Structure, randomness and complexity in quantum computation

Ashley Montanaro

A dissertation submitted to the University of Bristol in accordance with the requirements of the degree of Doctor of Philosophy (PhD) in the Faculty of Engineering, Department of Computer Science, October 2007.

40,000 words

Abstract

This thesis explores the interplay between structure and randomness in quantum computation, with the goal being to characterise the types of structure that give quantum computers an advantage over classical computation. The thesis begins by giving a necessary and sufficient condition for one notion of a quantum walk to be defined on a directed graph, and goes on to derive conditions on the structure of graphs that allow a quantum advantage in a non-local graph colouring game.

A lower bound on entanglement-assisted quantum communication complexity based on information-theoretic ideas is given, and applied to the communication complexity of random functions. New lower bounds on the probability of success of quantum state discrimination are derived, and are applied to the problem of distinguishing random quantum states. This result is used to show a quantum advantage in almost all instances of a bounded-error single-query oracle identification problem.

Lower bounds, and almost optimal algorithms, are given for two models of quantum search of partially ordered sets. This leads to the development of an optimal quantum algorithm to find the intersection of two sorted lists.

Abstractus

Haec thesis colludium explorat inter structuram et accidentiam in computatione quanta, ut genera structurae inscribantur quae computatris quantis opportunitatem dedent super computatione antiqua. Thesis incipit datione condicionem necessariam et satis definire unam notionem ambulatus quanti in directis graphicis formulis, et procedit condiciones derivare structura graphicarum formularum quae opportunitatem quantam permittunt in ludo non locale de graphicis formulis colorandis.

Terminus inferior consequens conceptos scientiae indicii datur complexitati communicationis quantis adiutae implicatione, et complexitatem communicationem attenditur functionum fortuitarum. Termini inferiores novi derivantur probabilitati distinguere rerum quantarum, et problemam attendiuntur rerum quantarum fortuitarum distinguendarum. Hoc proventus usus est docere opportunitatem quantam in paene omnibus exemplis problemae oraculum cognoscandum cum terminato errato et uno quaestione.

Termini inferiores, et algoritmi paene optimi, dantur in duobus exemplibus inquisitionis quantae in collectibus partiliter ordinatibus. Hoc ad algoritmum quantum optimumque factum ducat, quid intersectionem invenit duorum inventariorum compositorum.

Acknowledgements

I would first like to thank my supervisor Richard Jozsa for his help and support throughout this PhD, and in particular for his patience with my "random walk" approach to research.

I thank all my co-authors, and in particular Andreas Winter, who enabled me to achieve an Erdős number of 2, and more importantly a Winter number of 1. Thanks go to all the other members of the Bristol quantum information group and/or computer science department for making my time here so stimulating. In particular, thanks to Sean Clark for helpful and entertaining discussions; Raphaël Clifford for enlightening conversations; Aram Harrow for patient explanations; Dan Shepherd for deep cogitations; and Tony Short for foreign expeditions.

For a great three years in Bristol, thanks to all my other colleagues and friends. And, of course, thanks and love go to Catherine for making my time here infinitely better than I'd ever have thought possible.

This thesis is dedicated to my mother Caroline with deep gratitude for her love and support over the last three years, and the twenty-four before that.

Author's Declaration

I declare that the work in this dissertation was carried out in accordance with the Regulations of the University of Bristol. The work is original, except where indicated by special reference in the text, and no part of the dissertation has been submitted for any other academic award. Any views expressed in the dissertation are those of the author.

SIGNED: .....

DATE: .....

Contents

Chapter 1

Introduction

A quantum computer is a machine designed to use the magical properties of quantum physics to do things that computers built using only the principles of classical physics cannot. In particular, quantum algorithms have been developed that outperform known classical algorithms (the classic example being Shor's integer factorisation algorithm [123], which achieves an exponential speed-up over the best known classical algorithms), and in some cases this improvement is provable.

It is a well-known fact that quantum computers cannot achieve a speed-up greater than a quadratic factor for unstructured search problems [21]. Thus, in order to obtain the exponential speed-ups that we would like, we are led to consider **structured** problems. In the case of integer factorisation, Shor's algorithm relies on a periodicity structure which classical computers cannot (apparently) use. This leads to the general question: what types of structure are useful in quantum computation? That is, what types of structure can quantum computers use in ways that classical computers cannot?

