quad (6.21)$$

For the rest of the chapter, the term ‘Bin’ will refer the binary encoding, or the QA SVP algorithm using this encoding.

The bounds given Eq. (6.16) and Eq. (6.21) are upper bounds that guarantee the existence of a vector with length $\|\mathbf{v}_{\text{shortest}}\|$ (a shortest vector) within the Hilbert space explored by the quantum Ising algorithms described in subsections 6.2.3 and 6.2.2. In practice, an attacker may choose to reduce the complexity of the system by reducing the number of qubits used for each qudit in order to more easily find ‘short’ vectors, at the expense of the assurance of being able to find a shortest vector.

It is also the case that Eq. (6.16) and Eq. (6.21) give upper bounds for HNF input bases. Using different, (classically) better-reduced bases is also likely to require lower numbers of qubits and so possibly yield better results, but giving a bound for these is not straightforward.

6.3 Results

To analyse the performance of the two algorithms described in section 6.2, Ham and Bin were implemented on the D-Wave 2000Q quantum annealer, to explore what is possible with the current state of QA hardware. Numerical simulations of ideal, closed-system versions of these algorithms were also performed to give an indication of what may be possible in the future. This numerical analysis was limited to smaller experiment sizes due to the computational constraints of simulating quantum systems on classical hardware.

To perform both the numerical and experimental analyses, a collection of random full-rank integer lattices was generated, and a ‘bad’ input basis was obtained by post-multiplying with randomly generated unimodular matrices. These bad bases were not in HNF, and so the upper bounds given in Eq. (6.16) and Eq. (6.21) do not apply. Instead, the numbers of qubits used to construct each qudit were chosen so that each qudit operator $\hat{Q}$ has eigenvalues in the ranges

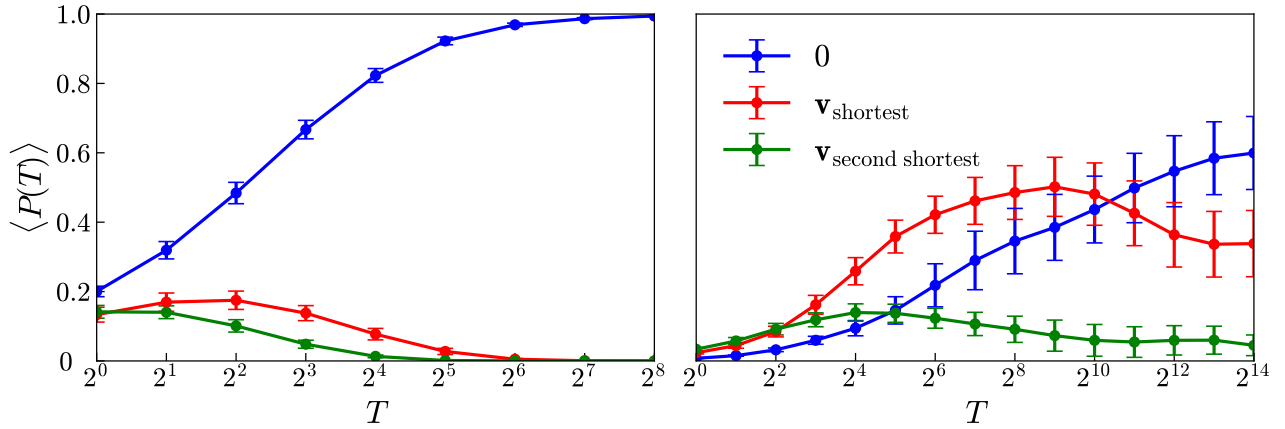


Figure 6.1: Numerical experiments carried out on 20 three-dimensional lattices. The lines show mean final closed-system probabilities for measuring the system to be in the ground (blue), first (red) and second (green) excited eigenstates (grouped by degeneracy) at completion for sweep lengths T , increasing in powers of 2, for the quantum Ising algorithm implemented with (left) Hamming-weight-encoded qudits and (right) binary-encoded qudits (Bin). The Ham algorithm nears the adiabatic limit at $T = 2^8$, after the shortest vector success probability peak at around $T = 2^2$, which is considerably lower than the peak time for Bin. The Bin algorithm nears the adiabatic limit past the rightmost data point but not before an encouraging peak in shortest vector success probability at $T = 2^9$. Standard errors across the 20 instances are shown in both plots by vertical error bars.

$[-4, 4]$ and $[-4, 3]$ for Ham and Bin respectively. This choice was made in order to improve the overall results by a range of metrics discussed in this section, while also fixing a degree of freedom to make the analysis easier to perform.

This means that it is not guaranteed that a shortest vector (of length $\|\mathbf{v}_{\text{shortest}}\|$) is represented in the Hilbert space for every instance. In fact, a shortest vector is represented in all of the three-dimensional lattice instances used, and approximately half of the instances in each of the higher dimensions. This is reflected in the results presented in this section for finding a shortest vector; no post-selection was performed to choose instances where shortest vectors are obtainable with coefficients in the specified range of qudit eigenvalues.

6.3.1 Numerical Simulation

Numerical simulations have been performed for both the Ham and Bin algorithms for three-dimensional lattices for a range of sweep times to investigate what performance could be expected in the limit of perfect QA hardware.

The numerical simulations in Fig. 6.1 demonstrate that, in the limit of fast sweeps, the outputs are of little use: their distribution will be close to uniform across all eigenstates. In the adiabatic limit, the ground state is observed with certainty. As the ground-state is the trivial zero-vector $\mathbf{0}$, this is of similarly little use. However, both sets of results also demonstrate a ‘Goldilocks zone’, similar to that first identified for SVP in a Bose-Hubbard model QA setting in Joseph et al. (2020). When a sweep time T is picked that is long enough that (at completion) a significant proportion of probability density is concentrated on low-energy eigenstates, but short enough that it is not adiabatic (with probability density concentrated on the ground state), eigenstates corresponding to short vectors are observed with good probability. Both versions, Ham and Bin, of the Ising SVP algorithm suggest that sweeps which are neither ‘too slow’ nor ‘too fast’ are optimal for achieving short but non-zero lattice vectors.

This observation is fully consistent with the adiabatic theorem. If a sweep time T is picked that is short compared to many of the gaps in the Hamiltonian, most of the amplitude will quickly escape to highly excited states leading to a low success probability. In the opposite extreme, if a sweep time T is picked that is long compared to the gap between the ground state and the first excited state, the evolution will be close to adiabatic and most of the amplitude will remain in the ground-state throughout the sweep, again leading to a low success probability. However, when a sweep time T is picked that is between these extremes, much of the amplitude will escape the ground-state but will remain in low-energy states resulting in the algorithm returning short vectors (including the shortest vector) with non-trivial probability.

The superior solution probabilities demonstrated by Bin can also be explained in terms of the adiabatic theorem. For Bin, the minimum gap $\min_t(\Delta E)$ between the ground-state and first excited can become extremely small due to the large range of energy scales present in the qubit coupling coefficients. For Ham, the minimum gap is typically much larger (see Fig. 6.2 for a typical example) than for Bin, meaning that even moderately short sweeps leave much of the amplitude in the uninteresting ground-state. This shows that, in the long term, with access to high fidelity qubits and low noise environments, Bin may perform better, and with lower space requirements.

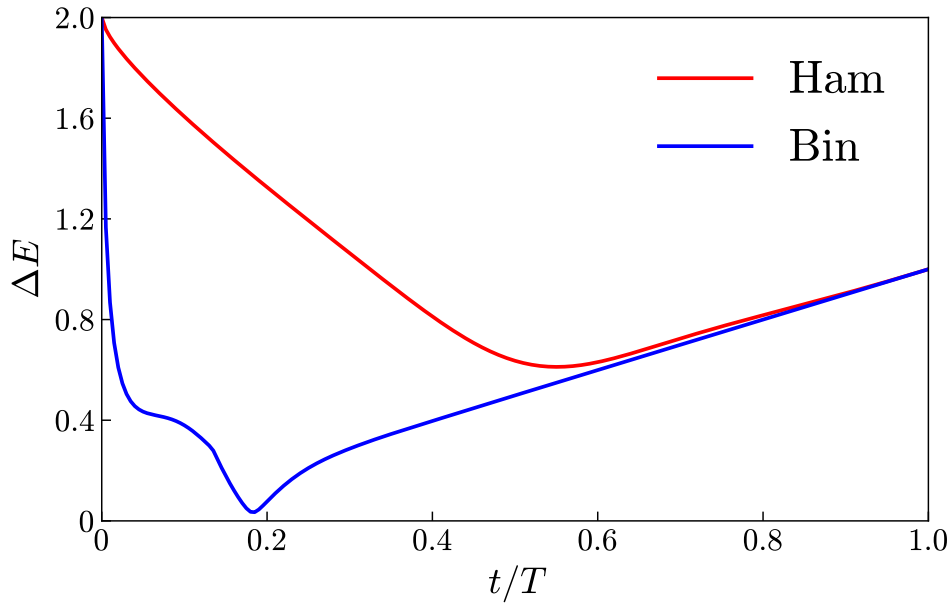


Figure 6.2: Energy gap ΔE between the ground-state and the first excited state at each time t along a linear sweep of total duration T for the Ham algorithm (red) and the Bin algorithm (blue). The minimum gap occurs much earlier for Bin and is much smaller.

6.3.2 D-Wave Quantum Annealer

The algorithms Ham and Bin, due to their Ising formulation, can be performed on the D-Wave 2000Q quantum annealer (McGeoch et al., 2019). In this subsection, the performance of the algorithms presented in this chapter are examined experimentally using the D-Wave quantum processor. These experiments were performed via the D-Wave Ocean SDK (D-Wave, 2017–) in Python.

The Ising model encodings described in this chapter assume full qubit-qubit connectivity, but the D-Wave 2000Q qubits are only connected according to a Chimera topology, in which each qubit is connected to a handful of nearby qubits that form a cluster, and each cluster is connected to only a few others. Such clusters are represented as the diamond formation of dots in Fig. 6.3. A problem Hamiltonian requiring full connectivity can be embedded into D-Wave’s Chimera topology, but this incurs an overhead in the number of qubits required. This embedding is done by creating qubit ‘chains’, which are illustrated in Fig. 6.3. In a qubit chain, strong (compared to interactions between different chains) qubit-qubit ferromagnetic interactions energetically incentivise each qubit to align in the same direction. When an error occurs such that not all qubits in a chain return the same value (called a chain break), a majority vote is taken to decide on the final value.

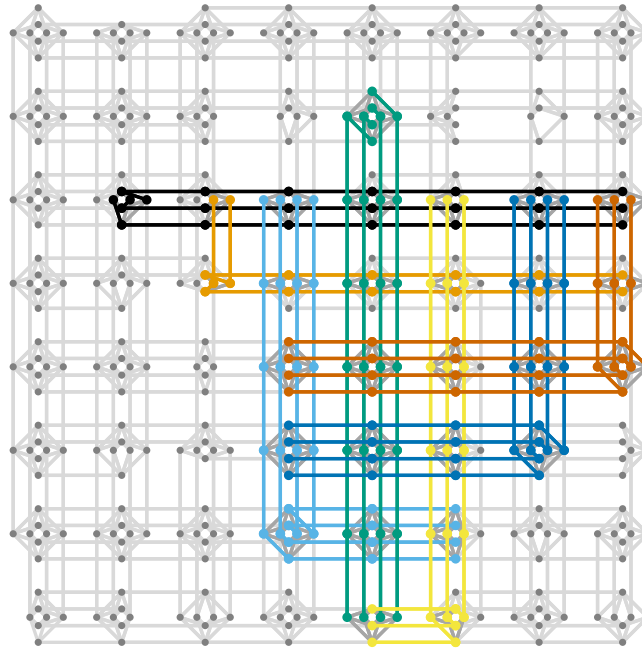


Figure 6.3: Diagram, from [Boothby et al. \(2016\)](#) showing logical qubits embedded into the Chimera topology as chains of physical qubits. The coloured lines indicate which qubits form the different chains, connected by strong ferromagnetic interactions.

For the experiments presented in this subsection, the default embedding provided by the API (based on the heuristic techniques described in [Cai et al. \(2014\)](#)) has been used to map from the fully connected qubit graph specified by the theoretical model described in section 6.2 (the logical qubits), and the QPU qubit graph (the physical qubits). This embedding incurs a roughly quadratic overhead in the number of physical qubits required, resulting in a maximum system size for the experiments listed of 56 logical qubits, which maps to over 1000 physical qubits. To reduce the number of chain breaks during the anneal, a chain coupling strength that is large compared to the largest couplings in the problem itself is desirable. This is achieved by setting the chain strength to its maximum possible value on the device, and scaling down the problem couplings. However, problem couplings that are too small can lead to worse solutions, due to the finite control precision and the finite device temperature. For the experiments in this chapter, the problems were scaled down so that the maximum problem coupling strengths were set to half the chain coupling strengths. A more systematic optimization might have lead to better results.