This thesis makes some partial progress towards answering this question. It begins, in Chapter 2, by considering questions related to the structure of graphs, which are fundamental objects both in pure mathematics and in computer science. A basic classical algorithmic tool is the random walk on a graph. This turns out to have a generalisation to the *quantum walk*, which has proven to be a useful tool in the development of quantum algorithms. Indeed, one of the few known provable exponential quantum speed-ups over classical computation is a quantum walk algorithm [38]. However, quantum walks are usually defined on undirected graphs. The main result in this chapter is the development of a necessary and sufficient condition for one notion of quantum walk to be defined on a *directed* graph. In the case where a "traditional" unitary quantum walk cannot be defined on a graph, a generalisation that alternates unitary evolution and measurement is proposed.

Chapter 3 turns to another graph-related task in which the strange properties of quantum mechanics can be beneficial: a distributed graph colouring game. Alice and Bob are separated and have to convince a referee that they have a k-colouring of a

graph G, but are in the unfortunate situation that k is less than the chromatic number of G. Surprisingly, for some graphs, if they have the aid of quantum entanglement they can still trick the referee with certainty. As they have no need to communicate, this is known as a *pseudo-telepathy game* [26]. This game allows the definition of a natural quantum generalisation of the classical chromatic number of a graph. Some properties of this quantum chromatic number are discussed, including limitations on how small the quantum chromatic number can be relative to the classical chromatic number, and the chapter finishes with an example of a small graph with a separation between its quantum and classical chromatic numbers.

We then proceed to considering communication tasks where sharing entanglement does not completely remove the necessity of communication, but might result in less communication being required. This is the realm of *communication complexity*, where shared entanglement has been shown to be advantageous [62, 60], but is far from being fully understood. Chapter 4 presents a new lower bound on the entanglement-assisted quantum communication complexity of total Boolean functions that is based on information-theoretic ideas. The bound unifies several existing bounds, and has an operational interpretation as a method of turning a protocol for computing a function into a communication protocol.

The second theme in the title of this thesis is **randomness**. As well as the intrinsic interest of studying the behaviour of random objects – after all, most objects are random – understanding the properties of a random (i.e. typical) object helps one understand the properties of atypical (i.e. structured) objects. In the case of quantum computation, the natural random objects of study are random quantum states. In Chapter 5, we consider one measure of the information content of a set of states: their global distinguishability. Lower bounds are developed on the ability of a specific measurement – the so-called *pretty good measurement* [68] – to distinguish a set of states. These bounds are then applied to sets of random quantum states, showing that a large number of random states can be distinguished with a constant probability of success. This lower bound uses results from random matrix theory to determine the distribution of the eigenvalues of the Gram matrix of a set of random states.

The relationship between randomness and complexity is exemplified by the useful principle that, for any reasonable measure of randomness, and any reasonable measure of complexity, *random objects are complex*. For example, a random language is undecidable; to compute a random n-bit Boolean function, an exponential-sized circuit is required; a random bit-string is incompressible.

This thesis gives several examples of this principle at work in quantum computation. These include the following:

- A random Boolean function has almost maximal entanglement-assisted quantum communication complexity (Chapter 4).

- A set of $O(n)$ n-dimensional random quantum states can be distinguished with constant probability of success as $n \to \infty$ (Chapter 5).
- Finding an object in a random partially ordered set of n elements requires $\Omega(\sqrt{n})$ quantum queries (Chapter 7).

This provides a link to the final theme of the thesis, **complexity**. Chapter 6 is devoted to an important measure of complexity in quantum computation: the number of queries to an oracle that are required to perform some task, i.e. the *query complexity* of that task. We begin by giving a simple proof of the fundamental $\Omega(\sqrt{n})$ quantum lower bound on unstructured search [21], and show how this can easily be applied to give an exponential lower bound on the quantum query complexity of Boolean satisfiability (SAT) in a natural oracular model.

We then drastically simplify to problems which we can solve using only *one* quantum query, and in particular the *oracle identification problem* [10] of determining, given an unknown "oracle" function picked from a known set of functions, which function we have been given. Some problems of this type that can be solved exactly by a quantum computer with a single query are characterised, and the results of Chapter 5 are used to show that quantum computation has a strong advantage over classical computation in a bounded-error version of this task, in that almost all sets of approximately 2^n Boolean functions on n bits can be distinguished with one query using a quantum computer (with a bounded probability of failure), whereas n queries are required classically.

The final chapter of the thesis investigates the quantum advantage that can be obtained when searching in a data set which is partially structured. Two different models of quantum search in partially ordered sets (posets) are considered. In both models, it is shown that (up to logarithmic factors) the quadratic reduction in query complexity obtained by Grover's algorithm [64] is the best possible for search of any partially ordered set, and quantum algorithms that almost achieve this bound are presented. In one model, we give an almost optimal quantum algorithm for searching forest-like posets; in the other, we give an optimal $O(\sqrt{n})$ quantum algorithm for searching posets derived from $n \times n$ arrays sorted along rows and columns. This leads to an optimal $O(\sqrt{n})$ quantum algorithm for finding the intersection of two sorted lists of n integers.

1.1 Quantum mechanics in a nutshell

This section briefly introduces some basic quantum mechanical notions that will be used throughout the remainder of this thesis, as well as some miscellaneous notation. It is written from an abstract, rather than physical, perspective; for a full introduction, see [108] or [111], and for the background in linear algebra, see [75]. More advanced concepts will be introduced later, where necessary.

Mathematically speaking, a d-dimensional pure quantum state $|\psi\rangle$ is a unit vector in d-dimensional complex space $\mathbb{C}^d$ equipped with the standard inner product (*Hilbert space*). The "ket" notation $|\psi\rangle$ denotes a column vector, as opposed to the conjugate transpose "bra" row vector $\langle\psi|$; so we have (for example) the inner product $\langle\alpha|\beta\rangle = \sum_k \alpha_k^* \beta_k$. A mixed quantum state ρ is a probabilistic mixture of pure states; $\rho = \sum_k p_k |\psi_k\rangle\langle\psi_k|$, where $\sum_k p_k = 1$. It is easy to show that ρ is a Hermitian matrix whose eigenvalues $\{\lambda_i\}$ are non-negative (i.e. ρ is positive semidefinite), and that $\mathrm{tr}(\rho) = \sum_i \lambda_i = 1$.

The von Neumann entropy $S(\rho) = -\sum_i \lambda_i \log \lambda_i$ is the quantum analogue of the Shannon entropy giving a measure of the "subjective uncertainty" of a probability distribution; it can be interpreted as the "information content" of a quantum state. (All logarithms will be taken to base 2 throughout, unless otherwise specified.)

Quantum states combine via the tensor product $\otimes$: $(|\psi\rangle \otimes |\phi\rangle)_{ij} = \psi_i \phi_j$ (the $\otimes$ symbol will often be omitted). A product state $|\psi\rangle$ can be written as $|\psi_A\rangle|\psi_B\rangle$ for some $|\psi_A\rangle$, $|\psi_B\rangle$; a state is called *entangled* if it is not product. The partial trace is the act of "throwing away" a subsystem: $\mathrm{tr}_B\left(|\alpha_A\rangle\langle\beta_A| \otimes |\alpha_B\rangle\langle\beta_B|\right) = \langle\beta_B|\alpha_B\rangle|\alpha_A\rangle\langle\beta_A|$.

In this thesis, the only operations applied to quantum states will be unitary evolution $|\psi\rangle \to U|\psi\rangle$ where $UU^\dagger = I$, measurement, and the partial trace. The most general form of measurement is given by a POVM (positive operator valued measure), namely a set of positive semidefinite operators $M = \{\mu_i\}$ with $\sum_i \mu_i = I$. Each operator is associated with a measurement outcome; the probability of obtaining measurement outcome i when measurement M is applied to state ρ is $\mathrm{tr}(\mu_i \rho)$. If we specialise to rank 1 projective measurement operators $\mu_i = |\nu_i\rangle\langle\nu_i|$ then the probability of outcome i is $|\langle\nu_i|\psi_i\rangle|^2$.

The fundamental object in quantum computation is the quantum bit, or *qubit*, the quantum analogue of the classical bit. A qubit is simply a two-dimensional quantum system where we define a "computational" basis $\{|0\rangle, |1\rangle\}$ corresponding to the classical $\{0, 1\}$. The ability for quantum systems to be in a superposition of basis states allows a qubit to be "both zero and one at the same time", which is an essential part of *quantum algorithms* that operate on systems of many qubits to achieve speed-ups over classical algorithms.

A quantum algorithm can be described by a quantum circuit, which (analogously to a classical circuit) is a sequence of elementary "quantum gates" (unitary operators) applied to a starting state and terminating with a measurement whose result is the output of the algorithm. We generally look for efficient quantum algorithms, where the number of gates is polynomial in the size of the input.

1.2 Previous publications

Much of this thesis has been published previously and some of it is joint work.