The experiments in this chapter were performed using various sweep durations, but it was seen that the duration had no systematic impact on the quality of the results, suggesting the dominance of thermal effects over quantum effects at even the shortest times. For the data

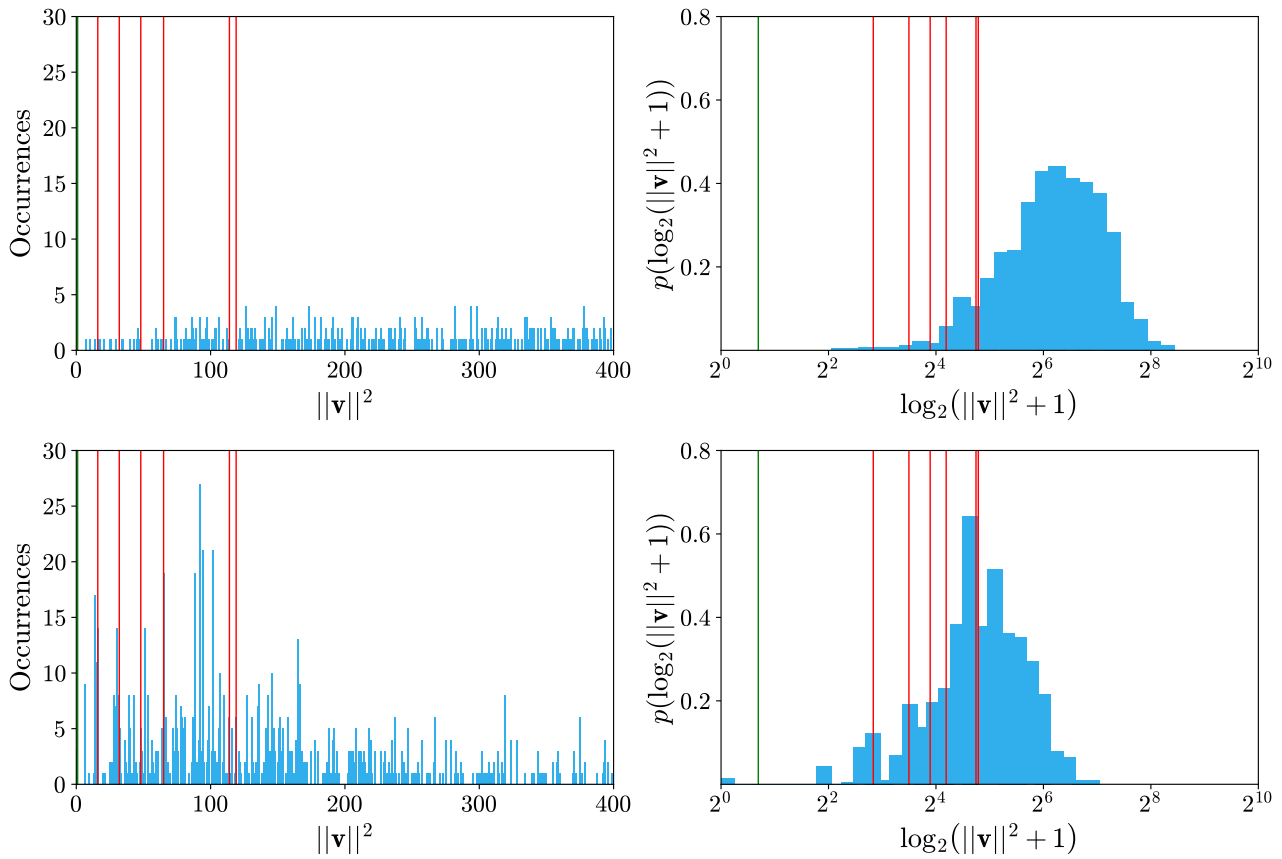


Figure 6.4: Illustrative results from D-Wave for Bin (top) and Ham (bottom) QA algorithms with histograms showing how the squares of the vector lengths returned by 900 runs with sweep duration $T = 64 \mu\text{s}$ on the device are distributed. The plots on the left show occurrences for each squared vector length (no binning), while the plots on the right show an inferred probability distribution (probability density $p(\log_2(\|\mathbf{v}\|^2 + 1))$) over the natural logarithm of squared vector lengths from the same data (with binning). The right hand plots show the same data as the left hand plots, but allow an easier comparison at a glance. The vertical green lines show where $\|\mathbf{v}_{\text{shortest}}\|^2$ lies, and the vertical red lines show the squared lengths of the input basis vectors. The heights of the green and red lines have no significance.

shown in 6.4, an arbitrarily chosen sweep time of $T = 64 \mu\text{s}$ was used, while for the data in 6.5, the minimum possible sweep time on the device of $T = 1 \mu\text{s}$ was used; no systematic optimization was performed.

Furthermore, the API offers an optimization feature known as ‘spin-reversal transforms’, which essentially randomizes the sign on the Pauli-Z operators in the problem (and changing any relevant signs on the problem fields and couplings to compensate) for each run. This is done in order to mitigate the biasing effects introduced by the implementation of the different coupling signs between the underlying superconducting qubits. This feature was not used for the experiments presented in this chapter; doing so may have produced better results.

Fig. 6.4 shows histograms of the squares of vector lengths returned in an experiment consisting of 900 runs of each algorithm on the same 6-dimensional lattice, all with a sweep duration $T = 64 \mu\text{s}$. The left and right plots display the same information, but on the right the plots show an inferred probability distribution (probability density $p(\log_2(\|\mathbf{v}\|^2 + 1))$) over the natural logarithm of squared vector lengths. The top row shows the data from Bin on a representative six-dimensional lattice example, and the bottom row shows the data from Ham on the same example.

In this case, Ham appears to return shorter vectors and higher rates of occurrence, with some coefficient vectors occurring over 25 times, whereas Bin only records a maximum of 4 occurrences for any combination. It should also be noted that Ham returns many more vectors in the ‘somewhat useful’ range (any nonzero vector shorter than the longest input-basis vector). In Fig. 6.4, this region lies between the green marker ($\|\mathbf{v}_{\text{shortest}}\|^2, \log \|\mathbf{v}_{\text{shortest}}\|^2$) for left and right plots respectively) and the furthest right red marker ($\max_i \{\|\mathbf{b}_i\|^2\}, \max_i \{\log(\|\mathbf{b}_i\|^2)\}$ for left and right plots respectively).

Fig. 6.4 helps to understand what the QA results look like for an individual example, as well as to qualitatively compare Ham and Bin. For more detailed performance analysis, other visualisations of results are appropriate, such as in Fig. 6.5.

Fig. 6.5 shows a summary of results obtained for 19 lattices in dimensions $d = \{3, 4, 5, 6, 7\}$, according to four key figures of merit (from left to right). Each lattice was run with a sweep duration of $1 \mu\text{s}$ 900 times, and a measurement was taken at the end of each sweep, producing 900 output vectors. In Fig. 6.5, each data point shows the probability of achieving a certain figure of merit, averaged over the 19 lattices. The red (green) lines show the performance of Ham (Bin), and baselines from uniform random sampling of the Hilbert spaces are shown in black solid and dotted lines (Ham and Bin have different, but asymptotically close, baselines due to the slight difference qudit operator eigenvalue range). The red lines sit higher than the green lines, indicating higher success probabilities Ham on the QA hardware.

The top-left plot shows the probability of the annealer returning the zero vector $\mathbf{0}$. This is not a directly useful figure of merit, because the zero vector is of no use computationally. However, the probability of returning $\mathbf{0}$ can be taken as a measure of adiabaticity. From this point of view, it can be seen that Ham is much more likely than Bin to return ground states on the noisy quantum annealing hardware.

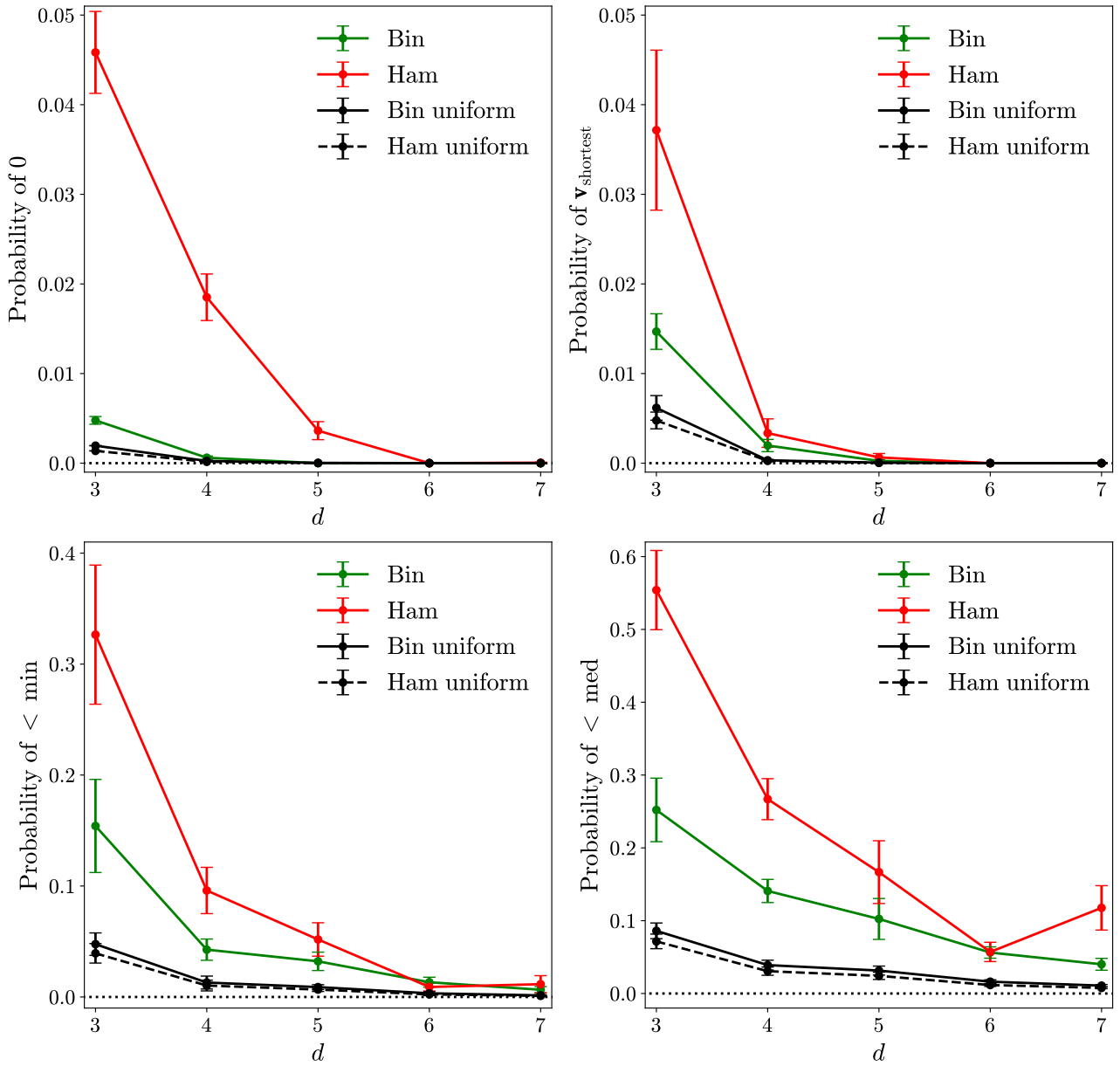


Figure 6.5: $T = 1 \mu\text{s}$ for Ham (red) and Bin (green) algorithms performed on 19 randomly generated ‘bad’ lattice bases in dimensions $d = 3, 4, 5, 6, 7$, for 4 different figures of merit: ground state (top-left, not desired), shortest vector (top-right, preferable), shorter than the minimum length input basis vector (bottom-left), and shorter than the median length input basis vector (bottom-right). The dotted black line indicates zero probability and is there to guide the eye. The other black lines are baselines indicating the results from uniform random sampling of coefficient vectors represented by the mapping Hilbert (dashed for Ham, solid for Bin), i.e. the proportion of lattice vectors that satisfy the figure of merit.

The top-right plot shows the probability of obtaining a shortest vector (of length $\|\mathbf{v}_{\text{shortest}}\|$), which is the most important figure of merit. It is interesting to observe that Ham and Bin perform remarkably similarly, (with Ham slightly higher than Bin). As expected, probabilities for both $\mathbf{0}$ and $\mathbf{v}_{\text{shortest}}$ decay as lattice dimension d (and Hilbert space size) increases. To verify the rate of decay would require quantum hardware that is beyond present capabilities.

The bottom-left plot gives the probability observed of returned vectors having length shorter than the shortest input basis vector, and the bottom-right plot gives the probability of vectors having length shorter than the median input basis vector. Both of these figures of merit provide the opportunity to improve on the bad basis provided as input to the algorithm. The basis vectors themselves are trivial results (given by vectors of the form $(0, \dots, 0, 1, 0, \dots, 0)$), but only represent a very small proportion of the outputs. It is encouraging that the probabilities in (c) and (d) do not seem to tail off as for $\mathbf{0}$ and $\mathbf{v}_{\text{shortest}}$ at dimensions 6 and 7, and in fact appear to increase in (d), but it is unclear what will happen at larger sizes.

The fact that the Ham algorithm consistently outperforms the Bin algorithm on the D-Wave hardware, despite the larger system size required to search the same solution space, can be partially understood by considering the energy scales of the J and h coefficients in the problem Hamiltonian $\hat{H}_{\text{SVP}}$ in each case. In the Bin encoding, the strength of qubit-qubit interactions span multiple orders of magnitude, due to the varying significance of each qubit in the binary representations they form. This large range means that some of the coefficients are small when the whole problem is scaled to fit within the range of possible settings on the hardware. If some of these coefficients become small compared to the errors in the interaction energies inherent to the device, the QPU is essentially asked to solve *the wrong problem* (Young et al., 2013), leading to poorer performance.

In Ham, however, each qubit has equal significance in the qudit it forms part of, and so the required range only depends on the actual SVP problem instance. This effect may be quite significant, but it will be tempered by the effects of having a larger system size: longer chains are required, which means chain-breaks are more likely, leading to more errors

Returning to Fig. 6.5, it is not surprising that the probability of obtaining $\mathbf{v}_{\text{shortest}}$ tends to zero as dimension increases, but it is how fast this occurs that is important. A polynomial decay would be catastrophic for LBC, whereas an exponential decay could even match the current state of the art for classical lattice algorithms. At this stage, not enough data points

are available to heuristically estimate the rate of decay. A key observation to note is that Ham outperforms Bin at almost every data point across the four subplots, demonstrating a significant improvement on Bin for the current regime of QA.