- Chapter 2 has been published previously as "Quantum walks on directed graphs", *Quantum Information and Computation* vol. 7 no. 1, pp. 93–102 (`quant-ph/0504116`[1]).
- Chapter 3 is joint work with Peter Cameron, Mike Newman, Simone Severini and Andreas Winter, and has been published previously as "On the quantum chromatic number of a graph", *Electronic Journal of Combinatorics* vol. 14 no. 1 (`quant-ph/0608016`).
- Chapter 4 is joint work with Andreas Winter and has been published previously as "A lower bound on entanglement-assisted quantum communication complexity", in the proceedings of ICALP 2007, pp. 122–133 (`quant-ph/0610085`).
- The majority of Chapter 5 has been published previously as "On the distinguishability of random quantum states", *Communications in Mathematical Physics* vol. 273 no. 3, pp. 619-636 (`quant-ph/0607011`).
- The portions of Chapter 6 relating to exact single-query oracle identification are joint work with Richard Jozsa.
- The majority of Chapter 7 is available as the pre-print "Quantum search of partially ordered sets" (`quant-ph/0702196`).

[1] "`quant-ph/xxx`" identifiers refer to the quant-ph arXiv e-print server `http://arxiv.org/`.

Chapter 2

Quantum walks on directed graphs

2.1 Introduction

Quantum walks are the quantum counterpart of classical random walks. Random walks play an important part in classical computer science, and it seems plausible that quantum walks could be equally important in the study of quantum computation. Quantum walks on undirected graphs have been defined using two different formulations (discrete-time [4] and continuous-time [54]), and are known to exhibit markedly different behaviour to classical random walks [86, 39]. Quantum walks have been used to produce novel quantum algorithms [122, 38, 8] displaying speed-ups over their classical equivalents. A natural question arises: can a quantum walk be defined on a *directed* graph? If so, which directed graphs allow a reasonable definition?

As motivation for this, there are many problems in graph theory that are known or suspected to be more difficult to solve for directed graphs than undirected graphs (an example being Reachability [110], c.f. Section 2.7 below). It is interesting to ask whether quantum walk algorithms can provide any straightforward improvement over classical algorithms for such problems.

The continuous-time formulation of quantum walks is defined by introducing a quantum system whose Hamiltonian is based on the adjacency matrix of the graph. This will not be suitable for walks on directed graphs, as this matrix will not in general be Hermitian, and hence the evolution of the system will not be unitary. Therefore, this chapter will only consider the discrete-time formulation, which consists of the iterated application of a unitary operator based on the structure of the graph.

We give a necessary and sufficient condition – which we term *reversibility* – on a graph for it to allow the definition of a discrete-time quantum walk that respects its structure. We then discuss the implications of this result. If a directed graph does

not allow the definition of a "fully quantum" walk that preserves coherence throughout, we provide a method for defining a walk that alternates unitary evolution and measurement, and still allows for a level of coherence to be maintained.

This chapter has been published previously as "Quantum walks on directed graphs", *Quantum Information and Computation* vol. 7 no. 1, pp. 93–102 (`quant-ph/0504116`).

2.2 Quantum walks on graphs

We begin with some standard graph-theoretic definitions that will be used throughout this chapter.

2.2.1 Graphs

A *graph* (or *digraph*; we will use the terms interchangeably) G is defined here as a set of vertices V and arcs A, where A is a set of ordered pairs of vertices. The i'th vertex is labelled v_i ($1 \leq i \leq |V|$). We assume that there may be at most one arc in each direction between each two vertices. An *undirected graph* has the further property $(v_i, v_j) \in A \Leftrightarrow (v_j, v_i) \in A$. When $(v_i, v_j) \in A$, we say that v_i is *connected to* v_j (or that there is *an arc between* v_i and v_j), and use the notation $v_i \to v_j$. We sometimes say that there is *an undirected edge between* v_i and v_j if $v_i \to v_j$ and $v_j \to v_i$. We say that G is *connected* if for every pair of vertices (v, w) there is a sequence of vertices $v_1, v_2, \ldots, v_k$ such that $v = v_1$, $w = v_k$, and each consecutive pair of vertices is connected by an arc (in either direction, which may vary along the sequence).

The *out-neighbours* of a vertex v_i are the vertices to which v_i is connected; similarly, the *in-neighbours* of v_i are the vertices that are connected to v_i. The *in-degree* and *out-degree* of v_i are the number of in-neighbours and out-neighbours it has, respectively. Every vertex in a *d-regular* graph has d in-neighbours and d out-neighbours. A *subgraph* G' of a graph G is a graph whose sets of vertices and arcs are subsets of those in G. A *connected component* C of G is a connected subgraph of G such that C does not remain connected if any further vertices of G are added to C. A *path* is an ordered list of vertices $\{v_1, v_2, \ldots\}$ where $v_{i-1} \to v_i$, for all $i > 1$. A *cycle* is a path whose final vertex is the same as its initial vertex.

The adjacency matrix of G is the matrix also called G, where $G_{ij} = 1 \Leftrightarrow j \to i$. The support of a matrix U is the matrix U', where $U'_{ij} = 0$ if $U_{ij} = 0$, and $U'_{ij} = 1$ otherwise. The (di)graph of a unitary matrix U is the graph whose adjacency matrix is the support of U.

2.2.2 Quantum walks

A *coined quantum walk* on a d-regular undirected graph G, as defined in [4], is produced by creating a Hilbert space $\mathcal{H}_v$ of dimension $|G|$ (where $|G|$ is the number of vertices in G), and identifying a basis state with each vertex. Each arc leaving a vertex is labelled by an integer from 0 to $d-1$. This space is then augmented with a "coin" Hilbert space $\mathcal{H}_c$ of dimension d to give an expanded space $\mathcal{H}_c \otimes \mathcal{H}_v$. A "coin toss" operator C is defined, which operates only on $\mathcal{H}_c$. A "shift" operator S is also defined, such that $S|c\rangle|v_i\rangle = |c\rangle|v_j\rangle$, where v_j is the vertex at the other end of the arc from v_i labelled by c. One step of the walk then consists of applying the unitary $S(C \otimes I)$ – i.e. a coin toss followed by a shift.

Several methods, resulting in potentially different dynamics, can be used to define a coined quantum walk on an irregular graph. Multiple coins may be used [86] (a d-dimensional coin for each vertex of degree d); a single coin of the same dimension as the maximum degree of any vertex in G may be used, and restricted to a d-dimensional subspace at each vertex of degree d [127]; alternatively, self-loops may be added to low-degree vertices to make the graph regular [86].

We now define a more general notion of a discrete-time quantum walk, using a similar definition to [4].

Definition 2.2.1. *A discrete-time quantum walk is the repeated application of a unitary operator W, where each application of W is one step of the walk. To define a quantum walk on a graph G, we identify a finite set of one or more basis states $\{|v_i^1\rangle, |v_i^2\rangle, \ldots\}$ with each vertex v_i of the graph. We say a quantum walk can be implemented on G if there exists a W such that, for all i, j, $v_i \to v_j$ if and only if there exist k, l such that $\langle v_j^k|W|v_i^l\rangle \neq 0$. We assume that G has self-loops at each vertex.*

2.3 Reversible and irreversible graphs

Definition 2.3.1. *An arc $a \to b$ is called* reversible *if there is a path from b to a. A graph whose arcs are all reversible is also called reversible; otherwise, it is called* irreversible.

Consider the following examples. A graph containing at least one source or sink is irreversible. All undirected graphs are reversible. An *Eulerian* graph is a graph whose every vertex has equal in-degree and out-degree. All Eulerian graphs are reversible, as they admit Eulerian tours (a cycle that visits every vertex, and traverses each arc once). Thus, all regular graphs are reversible. A *Cayley* graph is a graph associated with a group X and a set of generators Y, whose vertices are the elements of X, and which contains an arc $v_a \to v_b$ if and only if the associated element $b = ac$, for some $c \in Y$. All Cayley graphs are regular, and hence are reversible.

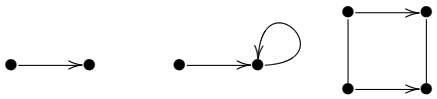


Figure 2.1: Some irreversible graphs. An undirected edge represents an arc in both directions.

In the language of graph theory [37], a reversible graph is a graph whose every connected component is *strongly connected* (a graph is strongly connected if there is a path from any vertex in the graph to any other vertex). A reversible graph is almost the same as the transition graph of an irreducible Markov chain. However, there is a minor difference in that the definition here allows a graph to have multiple disconnected components, whereas irreducible Markov chains do not. So-called "time reversible" Markov chains are quite different, referring to a Markov chain which is symmetric in time [85].

The first new result in this chapter is the following theorem. The proof will be given in Section 2.5 below.

Theorem 2.3.2. *A discrete-time quantum walk can be defined on a finite graph G if and only if G is reversible.*

Corollary 2.3.3. *The digraph of a unitary matrix is reversible.*

This corollary is simply the special case where each vertex of the graph is identified with one basis state.