6.4 Chapter summary

In this chapter, I introduced a pair of novel mappings of the shortest vector problem (SVP) from lattice-based, post-quantum cryptography into the Ising model, enabling the study of Ising model continuous-time quantum computing (CTQC) on SVP. This helps to develop the quantum annealing (QA) vector optimisation techniques proposed in [Joseph et al. \(2020\)](#), which considered the Bose-Hubbard model setting instead of the Ising model. The resulting problem Hamiltonians differ from those usually used in CTQC (including the others presented in this thesis) in that it's the first-excited state that corresponds to the problem solution, rather than the ground-state as is typical. This means that performing AQC by satisfying the adiabatic theorem with long-time scales to preserve the system in its ground state throughout the evolution would actually be counter-productive, in addition to being experimentally challenging. As such, the shorter run-times of QA particularly appropriate for this problem.

Through the numerical application of ideal, closed-system QA on the smallest examples (three-dimensional lattices), all the way from the shortest quenches up to long adiabatic timescales, I identified this 'Goldilocks zone', on much shorter timescales than the adiabatic limit. This is consistent with the results in [Joseph et al. \(2020\)](#). Then, by examining the results of a series of experiments of QA on SVP on a D-Wave device, I was able to show that the algorithm performs non-trivially better than random sampling at larger sizes (up to seven-dimensions), although still small compared to lattices of practical interest. Unfortunately, as the D-Wave device operates in the open-systems regime, and because an additional layer of problem embedding is required to overcome the limited connectivity of the device, the numerical and experimental results are not directly comparable.

While the results in this chapter are not conclusive, they represent a novel analysis of lattice cryptanalysis on real quantum annealing hardware. The results suggest that quantum cryptanalysis is nearing experimental tractability, moving beyond the purely theoretical realm. The spatially efficient Bin encoding, coupled with the ever increasing pace of development in quan-

tum hardware, means that is plausible than in the near future much larger experiments can be carried out to put quantum lattice cryptanalysis algorithms to the test, whether that is on quantum annealers as described in this chapter, or on gate model devices. On the other hand, the Ham encoding requires many more qubits than the Bin encoding, but they do not need to be as high quality (and their interactions to not need to be set with such precision). This could make the Ham encoding more suitable for settings where lots of lower quality qubits are available.

Through the work presented in this chapter, it was not possible to extract asymptotic time scaling for these algorithms. Finding a way to do this represents an important direction for future research. Furthermore, while this chapter has focused on SVP, there is an approximate form, known as SVP_γ , which only requires an attacker to find a vector of length polynomially ($\text{poly}(d)$) larger than $\|\mathbf{v}_{\text{shortest}}\|$. This is likely to be achievable in an even shorter timescales than those found here for SVP, and represents another interesting direction for future research.

Chapter 7

Conclusion

7.1 Summary of Thesis Achievements

In this thesis, after defining and explaining continuous-time quantum computing (CTQC) in the Ising model in Chapter 2, I presented a number of novel numerical, analytical, and experimental results, which I will summarise here.

In Chapter 3, I numerically explored the application of the continuous-time quantum walk (QW) to optimization problems, using the NP-Hard (and uniformly hard) problem of finding the ground-state of a Sherrington-Kirkpatrick (SK) spin-glass Hamiltonian as a prototypical example. Compared with the well-studied QW algorithm for the Grover search problem, I discovered a number of key differences. First, rather than achieving $O(1)$ success probability but requiring a long runtime on each run, QWs on spin-glasses achieve only an exponentially-decreasing (in the number of spins) success probability, and thus require exponentially many repeats, but each repeat only requires a polynomially-long runtime, resulting in a runtime scaling that, while still exponential, is more favourable than that for Grover search. As long coherence times are currently experimentally difficult to achieve, this behaviour is much more suitable for near-term application. Additionally, rather than requiring the QW parameters be set with exponential precision, the QWs on spin-glasses are much less sensitive to the parameters, meaning it is possible to set them via well-motivated heuristics. Furthermore, I demonstrated that these differences are due to structure present in the spin-glass problem, carried over into the very natural Ising model mapping, that is absent in the Grover search.

Specifically, the difference is caused by the correlation between the energy (or cost) difference between candidate solutions and the bit-flip distance between them. Finally, I demonstrated analytically an ‘energy conservation’ mechanism that partially explains how QWs work by showing that the QW dynamics will always improve the output distribution.

In Chapter 4, I showed that the behaviour of QWs on the SK problem generalises to other optimization problems by applying many of the same numerical analyses to the well-known problem MAX2SAT. Furthermore, I examined the performance of adiabatic quantum computing (AQC) on MAX2SAT, offering some preliminary evidence that AQC outperforms QWs on this problem on average quite considerably. Finally, I described and explored a state-of-the-art classical algorithm and showed that this outperforms both QW and AQC quite considerably, illustrating the need to look beyond these two extremes and find smarter CTQC algorithms to beat what is classically possible. Throughout the chapter, by comparing QW, AQC, and the classical algorithm on a per-instance basis, including a set of instances provided by [Crosson et al. \(2014\)](#) that were data-mined to be hard for quantum annealing (QA), I showed that, while the performance of each is moderately well-correlated, the correlation is not perfect, meaning that there may be different types of instances for which different approaches work best.

In Chapter 5, I explored numerically and analytically the application of rapid quenches in QA. These differ from QWs by being time-dependent, but differ from AQC by operating far from the adiabatic regime, making them suitable for current and near term hardware. By extending the energy conservation mechanism from Chapter 3 to time-dependent schedules, I showed that rapid quench dynamics will, under a very general set of conditions, always improve the output distribution compared to the input state. Next, I developed a (heuristically-motivated but analytical) framework that in many cases can efficiently quantify the level of dynamics during a rapid quench schedule, which can be used to identify when a quench might lead to a significant improvement. Finally, I showed how this framework can often be used (also efficiently) to find good heuristic choices for QW and rapid quench parameters.

In Chapter 6, I explored numerically and experimentally the application of AQC and QA to a problem of interest in post-quantum cryptography: a lattice-based problem known as the shortest vector problem (SVP). First, I presented two novel mappings of SVP into the Ising model, one being spatially efficient but requiring good quality qubits with precise controls, and one being spatially inefficient but much more tolerant to hardware imperfections. I gave analyt-

ical upper-bounds on the number of qubits required for each mapping. Through a combination of numerical simulation in ideal, closed system context with all-to-all qubit connectivity and experimental implementation on a noisy D-Wave device with only chimera connectivity, I was able to compare and contrast how well the AQC and QA algorithms can perform using the two mappings. While SVP problems at sizes of practical interest were far out of reach both numerically and experimentally, the experiments provided an analysis on real QA hardware of lattice problems of up to 7-dimensions, utilising up to 56 logical qubits spread over around 1000 physical qubits.

7.2 Future Work

Many of the results presented in this thesis, including all scaling estimates, are calculated numerically, based on problem instances small enough to be accessible to classical numerics on the hardware available to me. While numerics at these sizes can constitute strong evidence, it cannot be guaranteed that the behaviour at larger sizes will not deviate. One approach to resolving this would be to perform numerics at larger sizes using more powerful hardware; state of the art supercomputing hardware is capable of fully simulating algorithms on around 50 qubits ([De Raedt et al., 2019](#)), while compression techniques have been used to reach around 60 qubits with reduced fidelity ([Wu et al., 2019](#)). Numerical simulation will, nonetheless, always be limited in the sizes it can reach. An alternative approach would be to develop analytical techniques to study the performance of the CTQC algorithms without performing simulations. Whether based on the ‘fast-dynamics’ framework for rapid quenches developed in [Chapter 5](#) or on completely different techniques, such an analytical treatment could determine if finite-size effects are in play at the sizes studied here. Such an analytical approach may turn out not to be possible, however.

A perhaps more likely direction for future research is to begin testing these results on actual quantum hardware. While this has been done to some extent in [Chapter 6](#) regarding the QA algorithms for the SVP on a D-Wave device, the prevalence of open-systems effects along with the requirement for high-overhead embedding due to the limited connectivity of the interaction made the results difficult to interpret. Quantum computing (QC) hardware continues to improve. For example, D-Wave’s recent Advantage processor has significantly higher connectivity

than its predecessors, employing the Pegasus interaction graph (Boothby et al., 2020; Dattani et al., 2019) rather than the Chimera graph. The qubits remain quite noisy, however.

Other hardware tends to have far fewer but much higher quality qubits than the D-Wave devices. State of the art superconductor-based quantum hardware by Google (Arute et al., 2019) and IBM (Mooney et al., 2021) have 53 and 65 qubits respectively, while the latter promises a device with more than 1000 qubits within a few years (Cho, 2020). A large European project (QLSI, 2020) has been started which aims to produce and demonstrate a 16-qubit device based on semiconductor technology, again with the ultimate aim to produce a device with more than 1000 qubits. IonQ, a leader in ion-trap qubit technology, boasts a state of the art 32-qubit device (IonQ, 2020). While these devices are aimed at gate model applications, it is not implausible that CTQC algorithms could be run on them. Doing so would require low-level control of the devices in order to subject them to Hamiltonians other than those used to induce the standard quantum gate unitaries. As such, any of these or future gate model devices could be used as test-beds for the CTQC algorithms discussed in this thesis, although limitations on qubit connectivity remains a problem. It's difficult how to estimate how many physical qubits are required to test the results in this thesis. Taking the D-Wave Pegasus graph as an example, an embedding of an all-to-all problem with 20 logical qubits, calculated by the minorminer library (D-Wave, 2017-), required 56 physical qubits, while a problem with 50 logical qubits required 353 qubits. Calculating other sizes, the dependence appears roughly quadratic, scaling approximately as $n_{\text{physical}} = (0.17 \pm 0.04)n_{\text{logical}}^{(1.96 \pm 0.03)}$. As the transverse-field driver acts on the individual physical qubits, it is unclear what effect spreading logical qubits out over multiple qubits would have on, for example, the correlations shown to be exploited by QWs in Chapter 3.

The existence of powerful classical algorithms for optimization problems, an example of which is discussed in Chapter 4, illustrates the need for clever CTQC algorithm design to compete. While Chapter 5 began this process by studying rapid quenches in QA at an analytical level, there is plenty of other research to do in this direction. It would be useful to confirm that the techniques for efficient heuristic parameter setting described in that chapter really can be employed for other problems, such as MAX2SAT. More broadly, it is hopefully possible to develop the ideas in that chapter further to find new insight into the operation of rapid quenches, and to find new tools. Furthermore, one approach I have not discussed in any detail

is the hybridisation of CTQC techniques with otherwise classical algorithms. This approach is already employed in near term gate-model algorithms, such as variational quantum eigensolver (VQE) algorithms ([Peruzzo et al., 2014](#)), which consists of a simple parameterized quantum circuit that is run repeatedly within an outer classical loop that optimizes its parameters. A very similar combination of classical and quantum components may be possible in CTQC, and is worth exploring. Another example might involve embedding a biased two-stage QW (see [5.2.2](#)) within a classical loop to optimize the bias angle. Broadly, this approach would allow harnessing the power of well-developed classical algorithmic techniques, while enhancing with CTQC whichever components are most amenable, to produce hybrid algorithms that perform better than either classical or quantum algorithms alone.

Bibliography

- Steven H. Adachi and Maxwell P. Henderson. Application of quantum annealing to training of deep neural networks, 2015. URL <https://arxiv.org/abs/1519.06356>. arXiv:1519.06356.
- Dorit Aharonov and Michael Ben-Or. Fault tolerant quantum computation with constant error, 1996. URL <https://arxiv.org/abs/quant-ph/9611025>. arXiv:quant-ph/9611025.
- Dorit Aharonov, Andris Ambainis, Julia Kempe, and Umesh Vazirani. Quantum walks on graphs. In *Proceedings of the Thirty-third Annual ACM Symposium on Theory of Computing*, STOC '01, pages 50–59, New York, NY, USA, 2001. ACM. ISBN 1-58113-349-9. doi: 10.1145/380752.380758. URL <http://doi.acm.org/10.1145/380752.380758>.
- Dorit Aharonov, Wim van Dam, Julia Kempe, Zeph Landau, Seth Lloyd, and Oded Regev. Adiabatic Quantum Computation is Equivalent to Standard Quantum Computation. *SIAM Journal of Computing*, 37(1):166–194, 2004. ISSN 19492081. doi: 10.1109/Group4.2015.7305974. URL <http://arxiv.org/abs/quant-ph/0405098>.
- Dorit Aharonov, Daniel Gottesman, Sandy Irani, and Julia Kempe. The power of quantum systems on a line. *Communications in Mathematical Physics*, 287(1):41–65, Apr 2009. doi: 10.1007/s00220-008-0710-3. URL <https://doi.org/10.1007/s00220-008-0710-3>.
- Miklós Ajtai, Ravi Kumar, and Dandapani Sivakumar. A Sieve Algorithm for the Shortest Lattice Vector Problem. *Proc. 33rd ACM Symp. on Theory of Computing (STOC)*, 2146: 601–610, 2001. ISSN 16113349. doi: 10.1007/3-540-44670-2{_}1.
- Awad H. Al-Mohy and Nicholas J. Higham. Computing the action of the matrix exponential, with an application to exponential integrators. *SIAM Journal on Scientific Computing*, 33(2):488–511, 2011. doi: 10.1137/100788860. URL <https://doi.org/10.1137/100788860>.
- Tameem Albash and Daniel A. Lidar. Adiabatic quantum computation. *Rev. Mod. Phys.*, 90: 015002, Jan 2018a. doi: 10.1103/RevModPhys.90.015002. URL <https://link.aps.org/doi/10.1103/RevModPhys.90.015002>.
- Tameem Albash and Daniel A. Lidar. Demonstration of a scaling advantage for a quantum annealer over simulated annealing. *Phys. Rev. X*, 8:031016, Jul 2018b. doi: 10.1103/PhysRevX.8.031016. URL <https://link.aps.org/doi/10.1103/PhysRevX.8.031016>.