2.3.1 Determining reversibility

How easily can reversibility be determined for a given graph? On the one hand, it is clear that reversibility is a *global* attribute of a graph: it is not possible to determine whether a given arc is reversible without potentially considering all the other arcs in the graph. On the other hand, reversibility of a graph can be determined very efficiently. Assume that a graph $G = (V, E)$ is given as an adjacency list. Then [45] gives algorithms based on depth-first search which can be used to first decompose G into connected components, and then to decompose each connected component into strongly connected components. G is clearly only reversible if there is one strongly connected component for each connected component. Each of these two steps can be performed in time $O(|V| + |E|)$ for an overall runtime which is linear in the number of edges.

2.4 Previous work

A *directed bridge* is an arc in a graph G whose removal would increase the number of connected components of G. Severini has proven [121] that the digraph of a unitary

matrix does not contain any directed bridges. The reversibility condition given here is stronger, as a graph containing two connected components with multiple arcs between them, all going in the same direction, is irreversible. It has also been shown [119] that the digraph of a unitary matrix is strongly quadrangular.

The notion of a coined quantum walk and the term "quantum random walk" were introduced by Aharonov et al. [4], and quantum walks were used by Watrous to simulate classical random walks [127]. A systematic study of the potential algorithmic applications of discrete-time quantum walks was initiated by [4]. There have been many results on coined quantum walks since; see the survey [86] for details. In particular, Severini has shown that the underlying digraph of a coined quantum walk is a line digraph [120]. With the result given here, this implies that a line digraph is reversible. Lopez Acevedo and Gobron [99] have considered the classification of quantum walks on Cayley graphs.

Szegedy has developed [124] a method for defining quantum walks based on arbitrary Markov chains, which has recently been generalised and simplified by Magniez et al. [100]. This approach produces a quantum walk from any Markov chain, including chains whose transition graphs are irreversible. However, for such graphs the walk produced will not respect the graph's structure: there will be some probability to travel between basis states corresponding to vertices that are not connected by an arc in the correct direction.

2.5 Proof of Theorem 2.3.2

2.5.1 Necessity

Our definition of a quantum walk consists of an identification of states with vertices of a graph. We will show that, if it is possible to "walk" from state $|a\rangle$ to state $|b\rangle$ by performing a unitary operation W, it is also possible to walk from $|b\rangle$ to $|a\rangle$ by performing W a positive number of times. (Without this positivity condition, the problem is trivial, as multiplying by W^{-1} will perform the reverse of W.)

Lemma 2.5.1. *For any vector $|a\rangle$ in a finite-dimensional Hilbert space, any unitary operator W, and any $\epsilon > 0$, there exists $n \geq 1$ such that $|\langle a|W^n|a\rangle| > 1 - \epsilon$.*

Proof. This is simply a restatement in the terminology of unitary operators of the Quantum Recurrence Theorem proved by Bocchieri and Loinger in the language of wavefunctions [24], which in turn is a quantum equivalent of Poincaré's recurrence theorem for classical mechanics from 1890. □

The implication of this lemma is that repeating the same unitary operator enough times on $|a\rangle$ will produce a state arbitrarily close to $|a\rangle$.

Lemma 2.5.2. *For any vectors $|a\rangle$, $|b\rangle$ in a finite-dimensional Hilbert space, and for any unitary operator W, if $\langle b|W|a\rangle \neq 0$, then there exists $m \geq 0$ such that $\langle a|W^m|b\rangle \neq 0$.*

Proof. First, we have $\langle b|W|a\rangle \neq 0 \Rightarrow \langle a|W^{-1}|b\rangle \neq 0$. Consider a state close to $|b\rangle$, denoted by $|b'\rangle$. For sufficiently small ϵ, $|\langle b|b'\rangle| > 1 - \epsilon \Rightarrow \langle a|W^{-1}|b'\rangle \neq 0$. By Lemma 2.5.1, for arbitrarily small $\epsilon' > 0$, there exists $p \geq 1$ such that $|\langle b|W^p|b\rangle| > 1 - \epsilon'$. Set $|b'\rangle = W^p|b\rangle$ and we have $\langle a|W^{-1}W^p|b\rangle \neq 0$, and hence $\langle a|W^m|b\rangle \neq 0$ for $m = p - 1 \geq 0$. □

Lemma 2.5.3. *Let W be a quantum walk defined on a graph G with vertices $\{v_1, v_2, \ldots\}$ by associating basis states $\{|v_i^1\rangle, |v_i^2\rangle, \ldots\}$ with each vertex v_i. Then if, for some k and l, and some $n \geq 0$, $\langle v_j^l|W^n|v_i^k\rangle \neq 0$, there is a path from v_i to v_j.*