- Andris Ambainis, Kaspars Balodis, Jānis Iraids, Martins Kokainis, Krišjānis Prūsis, and Jevgēnijs Vihrovs. *Quantum Speedups for Exponential-Time Dynamic Programming Algorithms*, pages 1783–1793. Assoc. for Comp. Machinery, New York, 2019. doi: 10.1137/1.9781611975482.107. URL <https://epubs.siam.org/doi/abs/10.1137/1.9781611975482.107>.
- M. H. S. Amin. Consistency of the adiabatic theorem. *Phys. Rev. Lett.*, 102:220401, Jun 2009. doi: 10.1103/PhysRevLett.102.220401. URL <https://link.aps.org/doi/10.1103/PhysRevLett.102.220401>.
- Mohammad H. Amin, Evgeny Andriyash, Jason Rolfe, Bohdan Kulchytskyy, and Roger Melko. Quantum Boltzmann machine. *Phys. Rev. X*, 8:021050, May 2018a. doi: 10.1103/PhysRevX.8.021050. URL <https://link.aps.org/doi/10.1103/PhysRevX.8.021050>.
- Mohammad H. Amin, Evgeny Andriyash, Jason Rolfe, Bohdan Kulchytskyy, and Roger Melko. Quantum boltzmann machine. *Phys. Rev. X*, 8:021050, May 2018b. doi: 10.1103/PhysRevX.8.021050. URL <https://link.aps.org/doi/10.1103/PhysRevX.8.021050>.
- Yoshinori Aono, Phong Q. Nguyen, and Yixin Shen. Quantum lattice enumeration and tweaking discrete pruning. In *ASIACRYPT*, 2018. ISBN 9783030033255. doi: 10.1007/978-3-030-03326-2_14.
- Frank Arute, Kunal Arya, Ryan Babbush, Dave Bacon, Joseph C Bardin, Rami Barends, Rupak Biswas, Sergio Boixo, Fernando G S L Brandao, and David A Buell. Quantum supremacy using a programmable superconducting processor. *Nature*, 574(7779):505–510, 2019. ISSN 1476-4687.
- J. Atalaya, S. Zhang, M. Y. Niu, A. Babakhani, H. C. H. Chan, J. Epstein, and K. B. Whaley. Continuous quantum error correction for evolution under time-dependent hamiltonians, 2020. URL <https://arxiv.org/abs/2003.11248>. arXiv:2003.11248.
- C. L. Baldwin and C. R. Laumann. Quantum algorithm for energy matching in hard optimization problems. *Phys. Rev. B*, 97:224201, Jun 2018. doi: 10.1103/PhysRevB.97.224201. URL <https://link.aps.org/doi/10.1103/PhysRevB.97.224201>.
- Rene Beier and Berthold Vöcking. Random knapsack in expected polynomial time. *Journal of Computer and System Sciences*, 69(3):306 – 329, 2004. ISSN 0022-0000. doi: <https://doi.org/10.1016/j.jcss.2004.04.004>. URL <http://www.sciencedirect.com/science/article/pii/S0022000004000431>. Special Issue on STOC 2003.
- 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*, 22(5):563–591, 1980. ISSN 0022-4715.

- C. Bennett, E. Bernstein, G. Brassard, and U. Vazirani. Strengths and Weaknesses of Quantum Computing. *SIAM Journal on Computing*, 26(5):1510–1523, 1997. doi: 10.1137/S0097539796300933. URL <https://doi.org/10.1137/S0097539796300933>.
- Charles H Bennett and Gilles Brassard. Quantum cryptography: Public key distribution and con tos5. In *Proceedings of the International Conference on Computers, Systems and Signal Processing*, 1984.
- Charles H. Bennett, Gilles Brassard, Seth Breidbart, and Stephen Wiesner. Quantum cryptography, or unforgeable subway tokens. In David Chaum, Ronald L. Rivest, and Alan T. Sherman, editors, *Advances in Cryptology*, pages 267–275, Boston, MA, 1983. Springer US.
- Charles H. Bennett, Gilles Brassard, Claude Crépeau, Richard Jozsa, Asher Peres, and William K. Wootters. Teleporting an unknown quantum state via dual classical and einstein-podolsky-rosen channels. *Phys. Rev. Lett.*, 70:1895–1899, Mar 1993. doi: 10.1103/PhysRevLett.70.1895. URL <https://link.aps.org/doi/10.1103/PhysRevLett.70.1895>.
- Hannes Bernien, Sylvain Schwartz, Alexander Keesling, Harry Levine, Ahmed Omran, Hannes Pichler, Soonwon Choi, Alexander S. Zibrov, Manuel Endres, Markus Greiner, Vladan Vuletić, and Mikhail D. Lukin. Probing many-body dynamics on a 51-atom quantum simulator. *Nature*, 551:579 EP –, 11 2017a. URL <https://doi.org/10.1038/nature24622>.
- Hannes Bernien, Sylvain Schwartz, Alexander Keesling, Harry Levine, Ahmed Omran, Hannes Pichler, Soonwon Choi, Alexander S. Zibrov, Manuel Endres, Markus Greiner, Vladan Vuletić, and Mikhail D. Lukin. Probing many-body dynamics on a 51-atom quantum simulator. *Nature*, 551:579 EP –, 11 2017b. URL <https://doi.org/10.1038/nature24622>.
- Daniel Bernstein, Johannes Buchmann, and Erik Dahmen. *Introduction to post-quantum cryptography*. Springer-Verlag, 2009.
- Zhengbing Bian, Fabian Chudak, William G. Macready, Lane Clark, and Frank Gaitan. Experimental determination of ramsey numbers. *Phys. Rev. Lett.*, 111:130505, Sep 2013. doi: 10.1103/PhysRevLett.111.130505. URL <https://link.aps.org/doi/10.1103/PhysRevLett.111.130505>.
- Sergio Boixo, Tameem Albash, Federico M. Spedalieri, Nicholas Chancellor, and Daniel A. Lidar. Experimental signature of programmable quantum annealing. *Nature Communications*, 4:2067 EP –, 06 2013. URL <https://doi.org/10.1038/ncomms3067>.
- Adam D. Bookatz, Edward Farhi, and Leo Zhou. Error suppression in hamiltonian-based quantum computation using energy penalties. *Phys. Rev. A*, 92:022317, Aug 2015. doi: 10.1103/PhysRevA.92.022317. URL <https://link.aps.org/doi/10.1103/PhysRevA.92.022317>.

- Kelly Boothby, Paul Bunyk, Jack Raymond, and Aidan Roy. Next-generation topology of d-wave quantum processors, 2020. URL <https://arxiv.org/abs/2003.00133>. arXiv preprint arXiv:2003.00133.
- Tomas Boothby, Andrew D King, and Aidan Roy. Fast clique minor generation in Chimera qubit connectivity graphs. *Quantum Information Processing*, 15(1):495–508, 2016. ISSN 1570-0755.
- D. Boschi, S. Branca, F. De Martini, L. Hardy, and S. Popescu. Experimental realization of teleporting an unknown pure quantum state via dual classical and einstein-podolsky-rosen channels. *Phys. Rev. Lett.*, 80:1121–1125, Feb 1998. doi: 10.1103/PhysRevLett.80.1121. URL <https://link.aps.org/doi/10.1103/PhysRevLett.80.1121>.
- Jean-Philippe Bouchaud, Leticia F Cugliandolo, Jorge Kurchan, and Marc Mezard. *Out of equilibrium dynamics in spin-glasses and other glassy systems*, pages 161–223. World Scientific, Singapore, 1998. doi: 10.1142/9789812819437_0006. URL https://www.worldscientific.com/doi/abs/10.1142/9789812819437_0006.
- Dik Bouwmeester, Jian-Wei Pan, Klaus Mattle, Manfred Eibl, Harald Weinfurter, and Anton Zeilinger. Experimental quantum teleportation. *Nature*, 390(6660):575–579, Dec 1997. ISSN 1476-4687. doi: 10.1038/37539. URL <https://doi.org/10.1038/37539>.
- Lucas T. Brady, Christopher L. Baldwin, Aniruddha Bapat, Yaroslav Kharkov, and Alexey V. Gorshkov. Optimal protocols in quantum annealing and qaoa problems, 2020. URL <https://arxiv.org/abs/2003.08952>. arXiv:2003.08952.
- Michael J. Bremner, Christopher M. Dawson, Jennifer L. Dodd, Alexei Gilchrist, Aram W. Harrow, Duncan Mortimer, Michael A. Nielsen, and Tobias J. Osborne. Practical Scheme for Quantum Computation with Any Two-Qubit Entangling Gate. *Phys. Rev. Lett.*, 89:247902, Nov 2002. doi: 10.1103/PhysRevLett.89.247902. URL <https://link.aps.org/doi/10.1103/PhysRevLett.89.247902>.
- J. Brooke, D. Bitko, T. F. Rosenbaum, and G. Aeppli. Quantum annealing of a disordered magnet. *Science*, 284(5415):779–781, 1999. ISSN 0036-8075. doi: 10.1126/science.284.5415.779. URL <http://science.sciencemag.org/content/284/5415/779>.
- T. F. Rosenbaum Brooke, J. and G. Aeppli. Tunable quantum tunnelling of magnetic domain walls. *Nature*, 413:610–613, 2001. doi: 10.1038/35098037. URL <https://doi.org/10.1038/35098037>.
- Jun Cai, William G. Macready, and Aidan Roy. A practical heuristic for finding graph minors, 2014. URL <https://arxiv.org/abs/1406.2741>. arXiv preprint arXiv:1406.2741.
- A. R. Calderbank and Peter W. Shor. Good quantum error-correcting codes exist. *Phys. Rev. A*, 54:1098–1105, Aug 1996. doi: 10.1103/PhysRevA.54.1098. URL <https://link.aps.org/doi/10.1103/PhysRevA.54.1098>.

- Adam Callison, Nicholas Chancellor, Florian Mintert, and Viv Kendon. Finding spin glass ground states using quantum walks. *New Journal of Physics*, 21(12):123022, dec 2019. doi: 10.1088/1367-2630/ab5ca2. URL <https://doi.org/10.1088/1367-2630/ab5ca2>.
- Adam Callison, Max Festenstein, Jie Chen, Laurentiu Nita, Viv Kendon, and Nicholas Chancellor. Energetic perspective on rapid quenches in quantum annealing. *PRX Quantum*, 2: 010338, Mar 2021. doi: 10.1103/PRXQuantum.2.010338. URL <https://link.aps.org/doi/10.1103/PRXQuantum.2.010338>.
- Lorenzo Campos Venuti and Daniel A. Lidar. Error reduction in quantum annealing using boundary cancellation: Only the end matters. *Phys. Rev. A*, 98:022315, Aug 2018. doi: 10.1103/PhysRevA.98.022315. URL <https://link.aps.org/doi/10.1103/PhysRevA.98.022315>.
- Shantanav Chakraborty, Leonardo Novo, and Jérémie Roland. Finding a marked node on any graph by continuous-time quantum walk. *arXiv preprint arXiv:1807.05957*, 2018. URL <https://arxiv.org/abs/1807.05957>.
- N. Chancellor, S. Zohren, P. A. Warburton, S. C. Benjamin, and S. Roberts. A direct mapping of max k-sat and high order parity checks to a chimera graph. *Scientific Reports*, 6(1):37107, 2016a. ISSN 2045-2322. URL <https://doi.org/10.1038/srep37107>.
- N Chancellor, A Callison, V Kendon, and F Mintert. Finding spin-glass ground states using quantum walks [dataset], 2019. URL <https://doi.org/10.15128/r21544bp097>. Data archive at Durham University, UK, for spin glass instances used in this work.
- Nicholas Chancellor. Modernizing quantum annealing using local searches. *New Journal of Physics*, 19(2):023024, feb 2017. doi: 10.1088/1367-2630/aa59c4. URL <https://doi.org/10.1088/1367-2630/aa59c4>.
- Nicholas Chancellor, Szilard Szoke, Walter Vinci, Gabriel Aeppli, and Paul A. Warburton. Maximum-entropy inference with a programmable annealer. *Scientific Reports*, 6(1):22318, 2016b. ISSN 2045-2322. URL <https://doi.org/10.1038/srep22318>.
- Nicholas Chancellor, Stefan Zohren, and Paul A. Warburton. Circuit design for multi-body interactions in superconducting quantum annealing systems with applications to a scalable architecture. *npj Quantum Information*, 3(21), 2017. doi: 10.1038/s41534-017-0022-6. URL <https://www.nature.com/articles/s41534-017-0022-6>.
- Donny Cheung, Peter Høyer, and Nathan Wiebe. Improved error bounds for the adiabatic approximation. *Journal of Physics A: Mathematical and Theoretical*, 44(41):415302, sep 2011. doi: 10.1088/1751-8113/44/41/415302. URL <https://doi.org/10.1088/1751-8113/44/41/415302>.

- Andrew M. Childs. Universal computation by quantum walk. *Phys. Rev. Lett.*, 102:180501, May 2009. doi: 10.1103/PhysRevLett.102.180501. URL <https://link.aps.org/doi/10.1103/PhysRevLett.102.180501>.
- Andrew M. Childs and Jeffrey Goldstone. Spatial search by quantum walk. *Phys. Rev. A*, 70:022314, Aug 2004. doi: 10.1103/PhysRevA.70.022314. URL <https://link.aps.org/doi/10.1103/PhysRevA.70.022314>.
- Andrew M. Childs, Enrico Deotto, Edward Farhi, Jeffrey Goldstone, Sam Gutmann, and Andrew J. Landahl. Quantum search by measurement. *Phys. Rev. A*, 66:032314, Sep 2002. doi: 10.1103/PhysRevA.66.032314. URL <https://link.aps.org/doi/10.1103/PhysRevA.66.032314>.
- Andrew M. Childs, Richard Cleve, Enrico Deotto, Edward Farhi, Sam Gutmann, and Daniel A. Spielman. Exponential Algorithmic Speedup by a Quantum Walk. In *Proceedings of the Thirty-fifth Annual ACM Symposium on Theory of Computing*, STOC '03, pages 59–68, New York, NY, USA, 2003. ACM. ISBN 1-58113-674-9. doi: 10.1145/780542.780552. URL <http://doi.acm.org/10.1145/780542.780552>.
- Andrew M. Childs, David Gosset, and Zak Webb. Universal computation by multiparticle quantum walk. *Science*, 339(6121):791–794, 2013. ISSN 0036-8075. doi: 10.1126/science.1229957. URL <https://science.sciencemag.org/content/339/6121/791>.
- Adrian Cho. IBM promises 1000-qubit quantum computer—a milestone—by 2023, 2020. URL <https://www.sciencemag.org/news/2020/09/ibm-promises-1000-qubit-quantum-computer-milestone-2023>. Accessed: 2021-03-10.
- Vicky Choi. Minor-embedding in adiabatic quantum computation: I. the parameter setting problem. *Quantum Information Processing*, 7:193–209, 2008.
- Vicky Choi. Adiabatic quantum algorithms for the NP-complete Maximum-Weight Independent set, Exact Cover and 3SAT problems, 2010. URL <https://arxiv.org/abs/1004.2226>. arXiv:1004.2226.
- Vicky Choi. Minor-embedding in adiabatic quantum computation: II. minor-universal graph design. *Quantum Information Processing*, 10:343, 2011a. doi: 10.1007/s11128-010-0200-3.
- Vicky Choi. Different Adiabatic Quantum Optimization Algorithms for the NP-Complete Exact Cover and 3SAT Problems. *Quantum Info. Comput.*, 11(7–8):638–648, July 2011b. ISSN 1533-7146. URL <https://dl.acm.org/doi/10.5555/2230916.2230923>.
- G. E. Coxson, C. R. Hill, and J. C. Russo. Adiabatic quantum computing for finding low-peak-sidelobe codes. In *2014 IEEE High Performance Extreme Computing Conference (HPEC)*,

- pages 1–6, Sep. 2014. doi: 10.1109/HPEC.2014.7040953. URL <https://ieeexplore.ieee.org/abstract/document/7040953>.
- A. Crispin and A. Syriachas. Quantum annealing algorithm for vehicle scheduling. In *2013 IEEE International Conference on Systems, Man, and Cybernetics*, pages 3523–3528, Oct 2013. doi: 10.1109/SMC.2013.601. URL <http://doi.org/10.1109/SMC.2013.601>.
- E. J. Crosson and D. A. Lidar. Prospects for quantum enhancement with diabatic quantum annealing, 2020. URL <https://arxiv.org/abs/2008.09913>. arXiv preprint quant-ph/2008.09913.
- Elizabeth Crosson, Edward Farhi, Cedric Yen-Yu Lin, Han-Hsuan Lin, and Peter Shor. Different Strategies for Optimization Using the Quantum Adiabatic Algorithm, 2014. URL <https://arxiv.org/abs/1401.7320>. arXiv preprint quant-ph/1401.7320.
- Toby S. Cubitt, Ashley Montanaro, and Stephen Piddock. Universal quantum hamiltonians. *Proceedings of the National Academy of Sciences*, 115(38):9497–9502, 2018. ISSN 0027-8424. doi: 10.1073/pnas.1804949115. URL <https://www.pnas.org/content/115/38/9497>.
- Leticia F Cugliandolo. Dynamics of glassy systems, 2002. URL <https://arxiv.org/abs/cond-mat/0210312>. arXiv preprint cond-mat/0210312.
- D-Wave. D-Wave Systems Inc. website, 1999–. URL <http://www.dwavesys.com/>. Accessed: 2021-01-14.
- D-Wave. D-Wave Ocean Software Documentation, 2017–. URL <https://docs.ocean.dwavesys.com/en/stable/>. Accessed: 2021-03-05.
- D-Wave. minorminer, 2017–. URL <https://docs.ocean.dwavesys.com/projects/minorminer/en/latest/>. Accessed: 2021-03-10.
- D-Wave Systems Inc. Probing mid-band and broad-band noise in lower-noise d-wave 2000q fabrication stacks. https://www.dwavesys.com/sites/default/files/14-1034A-A_Probing_noise_in_LN_2000Q_fabrication_stacks.pdf, 2019a. Accessed: 2021-01-14.
- D-Wave Systems Inc. Reverse quantum annealing for local refinement of solutions. https://www.dwavesys.com/sites/default/files/14-1018A-A_Reverse_Quantum_Annealing_for_Local_Refinement_of_Solutions.pdf, 2019b. Accessed: 2021-01-14.
- Alexander M. Dalzell, Theodore J. Yoder, and Isaac L. Chuang. Fixed-point adiabatic quantum search. *Phys. Rev. A*, 95:012311, Jan 2017. doi: 10.1103/PhysRevA.95.012311. URL <https://link.aps.org/doi/10.1103/PhysRevA.95.012311>.
- Nike Dattani. Quadraticization in discrete optimization and quantum mechanics, 2019. URL <http://arxiv.org/abs/1901.04405>. arXiv:1901.04405.

- Nike Dattani, Szilard Szalay, and Nick Chancellor. Pegasus: The second connectivity graph for large-scale quantum annealing hardware, 2019. URL <https://arxiv.org/abs/1901.07636>. arXiv preprint arXiv:1901.07636.
- Gemma De las Cuevas and Toby S. Cubitt. Simple universal models capture all classical spin physics. *Science*, 351(6278):1180–1183, 2016. ISSN 0036-8075. doi: 10.1126/science.aab3326. URL <http://science.sciencemag.org/content/351/6278/1180>.
- Hans De Raedt, Fengping Jin, Dennis Willsch, Madita Willsch, Naoki Yoshioka, Nobuyasu Ito, Shengjun Yuan, and Kristel Michielsen. Massively parallel quantum computer simulator, eleven years later. *Computer Physics Communications*, 237:47–61, 2019. ISSN 0010-4655. doi: <https://doi.org/10.1016/j.cpc.2018.11.005>. URL <https://www.sciencedirect.com/science/article/pii/S0010465518303977>.
- Percy Deift, Mary Beth Ruskai, and Wolfgang Spitzer. Improved gap estimates for simulating quantum circuits by adiabatic evolution. *Quantum Information Processing*, 2(6):121–125, 2007.
- Misha Denil and Nando de Freitas. Toward the implementation of a quantum RBM. In *NIPS 2011 Deep Learning and Unsupervised Feature Learning Workshop*, 2011. URL <http://www.cs.ubc.ca/~nando/papers/quantumrbm.pdf>.
- B. Derrida. Random-Energy Model: Limit of a Family of Disordered Models. *Phys. Rev. Lett.*, 45:79–82, Jul 1980. doi: 10.1103/PhysRevLett.45.79. URL <https://link.aps.org/doi/10.1103/PhysRevLett.45.79>.
- David Deutsch and Roger Penrose. Quantum theory, the church’s turing principle and the universal quantum computer. *Proceedings of the Royal Society of London. A. Mathematical and Physical Sciences*, 400(1818):97–117, 1985. doi: 10.1098/rspa.1985.0070. URL <https://royalsocietypublishing.org/doi/abs/10.1098/rspa.1985.0070>.
- David Elieser Deutsch and Roger Penrose. Quantum computational networks. *Proceedings of the Royal Society of London. A. Mathematical and Physical Sciences*, 425(1868):73–90, 1989. doi: 10.1098/rspa.1989.0099. URL <https://royalsocietypublishing.org/doi/abs/10.1098/rspa.1989.0099>.
- N. G. Dickson et al. Thermally assisted quantum annealing of a 16-qubit problem. *Nature Communications*, 4(1):1903, 2013. ISSN 2041-1723. URL <https://doi.org/10.1038/ncomms2920>.
- Whitfield Diffie and Martin Hellman. New directions in cryptography. *IEEE transactions on Information Theory*, 22(6):644–654, 1976.
- A. R. Dixon, Z. L. Yuan, J. F. Dynes, A. W. Sharpe, and A. J. Shields. Gigahertz decoy quantum key distribution with 1 mbit/s secure key rate. *Opt. Express*, 16(23):18790–18797,

- Nov 2008. doi: 10.1364/OE.16.018790. URL <http://www.opticsexpress.org/abstract.cfm?URI=oe-16-23-18790>.
- Jennifer L. Dodd, Michael A. Nielsen, Michael J. Bremner, and Robert T. Thew. Universal quantum computation and simulation using any entangling hamiltonian and local unitaries. *Phys. Rev. A*, 65:040301, Apr 2002. doi: 10.1103/PhysRevA.65.040301. URL <https://link.aps.org/doi/10.1103/PhysRevA.65.040301>.
- A. Ben Dodds, Viv Kendon, Charles S. Adams, and Nicholas Chancellor. Practical designs for permutation-symmetric problem hamiltonians on hypercubes, Sep 2019. URL <https://link.aps.org/doi/10.1103/PhysRevA.100.032320>.
- Qian-Heng Duan, Shuo Zhang, Wei Wu, and Ping-Xing Chen. An alternative approach to construct the initial hamiltonian of the adiabatic quantum computation. *Chinese Physics Letters*, 30(1):010302, Jan 2013. doi: 10.1088/0256-307x/30/1/010302. URL <https://doi.org/10.1088/0256-307x/30/1/010302>.
- Edward Farhi and Sam Gutmann. Quantum computation and decision trees. *Phys. Rev. A*, 58:915–928, Aug 1998. doi: 10.1103/PhysRevA.58.915. URL <http://link.aps.org/doi/10.1103/PhysRevA.58.915>.
- Edward Farhi, Jeffrey Goldstone, Sam Gutmann, and Michael Sipser. Quantum Computation by Adiabatic Evolution. *arXiv preprint arXiv: quant-ph/0001106*, 2000. URL <http://arxiv.org/abs/quant-ph/0001106>.
- Edward Farhi, Jeffrey Goldstone, Sam Gutmann, Joshua Lapan, Andrew Lundgren, and Daniel Preda. A Quantum Adiabatic Evolution Algorithm Applied to Random Instances of an NP-Complete Problem. *Science*, 292(5516):472–475, 2001. ISSN 0036-8075. doi: 10.1126/science.1057726. URL <http://science.sciencemag.org/content/292/5516/472>.
- Edward Farhi, Jeffrey Goldstone, Sam Gutmann, and Daniel Nagaj. How to make the quantum adiabatic algorithm fail. *International Journal of Quantum Information*, 06(03):503–516, 2008. doi: 10.1142/S021974990800358X. URL <https://doi.org/10.1142/S021974990800358X>.
- Edward Farhi, Jeffrey Goldstone, David Gosset, Sam Gutmann, and Peter Shor. Unstructured randomness, small gaps and localization. *Quantum Information & Computation*, 11(9-10):840–854, Sep 2011. URL <http://www.rintonpress.com/xxqic11/qic-11-910/0840-0854.pdf>.
- Edward Farhi, Jeffrey Goldstone, and Sam Gutmann. A quantum approximate optimization algorithm applied to a bounded occurrence constraint problem, 2014a. URL <https://arxiv.org/abs/1412.6062>. arXiv:1412.6062.

- Edward Farhi, Jeffrey Goldstone, and Sam Gutmann. A Quantum Approximate Optimization Algorithm. *arXiv:1411.4028*, pages 1–16, 2014b. URL <http://arxiv.org/abs/1411.4028>.
- Sheng Feng, Ye Fang, Ka-Ming Tam, Zhifeng Yun, J Ramanujam, Juana Moreno, and Mark Jarrell. Three Dimensional Edwards-Anderson Spin Glass Model in an External Field, 2014. URL <https://arxiv.org/abs/1403.4560>. arXiv:1403.4560.
- Richard P. Feynman. Simulating physics with computers. *International Journal of Theoretical Physics*, 21(6):467–488, Jun 1982. ISSN 1572-9575. doi: 10.1007/BF02650179. URL <https://doi.org/10.1007/BF02650179>.
- Ulrich Fincke and Michael Pohst. Improved methods for calculating vectors of short length in a lattice, including a complexity analysis. *Mathematics of computation*, 44(170):463–471, 1985. ISSN 0025-5718.
- A.B. Finnila, M.A. Gomez, C. Sebenik, C. Stenson, and J.D. Doll. Quantum annealing: A new method for minimizing multidimensional functions. *Chemical Physics Letters*, 219(5): 343 – 348, 1994. ISSN 0009-2614. doi: [https://doi.org/10.1016/0009-2614\(94\)00117-0](https://doi.org/10.1016/0009-2614(94)00117-0). URL <http://www.sciencedirect.com/science/article/pii/0009261494001170>.
- D S Fisher and D A Huse. Absence of many states in realistic spin glasses. *Journal of Physics A: Mathematical and General*, 20(15):L1005–L1010, oct 1987. doi: 10.1088/0305-4470/20/15/013. URL <https://doi.org/10.1088/0305-4470/20/15/013>.
- Daniel S. Fisher and David A. Huse. Equilibrium behavior of the spin-glass ordered phase. *Phys. Rev. B*, 38:386–411, Jul 1988. doi: 10.1103/PhysRevB.38.386. URL <https://link.aps.org/doi/10.1103/PhysRevB.38.386>.
- C. Daniel Freeman, Mohan Sarovar, C. M. Herdman, and K. B. Whaley. Stable quantum memories with limited measurement. *Phys. Rev. A*, 98:032322, Sep 2018. doi: 10.1103/PhysRevA.98.032322. URL <https://link.aps.org/doi/10.1103/PhysRevA.98.032322>.
- A. Friedenauer, H. Schmitz, J. T. Glueckert, D. Porras, and T. Schaetz. Simulating a quantum magnet with trapped ions. *Nature Physics*, 4(10):757–761, Oct 2008. ISSN 1745-2481. doi: 10.1038/nphys1032. URL <https://doi.org/10.1038/nphys1032>.
- Nicolas Gama, Phong Q Nguyen, and Oded Regev. Lattice Enumeration Using Extreme Pruning. *EUROCRYPT 2010, Lecture Notes in Computer Science*, 6110:257–278, 2010.
- M.R. Garey, D.S. Johnson, and L. Stockmeyer. Some simplified np-complete graph problems. *Theoretical Computer Science*, 1(3):237–267, 1976. ISSN 0304-3975. doi: [https://doi.org/10.1016/0304-3975\(76\)90059-1](https://doi.org/10.1016/0304-3975(76)90059-1). URL <https://www.sciencedirect.com/science/article/pii/0304397576900591>.
- Craig Gentry. Fully homomorphic encryption using ideal lattices. In *Proceedings of the forty-first annual ACM symposium on Theory of computing*, pages 169–178, 2009.