Proof. $W^n|v_i^k\rangle$ describes n steps of the walk starting from state $|v_i^k\rangle$, and hence produces a superposition over possible paths of length n that the walk can take from vertex v_i. If $\langle v_j^l|W^n|v_i^k\rangle \neq 0$ for some k, l, this implies that at least one of these paths must reach vertex v_j. For the case $n = 0$, $\langle v_j^l|W^0|v_i^k\rangle \neq 0$ only if $v_i = v_j$, as expected. □

Lemma 2.5.2 shows that, if there is some amplitude to travel from some basis state $|v_i^k\rangle$ to some basis state $|v_j^l\rangle$ after 1 step of the walk, there must also be some amplitude to travel from $|v_j^l\rangle$ to $|v_i^k\rangle$ after some $m \geq 0$ steps of the walk. With Lemma 2.5.3, this shows that, if there is an arc from the corresponding vertex v_i to v_j, then there is a path from v_j to v_i, and hence the necessity of Theorem 2.3.2 is proven.

2.5.2 Sufficiency

We will show that a coined quantum walk can be defined on any reversible graph. As defined in Section 2.2.2, we will use a Hilbert space $\mathcal{H}_v$ which associates one basis state with each vertex of the graph, augmented with a "coin" space $\mathcal{H}_c$. Our construction will be determined by the cycles in the graph.

Lemma 2.5.4. *Every arc in a reversible graph G is included in at least one cycle.*

Proof. Let $v_i \to v_j$ be any arc in G. Since G is reversible, there is a path from v_j to v_i, and hence there exists a cycle that includes the given arc. □

This shows that it is possible to find a set $\{c_1, c_2, \ldots\}$ of cycles in G such that every arc in G is included in at least one cycle. Each cycle c_i gives rise to a permutation P_i as follows. If a vertex v is in the cycle with arc $v \to v'$, then $P_i(v) = v'$; otherwise, $P_i(v) = v$.

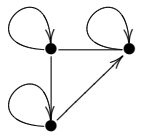
We then associate one coin basis state with each permutation, and select a coin operator C as in the standard definition of a coined quantum walk (Section 2.2.2). In

the study of quantum walks on undirected graphs, it has been found that the choice of coin operator may have a significant effect on the dynamics of a quantum walk [104, 125, 90], and it is a non-trivial question to determine the "optimal" coin (in terms of the desired dynamics) for a given graph. The same applies here.

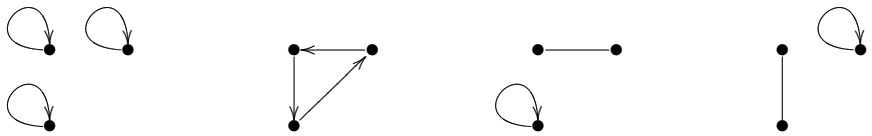
The quantum walk operator W, which operates on $\mathcal{H}_c \otimes \mathcal{H}_v$, can then be expressed as

$$W = \left(\sum_i |i\rangle\langle i| \otimes P_i \right) (C \otimes I) \tag{2.1}$$

This proves that reversibility is a sufficient condition for the definition of a quantum walk. As an example of this construction, consider the following directed graph with labelled vertices. (Recall that self-loops are always included at every vertex.)



This graph admits the following four cycles, each augmented by self-loops at vertices not included in the cycle. Between them, these include every arc in the graph.



We can now use a four-dimensional coin space to select between these four cycles. This example illustrates the fact that, depending on the structure of the graph in question, this algorithm may require a number of coin basis states exceeding the number of vertices in the graph. However, the number of coin states need never exceed the number of arcs. Also note that, for some graphs, the number of coin states used can be reduced by combining disjoint cycles into a single permutation.

2.6 Simulating irreversible arcs with measurement

There appears to be an intuitive correspondence between walking on a reversible graph and the reversibility of unitary evolution. Can we take this analogy a step further and define a quantum walk on a graph containing irreversible arcs by making use of the irreversible process of measurement? It turns out to be possible to define a "partially quantum" walk that maintains some quantum coherence in the reversible portions of the graph.

We will first define what is meant by "reversible portions" of a graph. Consider a subgraph G' of a graph G. G' is called a *reversible subgraph* of G if, considered as a graph itself, G' is reversible.