- Oded Goldreich, Shafi Goldwasser, and Shai Halevi. Public-Key Cryptosystems from Lattice Reduction Problems. In *CRYPTO*, London, 1997. Springer-Verlag.
- Tobias Graß. Quantum annealing with longitudinal bias fields. *Phys. Rev. Lett.*, 123:120501, Sep 2019. doi: 10.1103/PhysRevLett.123.120501. URL <https://link.aps.org/doi/10.1103/PhysRevLett.123.120501>.
- Tobias Graß and Maciej Lewenstein. Hybrid annealing: Coupling a quantum simulator to a classical computer. *Phys. Rev. A*, 95:052309, May 2017. doi: 10.1103/PhysRevA.95.052309. URL <https://link.aps.org/doi/10.1103/PhysRevA.95.052309>.
- Johnnie Gray. quimb: A python package for quantum information and many-body calculations. *Journal of Open Source Software*, 3(29):819, 2018. doi: 10.21105/joss.00819. URL <https://joss.theoj.org/papers/10.21105/joss.00819>.
- Markus Greiner, Olaf Mandel, Tilman Esslinger, Theodor W. Hänsch, and Immanuel Bloch. Quantum phase transition from a superfluid to a mott insulator in a gas of ultracold atoms. *Nature*, 415(6867):39–44, Jan 2002. ISSN 1476-4687. doi: 10.1038/415039a. URL <https://doi.org/10.1038/415039a>.
- Lov K. Grover. A fast quantum mechanical algorithm for database search. *Proceedings, 28th Annual ACM Symposium on the Theory of Computing*, pages 212–219, 1996. URL <http://arxiv.org/abs/quant-ph/9605043>.
- Stuart Hadfield, Zhihui Wang, Bryan O’Gorman, Eleanor G. Rieffel, Davide Venturelli, and Rupak Biswas. From the quantum approximate optimization algorithm to a quantum alternating operator ansatz. *Algorithms*, 12.2:34, 2019. doi: 10.3390/a12020034. (Special issue “Quantum Optimization Theory, Algorithms, and Applications”).
- Firas Hamze, Jack Raymond, Christopher A. Pattison, Katja Biswas, and Helmut G. Katzgraber. The wishart planted ensemble: A tunably-rugged pairwise ising model with a first-order phase transition, 2019. URL <https://arxiv.org/abs/1906.00275>. arXiv preprint arXiv:1906.00275.
- A. Hartwig, F. Daske, and S. Kobe. A recursive branch-and-bound algorithm for the exact ground state of ising spin-glass models. *Computer Physics Communications*, 32(2):133 – 138, 1984. ISSN 0010-4655. doi: [https://doi.org/10.1016/0010-4655\(84\)90066-3](https://doi.org/10.1016/0010-4655(84)90066-3). URL <http://www.sciencedirect.com/science/article/pii/0010465584900663>.
- M. B. Hastings. The power of adiabatic quantum computation with no sign problem, 2020. URL <https://arxiv.org/abs/2005.03791>. arXiv preprint quant-ph/2005.03791.
- Matthew B. Hastings. Duality in Quantum Quenches and Classical Approximation Algorithms: Pretty Good or Very Bad. *Quantum*, 3:201, November 2019. ISSN 2521-327X. doi: 10.22331/q-2019-11-11-201. URL <https://doi.org/10.22331/q-2019-11-11-201>.

- Bettina Helfrich. Algorithms to construct Minkowski reduced and Hermite reduced lattice bases. *Theoretical Computer Science*, 41:125–139, 1985. ISSN 0304-3975.
- Itay Hen. Continuous-time quantum algorithms for unstructured problems. *Journal of Physics A: Mathematical and Theoretical*, 47(4):045305, Jan 2014. doi: 10.1088/1751-8113/47/4/045305. URL <https://doi.org/10.1088/1751-8113/47/4/045305>.
- Itay Hen. Equation planting: A tool for benchmarking ising machines. *Phys. Rev. Applied*, 12: 011003, Jul 2019. doi: 10.1103/PhysRevApplied.12.011003. URL <https://link.aps.org/doi/10.1103/PhysRevApplied.12.011003>.
- John D Hunter. Matplotlib: A 2D graphics environment. *Computing in science & engineering*, 9(3):90, 2007.
- Takahiro Inagaki, Yoshitaka Haribara, Koji Igarashi, Tomohiro Sonobe, Shuhei Tamate, Toshimori Honjo, Alireza Marandi, Peter L. McMahon, Takeshi Umeki, Koji Enbutsu, Osamu Tadanaga, Hirokazu Takenouchi, Kazuyuki Aihara, Ken-ichi Kawarabayashi, Kyo Inoue, Shoko Utsunomiya, and Hiroki Takesue. A coherent Ising machine for 2000-node optimization problems. *Science*, 354(6312):603–606, 2016. ISSN 0036-8075. doi: 10.1126/science.aah4243. URL <http://science.sciencemag.org/content/354/6312/603>.
- IonQ. IonQ Unveils World’s Most Powerful Quantum Computer, 2020. URL <https://ionq.com/news/october-01-2020-most-powerful-quantum-computer>. Accessed: 2021-03-10.
- Sabine Jansen, Mary-Beth Ruskai, and Ruedi Seiler. Bounds for the adiabatic approximation with applications to quantum computation. *Journal of Mathematical Physics*, 48(10):102111, 2007. doi: 10.1063/1.2798382. URL <https://doi.org/10.1063/1.2798382>.
- Hoffstein Jeffrey, Jill Pipher, and Joseph H Silverman. NTRU : A Ring-Based Public Key Cryptosystem. *Algorithmic Number Theory (ANTS-III)(LNCS)*, 1423, 1998.
- M. W. Johnson, M. H. S. Amin, S. Gildert, T. Lanting, F. Hamze, N. Dickson, R. Harris, A. J. Berkley, J. Johansson, P. Bunyk, E. M. Chapple, C. Enderud, J. P. Hilton, K. Karimi, E. Ladizinsky, N. Ladizinsky, T. Oh, I. Perminov, C. Rich, M. C. Thom, E. Tolkacheva, C. J. S. Truncik, S. Uchaikin, J. Wang, B. Wilson, and G. Rose. Quantum annealing with manufactured spins. *Nature*, 473:194 EP –, 05 2011. URL <https://doi.org/10.1038/nature10012>.
- Eric Jones, Travis Oliphant, Pearu Peterson, et al. SciPy: Open source scientific tools for Python, 2001–. URL <http://www.scipy.org/>. [Online; accessed June 7, 2021].
- Stephen Jordan and Edward Farhi. Perturbative gadgets at arbitrary orders. *Physical Review A*, 77, 02 2008. URL <https://link.aps.org/doi/10.1103/PhysRevA.77.062329>.

- Stephen P. Jordan, Edward Farhi, and Peter W. Shor. Error-correcting codes for adiabatic quantum computation. *Phys. Rev. A*, 74:052322, Nov 2006. doi: 10.1103/PhysRevA.74.052322. URL <https://link.aps.org/doi/10.1103/PhysRevA.74.052322>.
- David Joseph, Alexandros Ghionis, Cong Ling, and Florian Mintert. Not-so-adiabatic quantum computation for the shortest vector problem. *Physical Review Research*, 013361:1–13, 2020. doi: 10.1103/PhysRevResearch.2.013361. URL <http://arxiv.org/abs/1910.10462>.
- David Joseph, Adam Callison, Cong Ling, and Florian Mintert. Two quantum ising algorithms for the shortest vector problem: one for now and one for later, 2021a. URL <https://arxiv.org/abs/2006.140571>. arXiv preprint arXiv:2006.14057.
- David Joseph, Adam Callison, Cong Ling, and Florian Mintert. Two quantum ising algorithms for the shortest-vector problem. *Phys. Rev. A*, 103:032433, Mar 2021b. doi: 10.1103/PhysRevA.103.032433. URL <https://link.aps.org/doi/10.1103/PhysRevA.103.032433>.
- Tadashi Kadowaki and Hidetoshi Nishimori. Quantum annealing in the transverse Ising model. *Phys. Rev. E*, 58:5355–5363, Nov 1998. doi: 10.1103/PhysRevE.58.5355. URL <https://link.aps.org/doi/10.1103/PhysRevE.58.5355>.
- Ravi Kannan. Improved algorithms for integer programming and related lattice problems. In *Proceedings of the fifteenth annual ACM symposium on Theory of computing*, pages 193–206, 1983.
- Richard M. Karp. *Reducibility among Combinatorial Problems*, pages 85–103. Springer US, Boston, MA, 1972. ISBN 978-1-4684-2001-2. doi: 10.1007/978-1-4684-2001-2_9. URL https://doi.org/10.1007/978-1-4684-2001-2_9.
- Hitoshi Katsuda and Hidetoshi Nishimori. Nonadiabatic quantum annealing for one-dimensional trasverse-field ising model, 2013. URL <https://arxiv.org/abs/1303.6045>. arXiv preprint cond-mat.stat-mech/1303.6045.
- Helmut G. Katzgraber, Firas Hamze, and Ruben S. Andrist. Glassy chimeras could be blind to quantum speedup: Designing better benchmarks for quantum annealing machines. *Phys. Rev. X*, 4:021008, Apr 2014. doi: 10.1103/PhysRevX.4.021008. URL <https://link.aps.org/doi/10.1103/PhysRevX.4.021008>.
- Kostyantyn Kechedzhi, Vadim Smelyanskiy, Jarrod R. McClean, Vasil S. Denchev, Masoud Mohseni, Sergei Isakov, Sergio Boixo, Boris Altshuler, and Hartmut Neven. Efficient Population Transfer via Non-Ergodic Extended States in Quantum Spin Glass. In Stacey Jeffery, editor, *13th Conference on the Theory of Quantum Computation, Communication and Cryptography (TQC 2018)*, volume 111 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 9:1–9:16, Dagstuhl, Germany, 2018. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik. ISBN 978-3-95977-080-4. doi: 10.4230/LIPIcs.TQC.2018.9. URL <http://drops.dagstuhl.de/opus/volltexte/2018/9256>.

- Julia Kempe, Alexei Kitaev, and Oded Regev. The complexity of the local hamiltonian problem. *SIAM Journal on Computing*, 35(5):1070–1097, 2006. doi: 10.1137/S0097539704445226. URL <https://doi.org/10.1137/S0097539704445226>.
- K Kim, S Korenblit, R Islam, E E Edwards, M-S Chang, C Noh, H Carmichael, G-D Lin, L-M Duan, C C Joseph Wang, J K Freericks, and C Monroe. Quantum simulation of the transverse ising model with trapped ions. *New Journal of Physics*, 13(10):105003, oct 2011. doi: 10.1088/1367-2630/13/10/105003. URL <https://doi.org/10.1088/1367-2630/13/10/105003>.
- S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi. Optimization by simulated annealing. *Science*, 220(4598):671–680, 1983. ISSN 0036-8075. doi: 10.1126/science.220.4598.671. URL <https://science.sciencemag.org/content/220/4598/671>.
- Alexei Yu Kitaev, Alexander Shen, Mikhail N Vyalyi, and Mikhail N Vyalyi. *Classical and quantum computation*. Number 47 in Graduate Studies in Mathematics. American Mathematical Soc., 2002.
- Thomas Kluyver, Benjamin Ragan-Kelley, Fernando Pérez, Brian Granger, Matthias Bussonnier, Jonathan Frederic, Kyle Kelley, Jessica Hamrick, Jason Grout, Sylvain Corlay, Paul Ivanov, Damián Avila, Safia Abdalla, and Carol Willing. Jupyter notebooks – a publishing format for reproducible computational workflows. In F. Loizides and B. Schmidt, editors, *Positioning and Power in Academic Publishing: Players, Agents and Agendas*, pages 87 – 90. IOS Press, 2016.
- E. Knill, R. Laflamme, and W. Zurek. Threshold accuracy for quantum computation, 1996. URL <https://arxiv.org/abs/quant-ph/9610011>. arXiv:quant-ph/9610011.
- Sergey Knysh. Zero-temperature quantum annealing bottlenecks in the spin-glass phase. *Nature Communications*, 7:12370 EP –, 08 2016. URL <https://doi.org/10.1038/ncomms12370>.
- Michael Krivelevich and Dan Vilenchik. Solving random satisfiable 3CNF formulas in expected polynomial time. In *Proceedings of the Seventeenth Annual ACM-SIAM Symposium on Discrete Algorithm*, SODA '06, pages 454–463, Philadelphia, PA, USA, 2006. Society for Industrial and Applied Mathematics. ISBN 0-89871-605-5. URL <http://dl.acm.org/citation.cfm?id=1109557.1109608>.
- Thijs Laarhoven, Michele Mosca, and Joop Van De Pol. Solving the shortest vector problem in lattices faster using quantum search. In *PQCrypto*, pages 83–101, 2013. ISBN 9783642386152. doi: 10.1007/978-3-642-38616-9{_}6.
- Trevor Lanting. Next Generation of QA hardware, 2018. AQC 2018 <https://www.youtube.com/watch?v=05ovPNxmfjE>.

- Derek Larson, Helmut G. Katzgraber, M. A. Moore, and A. P. Young. Spin glasses in a field: Three and four dimensions as seen from one space dimension. *Phys. Rev. B*, 87:024414, Jan 2013. doi: 10.1103/PhysRevB.87.024414. URL <https://link.aps.org/doi/10.1103/PhysRevB.87.024414>.
- E. L. Lawler and D. E. Wood. Branch-and-bound methods: A survey. *Operations Research*, 14(4):699–719, 1966. doi: 10.1287/opre.14.4.699. URL <https://doi.org/10.1287/opre.14.4.699>.
- Wolfgang Lechner, Philipp Hauke, and Peter Zoller. A quantum annealing architecture with all-to-all connectivity from local interactions. *Science Advances*, 1(9), 2015. doi: 10.1126/sciadv.1500838. URL <http://advances.sciencemag.org/content/1/9/e1500838>.
- Martin Leib, Peter Zoller, and Wolfgang Lechner. A transmon quantum annealer: decomposing many-body Ising constraints into pair interactions. *Quantum Science and Technology*, 1(1): 015008, 2016. URL <http://stacks.iop.org/2058-9565/1/i=1/a=015008>.
- A. K. Lenstra, H. W. Lenstra, and L. Lovász. Factoring polynomials with rational coefficients. *Mathematische Annalen*, 261(4):515–534, 1982. ISSN 00255831. doi: 10.1007/BF01457454.
- Arjen K Lenstra and W Hendrik Jr. *The development of the number field sieve*, volume 1554. Springer Science & Business Media, 1993. ISBN 3540570136.
- Zhaokai Li, Nikesh S Dattani, Xi Chen, Xiaomei Liu, Hengyan Wang, Richard Tanburn, Hongwei Chen, Xinhua Peng, and Jiangfeng Du. High-fidelity adiabatic quantum computation using the intrinsic Hamiltonian of a spin system: Application to the experimental factorization of 291311, 2017. URL <https://arxiv.org/abs/1706.08061>. arXiv preprint arXiv:1706.08061.
- Daniel A. Lidar. Towards fault tolerant adiabatic quantum computation. *Phys. Rev. Lett.*, 100: 160506, Apr 2008. doi: 10.1103/PhysRevLett.100.160506. URL <https://link.aps.org/doi/10.1103/PhysRevLett.100.160506>.
- Daniel A. Lidar, Ali T. Rezakhani, and Alioscia Hamma. Adiabatic approximation with exponential accuracy for many-body systems and quantum computation. *Journal of Mathematical Physics*, 50:102106, 2009. doi: 10.1063/1.3236685.
- Seth Lloyd. Universal quantum simulators. *Science*, 273(5278):1073–1078, 1996. ISSN 00368075, 10959203. URL <http://www.jstor.org/stable/2899535>.
- Neil B. Lovett, Sally Cooper, Matthew Everitt, Matthew Trevers, and Viv Kendon. Universal quantum computation using the discrete-time quantum walk. *Phys. Rev. A*, 81:042330, Apr 2010. doi: 10.1103/PhysRevA.81.042330. URL <https://link.aps.org/doi/10.1103/PhysRevA.81.042330>.

- Andrew Lucas. Ising formulations of many NP problems. *Frontiers in Physics*, 2:5, 2014. ISSN 2296-424X. doi: 10.3389/fphy.2014.00005. URL <https://www.frontiersin.org/article/10.3389/fphy.2014.00005>.
- S. G. Magalhaes, C. V. Morais, F. M. Zimmer, M. J. Lazo, and F. D. Nobre. Nonlinear susceptibility of a quantum spin glass under uniform transverse and random longitudinal magnetic fields. *Phys. Rev. B*, 95:064201, Feb 2017. doi: 10.1103/PhysRevB.95.064201. URL <https://link.aps.org/doi/10.1103/PhysRevB.95.064201>.
- Y Ivanovich Manin. *Vychislimoe i nevychislimoe [Computable and Noncomputable] (in Russian)*. Sov. radio, 1980.
- E. Manousakis. A quantum-dot array as model for copper-oxide superconductors: A dedicated quantum simulator for the many-fermion problem. *Journal of Low Temperature Physics*, 126(5):1501–1513, Mar 2002. ISSN 1573-7357. doi: 10.1023/A:1014295416763. URL <https://doi.org/10.1023/A:1014295416763>.
- S. Marsh and J. B. Wang. A quantum walk-assisted approximate algorithm for bounded NP optimisation problems. *Quantum Information Processing*, 18(3):61, Jan 2019. ISSN 1573-1332. doi: 10.1007/s11128-019-2171-3. URL <https://doi.org/10.1007/s11128-019-2171-3>.
- Jeffrey Marshall, Davide Venturelli, Itay Hen, and Eleanor G. Rieffel. Power of pausing: Advancing understanding of thermalization in experimental quantum annealers. *Phys. Rev. Applied*, 11:044083, Apr 2019. doi: 10.1103/PhysRevApplied.11.044083. URL <https://link.aps.org/doi/10.1103/PhysRevApplied.11.044083>.
- Michael Marzec. *Portfolio Optimization: Applications in Quantum Computing*, pages 73–106. John Wiley & Sons, Inc., 2016. ISBN 9781118593486. doi: 10.1002/9781118593486.ch4. URL <http://dx.doi.org/10.1002/9781118593486.ch4>.
- Catherine C McGeoch, Richard Harris, Steven P Reinhardt, and Paul I Bunyk. Practical annealing-based quantum computing. *Computer*, 52(6):38–46, 2019. ISSN 0018-9162.
- Wes McKinney et al. Data structures for statistical computing in python. In *Proceedings of the 9th Python in Science Conference*, volume 445, pages 51–56. Austin, TX, 2010.
- Peter L. McMahon, Alireza Marandi, Yoshitaka Haribara, Ryan Hamerly, Carsten Langrock, Shuhei Tamate, Takahiro Inagaki, Hiroki Takesue, Shoko Utsunomiya, Kazuyuki Aihara, Robert L. Byer, M. M. Fejer, Hideo Mabuchi, and Yoshihisa Yamamoto. A fully-programmable 100-spin coherent Ising machine with all-to-all connections. *Science*, 2016. ISSN 0036-8075. doi: 10.1126/science.aah5178. URL <http://science.sciencemag.org/content/early/2016/10/19/science.aah5178>.
- Hermann Minkowski. *Geometrie der zahlen*, volume 2. Ripol Classic, 2016. ISBN 588093313X.

- Ashley Montanaro. Quantum-walk speedup of backtracking algorithms. *Theory Comput.*, 14 (15):1–24, 9 2018. URL <http://arxiv.org/abs/1509.02374>.
- Ashley Montanaro. Quantum speedup of branch-and-bound algorithms. *Physical Review Research*, 2(1):13056, 2020.
- Gary J. Mooney, Gregory A. L. White, Charles D. Hill, and Lloyd C. L. Hollenberg. Whole-device entanglement in a 65-qubit superconducting quantum computer, 2021. URL <https://arxiv.org/abs/2102.11521>. arXiv preprint arXiv:2102.11521.
- James G. Morley, Nicholas Chancellor, Sougato Bose, and Viv Kendon. Quantum search with hybrid adiabatic–quantum-walk algorithms and realistic noise. *Phys. Rev. A*, 99:022339, Feb 2019. doi: 10.1103/PhysRevA.99.022339. URL <https://link.aps.org/doi/10.1103/PhysRevA.99.022339>.
- Dominic J. Moylett, Noah Linden, and Ashley Montanaro. Quantum speedup of the traveling-salesman problem for bounded-degree graphs. *Phys. Rev. A*, 95:032323, Mar 2017. doi: 10.1103/PhysRevA.95.032323. URL <https://link.aps.org/doi/10.1103/PhysRevA.95.032323>.
- Siddharth Muthukrishnan, Tameem Albash, and Daniel A. Lidar. Tunneling and speedup in quantum optimization for permutation-symmetric problems. *Phys. Rev. X*, 6:031010, Jul 2016. doi: 10.1103/PhysRevX.6.031010. URL <https://link.aps.org/doi/10.1103/PhysRevX.6.031010>.
- Michael A. Nielsen, Michael J. Bremner, Jennifer L. Dodd, Andrew M. Childs, and Christopher M. Dawson. Universal simulation of hamiltonian dynamics for quantum systems with finite-dimensional state spaces. *Phys. Rev. A*, 66:022317, Aug 2002. doi: 10.1103/PhysRevA.66.022317. URL <https://link.aps.org/doi/10.1103/PhysRevA.66.022317>.
- NIST. Post-quantum cryptography standardization. <https://csrc.nist.gov/projects/post-quantum-cryptography/post-quantum-cryptography-standardization>, 2017. Accessed: 2021-01-13.
- Masaki Ohkuwa, Hidetoshi Nishimori, and Daniel A. Lidar. Reverse annealing for the fully connected p -spin model. *Phys. Rev. A*, 98:022314, Aug 2018. doi: 10.1103/PhysRevA.98.022314. URL <https://link.aps.org/doi/10.1103/PhysRevA.98.022314>.
- Travis E Oliphant. *A guide to NumPy*, volume 1. Trelgol Publishing USA, 2006.
- Roberto Oliveira and Barbara M. Terhal. The complexity of quantum spin systems on a two-dimensional square lattice. *Quant. Inf. Comp.*, 8(10):900–924, November 2008. ISSN 1533-7146.

- Román Orús, Samuel Mugel, and Enrique Lizaso. Forecasting financial crashes with quantum computing. *Phys. Rev. A*, 99:060301, Jun 2019. doi: 10.1103/PhysRevA.99.060301. URL <https://link.aps.org/doi/10.1103/PhysRevA.99.060301>.
- Daniel O'Malley. An approach to quantum-computational hydrologic inverse analysis. *Scientific Reports*, 8(1):6919, 2018. ISSN 2045-2322. URL <https://doi.org/10.1038/s41598-018-25206-0>.
- G Parisi. The order parameter for spin glasses: a function on the interval 0-1. *Journal of Physics A: Mathematical and General*, 13(3):1101–1112, mar 1980. doi: 10.1088/0305-4470/13/3/042. URL <https://doi.org/10.1088/0305-4470/13/3/042>.
- G. Passarelli, V. Cataudella, and P. Lucignano. Improving quantum annealing of the ferromagnetic p -spin model through pausing. *Phys. Rev. B*, 100:024302, Jul 2019. doi: 10.1103/PhysRevB.100.024302. URL <https://link.aps.org/doi/10.1103/PhysRevB.100.024302>.
- M.R. Passos, M.M. Taddei, and R.L. de Matos Filho. Error-run-time trade-off in the adiabatic approximation beyond scaling relations. *Annals of Physics*, 418:168172, 2020. ISSN 0003-4916. doi: <https://doi.org/10.1016/j.aop.2020.168172>. URL <http://www.sciencedirect.com/science/article/pii/S0003491620301056>.
- Gerardo A. Paz-Silva, A. T. Rezakhani, Jason M. Dominy, and D. A. Lidar. Zeno effect for quantum computation and control. *Phys. Rev. Lett.*, 108:080501, Feb 2012. doi: 10.1103/PhysRevLett.108.080501. URL <https://link.aps.org/doi/10.1103/PhysRevLett.108.080501>.
- Alejandro Perdomo-Ortiz, Salvador E. Venegas-Andraca, and Alán Aspuru-Guzik. A study of heuristic guesses for adiabatic quantum computation. *Quantum Information Processing*, 10(1):33–52, Feb 2011. ISSN 1573-1332. doi: 10.1007/s11128-010-0168-z. URL <https://doi.org/10.1007/s11128-010-0168-z>.
- Alejandro Perdomo-Ortiz, Neil Dickson, Marshall Drew-Brook, Geordie Rose, and Alán Aspuru-Guzik. Finding low-energy conformations of lattice protein models by quantum annealing. *Scientific Reports*, 2(1):571, 2012. ISSN 2045-2322. URL <https://doi.org/10.1038/srep00571>.
- Fernando Pérez and Brian E Granger. IPython: a system for interactive scientific computing. *Computing in Science & Engineering*, 9(3), 2007.
- Alberto Peruzzo, Jarrod McClean, Peter Shadbolt, Man-Hong Yung, Xiao-Qi Zhou, Peter J. Love, Alán Aspuru-Guzik, and Jeremy L. O'Brien. A variational eigenvalue solver on a photonic quantum processor. *Nature communications*, 5:4213–4213, Jul 2014. ISSN 2041-1723. doi: 10.1038/ncomms5213. URL <https://doi.org/10.1038/ncomms5213>.

- Thomas Plantard, Willy Susilo, and Khin Than Win. A digital signature scheme based on cvp_∞ . In *Public Key Cryptography – PKC 2008*, pages 288–307, Berlin, Heidelberg, 2008. Springer Berlin Heidelberg.
- Kristen L. Pudenz, Tameem Albash, and Daniel A. Lidar. Error-corrected quantum annealing with hundreds of qubits. *Nature Communications*, 5(1):3243, 2014. ISSN 2041-1723. URL <https://doi.org/10.1038/ncomms4243>.
- Kristen L. Pudenz, Tameem Albash, and Daniel A. Lidar. Quantum annealing correction for random ising problems. *Phys. Rev. A*, 91:042302, 2015. doi: 10.1103/PhysRevA.91.042302. URL <https://link.aps.org/doi/10.1103/PhysRevA.91.042302>.
- QLSI. Quantum large scale integretion in silicon, 2020. URL <https://cordis.europa.eu/project/id/951852>. Accessed: 2021-03-10.
- Gregory Quiroz and Daniel A. Lidar. High-fidelity adiabatic quantum computation via dynamical decoupling. *Phys. Rev. A*, 86:042333, Oct 2012. doi: 10.1103/PhysRevA.86.042333. URL <https://link.aps.org/doi/10.1103/PhysRevA.86.042333>.
- Robert Raussendorf, Daniel E. Browne, and Hans J. Briegel. Measurement-based quantum computation on cluster states. *Phys. Rev. A*, 68:022312, Aug 2003. doi: 10.1103/PhysRevA.68.022312. URL <https://link.aps.org/doi/10.1103/PhysRevA.68.022312>.
- Oded Regev. On lattices, learning with errors, random linear codes, and cryptography. *Proceedings of STOC*, pages 84–93, 2005. ISSN 00045411. doi: 10.1145/1568318.1568324. URL <http://portal.acm.org/citation.cfm?doid=1568318.1568324>.
- R. L. Rivest, A. Shamir, and L. Adleman. A method for obtaining digital signatures and public-key cryptosystems. *Communications of the ACM*, 21(2):120–126, 1978. ISSN 00010782. doi: 10.1145/359340.359342. URL <http://portal.acm.org/citation.cfm?doid=359340.359342>.
- Andrea Rocchetto, Simon C. Benjamin, and Ying Li. Stabilisers as a design tool for new forms of Lechner-Hauke-Zoller annealer. *Science Advances*, 2:e1601246, 2016. doi: 10.1126/sciadv.1601246.
- Joschka Roffe, Stefan Zohren, Dominic Horsman, and Nicholas Chancellor. Decoding quantum error correction with ising model hardware, 2019. URL <https://arxiv.org/abs/quant-ph/1903.10254>. arXiv preprint quant-ph/1903.10254.
- J  r  mie Roland and Nicolas J. Cerf. Quantum search by local adiabatic evolution. *Phys. Rev. A*, 65:042308, Mar 2002. doi: 10.1103/PhysRevA.65.042308. URL <http://link.aps.org/doi/10.1103/PhysRevA.65.042308>.

- Michael Rose, Thomas Plantard, and Willy Susilo. Improving BDD cryptosystems in general lattices. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 6672 LNCS:152–167, 2011. ISSN 03029743. doi: 10.1007/978-3-642-21031-0{_}12.
- Siddhartha Santra, Gregory Quiroz, Greg Ver Steeg, and Daniel A Lidar. Max 2-SAT with up to 108 qubits. *New Journal of Physics*, 16(4):045006, apr 2014. doi: 10.1088/1367-2630/16/4/045006. URL <https://doi.org/10.1088/1367-2630/16/4/045006>.
- Mohan Sarovar and G. J. Milburn. Continuous quantum error correction by cooling. *Phys. Rev. A*, 72:012306, Jul 2005. doi: 10.1103/PhysRevA.72.012306. URL <https://link.aps.org/doi/10.1103/PhysRevA.72.012306>.
- Mohan Sarovar and Kevin C Young. Error suppression and error correction in adiabatic quantum computation: non-equilibrium dynamics. *New Journal of Physics*, 15(12):125032, dec 2013. doi: 10.1088/1367-2630/15/12/125032. URL <https://doi.org/10.1088/1367-2630/15/12/125032>.
- C. P. Schnorr and M. Euchner. Lattice basis reduction: Improved practical algorithms and solving subset sum problems. *Mathematical Programming*, 66(1-3):181–199, 1994. ISSN 16113349. doi: 10.1007/3-540-54458-5{_}51.
- Neil Shenvi, Julia Kempe, and K. Birgitta Whaley. Quantum random-walk search algorithm. *Phys. Rev. A*, 67:052307, May 2003. doi: 10.1103/PhysRevA.67.052307. URL <https://link.aps.org/doi/10.1103/PhysRevA.67.052307>.
- David Sherrington and Scott Kirkpatrick. Solvable model of a spin-glass. *Phys. Rev. Lett.*, 35:1792–1796, Dec 1975. doi: 10.1103/PhysRevLett.35.1792. URL <https://link.aps.org/doi/10.1103/PhysRevLett.35.1792>.
- 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. doi: 10.1109/SFCS.1994.365700.
- Peter W. Shor. Scheme for reducing decoherence in quantum computer memory. *Phys. Rev. A*, 52:R2493–R2496, Oct 1995. doi: 10.1103/PhysRevA.52.R2493. URL <https://link.aps.org/doi/10.1103/PhysRevA.52.R2493>.
- M. Stewart Siu. From quantum circuits to adiabatic algorithms. *Phys. Rev. A*, 71:062314, Jun 2005. doi: 10.1103/PhysRevA.71.062314. URL <https://link.aps.org/doi/10.1103/PhysRevA.71.062314>.
- A. Yu. Smirnov, S. Savel’ev, L. G. Mourokh, and Franco Nori. Modelling chemical reactions using semiconductor quantum dots. *Europhysics Letters (EPL)*, 80(6):67008, nov 2007. doi: 10.1209/0295-5075/80/67008. URL <https://doi.org/10.1209/0295-5075/80/67008>.

- Andrew Steane. Multiple-particle interference and quantum error correction. *Proceedings of the Royal Society of London. Series A: Mathematical, Physical and Engineering Sciences*, 452(1954):2551–2577, 1996. doi: 10.1098/rspa.1996.0136. URL <https://royalsocietypublishing.org/doi/abs/10.1098/rspa.1996.0136>.
- T. Stollenwerk, B. O’Gorman, D. Venturelli, S. Mandrà, O. Rodionova, H. Ng, B. Sridhar, E. G. Rieffel, and R. Biswas. Quantum annealing applied to de-conflicting optimal trajectories for air traffic management. *IEEE Transactions on Intelligent Transportation Systems*, 21(1): 285–297, 2020. doi: 10.1109/TITS.2019.2891235. URL <https://doi.org/10.1109/TITS.2019.2891235>.
- Tobias Stollenwerk, Elisabeth Lobe, and Martin Jung. Flight gate assignment with a quantum annealer. In Sebastian Feld and Claudia Linnhoff-Popien, editors, *Quantum Technology and Optimization Problems*, pages 99–110, Cham, 2019. Springer International Publishing.
- Masuo Suzuki. Improved trotter-like formula. *Physics Letters A*, 180(3):232 – 234, 1993. ISSN 0375-9601. doi: [https://doi.org/10.1016/0375-9601\(93\)90701-Z](https://doi.org/10.1016/0375-9601(93)90701-Z). URL <http://www.sciencedirect.com/science/article/pii/037596019390701Z>.
- D Thirumalai, Qiang Li, and T R Kirkpatrick. Infinite-range Ising spin glass in a transverse field. *Journal of Physics A: Mathematical and General*, 22(16):3339–3349, aug 1989. doi: 10.1088/0305-4470/22/16/023. URL <https://doi.org/10.1088/0305-4470/22/16/023>.
- Tony T. Tran, Minh N. Do, Eleanor G. Rieffel, Jeremy Frank, Zhihui Wang, Bryan O’Gorman, Davide Venturelli, and J. Christopher Beck. A hybrid quantum-classical approach to solving scheduling problems. In *SOCS*, 2016. URL <https://www.aaai.org/ocs/index.php/SOCS/SOCS16/paper/view/1395>.
- H. F. Trotter. On the product of semi-groups of operators. *Proceedings of the American Mathematical Society*, 10(4):545–551, 1959. ISSN 00029939, 10886826. doi: <https://doi.org/10.1090/S0002-9939-1959-0108732-6>. URL <http://www.jstor.org/stable/2033649>.
- Guido Van Rossum and Fred L Drake. *Python language reference manual*. Network Theory United Kingdom, 2003.
- Davide Venturelli and Alexei Kondratyev. Reverse quantum annealing approach to portfolio optimization problems, 2019. ISSN 2524-4914. URL <https://doi.org/10.1007/s42484-019-00001-w>.
- Davide Venturelli, Dominic J. J. Marchand, and Galo Rojo. Quantum Annealing Implementation of Job-Shop Scheduling, 2000. URL <https://arxiv.org/abs/quant-ph/1506.08479>. arXiv preprint quant-ph/1506.08479.
- Lorenzo Campos Venuti, Tameem Albash, Daniel A. Lidar, and Paolo Zanardi. Adiabaticity in open quantum systems. *Phys. Rev. A*, 93:032118, Mar 2016. doi: 10.1103/PhysRevA.93.032118. URL <https://link.aps.org/doi/10.1103/PhysRevA.93.032118>.

- Walter Vinci and Daniel A. Lidar. Scalable effective-temperature reduction for quantum annealers via nested quantum annealing correction. *Phys. Rev. A*, 97:022308, Feb 2018. doi: 10.1103/PhysRevA.97.022308. URL <https://link.aps.org/doi/10.1103/PhysRevA.97.022308>.
- Walter Vinci, Tameem Albash, Gerardo Paz-Silva, Itay Hen, and Daniel A. Lidar. Quantum annealing correction with minor embedding. *Phys. Rev. A*, 92:042310, Oct 2015. doi: 10.1103/PhysRevA.92.042310. URL <https://link.aps.org/doi/10.1103/PhysRevA.92.042310>.
- Walter Vinci, Tameem Albash, and Daniel A. Lidar. Nested quantum annealing correction. *npj Quantum Information*, 2(1):16017, 2016. ISSN 2056-6387. URL <https://doi.org/10.1038/npjqi.2016.17>.
- Po-Wei Wang and J. Zico Kolter. Low-rank semidefinite programming for the max2sat problem, 2018. URL <https://arxiv.org/abs/1812.06362>. arXiv:1812.06362.
- Po-Wei Wang, Wei-Cheng Chang, and J. Zico Kolter. The mixing method: low-rank coordinate descent for semidefinite programming with diagonal constraints, 2018. URL <https://arxiv.org/abs/1706.00476>. arXiv:1706.00476.
- Nathan Wiebe and Nathan S Babcock. Improved error-scaling for adiabatic quantum evolutions. *New Journal of Physics*, 14(1):013024, 2012. doi: 10.1088/1367-2630/14/1/013024. URL <https://doi.org/10.1088/1367-2630/14/1/013024>.
- P. Wocjan, D. Janzing, and Th. Beth. Simulating arbitrary pair-interactions by a given hamiltonian: Graph-theoretical bounds on the time complexity, 2001a. URL <https://arxiv.org/abs/quant-ph/0106077>. arXiv:quant-ph/0106077.
- Pawel Wocjan, Martin Roetteler, Dominik Janzing, and Thomas Beth. Universal simulation of hamiltonians using a finite set of control operations, 2001b. URL <https://arxiv.org/abs/quant-ph/0109063>. arXiv:quant-ph/0109063.
- Pawel Wocjan, Martin Rötteler, Dominik Janzing, and Thomas Beth. Simulating hamiltonians in quantum networks: Efficient schemes and complexity bounds. *Phys. Rev. A*, 65:042309, Mar 2002. doi: 10.1103/PhysRevA.65.042309. URL <https://link.aps.org/doi/10.1103/PhysRevA.65.042309>.
- Thomas G. Wong and David A. Meyer. Irreconcilable difference between quantum walks and adiabatic quantum computing. *Phys. Rev. A*, 93:062313, Jun 2016. doi: 10.1103/PhysRevA.93.062313. URL <http://link.aps.org/doi/10.1103/PhysRevA.93.062313>.
- Xin-Chuan Wu, Sheng Di, Emma Maitreyee Dasgupta, Franck Cappello, Hal Finkel, Yuri Alexeev, and Frederic T. Chong. Full-state quantum circuit simulation by using data compression. In *Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis*, SC '19, New York, NY, USA, 2019. Association

- for Computing Machinery. ISBN 9781450362290. doi: 10.1145/3295500.3356155. URL <https://doi.org/10.1145/3295500.3356155>.
- Yu Yamashiro, Masaki Ohkuwa, Hidetoshi Nishimori, and Daniel A. Lidar. Dynamics of reverse annealing for the fully connected p -spin model. *Phys. Rev. A*, 100:052321, Nov 2019. doi: 10.1103/PhysRevA.100.052321. URL <https://link.aps.org/doi/10.1103/PhysRevA.100.052321>.
- Zhi-Cheng Yang, Armin Rahmani, Alireza Shabani, Hartmut Neven, and Claudio Chamon. Optimizing variational quantum algorithms using Pontryagin’s minimum principle. *Phys. Rev. X*, 7:021027, May 2017. doi: 10.1103/PhysRevX.7.021027. URL <https://link.aps.org/doi/10.1103/PhysRevX.7.021027>.
- Theodore J. Yoder, Guang Hao Low, and Isaac L. Chuang. Fixed-point quantum search with an optimal number of queries. *Phys. Rev. Lett.*, 113:210501, Nov 2014. doi: 10.1103/PhysRevLett.113.210501. URL <https://link.aps.org/doi/10.1103/PhysRevLett.113.210501>.
- A. P. Young. Stability of the quantum Sherrington-Kirkpatrick spin glass model. *Phys. Rev. E*, 96:032112, Sep 2017. doi: 10.1103/PhysRevE.96.032112. URL <https://link.aps.org/doi/10.1103/PhysRevE.96.032112>.
- A. P. Young and Helmut G. Katzgraber. Absence of an Almeida-Thouless line in three-dimensional spin glasses. *Phys. Rev. Lett.*, 93:207203, Nov 2004. doi: 10.1103/PhysRevLett.93.207203. URL <https://link.aps.org/doi/10.1103/PhysRevLett.93.207203>.
- Kevin C. Young, Robin Blume-Kohout, and Daniel A. Lidar. Adiabatic quantum optimization with the wrong Hamiltonian. *Phys. Rev. A*, 88:062314, Dec 2013. doi: 10.1103/PhysRevA.88.062314. URL <https://link.aps.org/doi/10.1103/PhysRevA.88.062314>.
- Leo Zhou, Sheng-Tao Wang, Soonwon Choi, Hannes Pichler, and Mikhail D. Lukin. Quantum approximate optimization algorithm: Performance, mechanism, and implementation on near-term devices, 2018. URL <https://arxiv.org/abs/1812.01041>. arXiv preprint arXiv:1812.01041.