Lemma 2.6.1. *Let G^{rev} be the subgraph of G whose arcs consist of all the reversible arcs of G. Then G^{rev} is reversible.*

Proof. For every arc $v_i \to v_j$ in G^{rev}, we require there to exist a path from v_j to v_i. But this will be the case, because there is a path in G from v_j to v_i. Every arc in this path is reversible, and hence will be included in G^{rev}'s set of arcs. □

Lemma 2.6.2. *It is possible to partition any graph G into reversible subgraphs such that the arcs in G that connect different subgraphs are all irreversible.*

Proof. Consider the connected components of G^{rev}, which are clearly reversible subgraphs of G. By definition, these do not contain any irreversible arcs. All the irreversible arcs in G must therefore connect vertices in different reversible subgraphs of G. □

One possible way of defining a walk on an irreversible graph G is the following approach. Informally, we consider G as consisting of the connected components of G^{rev} "patched together" with irreversible arcs. We produce a set of quantum walk operators, each corresponding to one component of G^{rev}. The irreversible arcs of G are then simulated by replacing them with undirected edges. If such an edge is traversed by the "walker", we change to a different walk operator to ensure that it cannot be traversed in the opposite direction.

More specifically, consider vertices v_1 and v_2 that are in different reversible subgraphs of G (called C_1 and C_2 respectively), and consider an irreversible arc $v_1 \to v_2$. This arc can be simulated by the following two-step process. First, we perform an incomplete measurement to determine whether the walker is in C_1 or C_2. Then, if it is in C_1, we perform one step of a quantum walk defined on the graph consisting of C_1 augmented with an undirected edge $v_1 \leftrightarrow v_2$. Alternatively, if the walker is in C_2, we perform one step of a walk only defined on the graph C_2. This ensures that the irreversible arc cannot be traversed in the wrong direction.

A more formal definition of this algorithm is given below.

2.6.1 Algorithm to produce a partially quantum walk

1. Produce a set of reversible subgraphs of G (Lemma 2.6.2) using the algorithm of Section 2.3.1.

2. Create a set of reversible graphs $\{G_1, G_2, \ldots\}$ from the set of reversible subgraphs of G. These graphs partition all the vertices of G. Consider a Hilbert space $\mathcal{H}_v$ labelled by the vertices, and let M be the incomplete measurement that projects onto this partition. Thus one measurement outcome corresponds to each reversible subgraph.

3. Consider each graph in turn, denoting the graph under consideration G_i. Some graphs G_i will contain vertices that were the heads of irreversible arcs in G. Augment each graph G_i with undirected links from these vertices to the corresponding targets of the arcs. Each of these links corresponds to moving to a new reversible subgraph. Call each augmented graph G_i'.

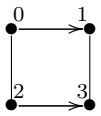
4. Define a coined quantum walk W_i on each graph G_i', using the approach of Section 2.5.2.

5. We now have a set of quantum walks, each operating on a subgraph of the original graph. The overall walk consists of repeatedly alternating the measurement M and one of the unitary walk operators. We perform measurement M, and if we see outcome i, we perform one step of the walk W_i.

This approach has the advantage that it preserves coherence within each reversible subgraph; however, coherence across reversible subgraphs is not possible. That is, it is impossible to maintain a coherent superposition of states corresponding to vertices in two different subgraphs. An obvious implication of this is that a quantum walk on a graph whose arcs are all irreversible will be the same as the equivalent classical random walk.

2.6.2 Example of the algorithm operating on an irreversible graph

Consider the following labelled irreversible graph G and its adjacency matrix. Self-loops are not shown here but should be considered to be present.



$$\begin{pmatrix} 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 \end{pmatrix}$$

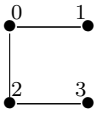
We can split the graph into reversible subgraphs R_1 and R_2 consisting of the vertices $\{0, 2\}$ and $\{1, 3\}$, joined by irreversible arcs $0 \to 1$ and $2 \to 3$. These reversible subgraphs have adjacency matrices

$$R_1 = \begin{pmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix} \text{ and } R_2 = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix}$$

Define an incomplete projective measurement M that distinguishes between R_1 and R_2. This measurement is made up of the operators

$$M_1 = |0\rangle\langle 0| + |2\rangle\langle 2| \text{ and } M_2 = |1\rangle\langle 1| + |3\rangle\langle 3|$$

Then augment R_1 with undirected links corresponding to the irreversible arcs to R_2. This graph, denoted here by R_1', is still reversible and allows the definition (omitted) of a coined quantum walk W_1. The subgraph R_2 does not need augmenting, as it does not contain the heads of any irreversible arcs, and a quantum walk W_2 can be defined on it directly.

$R_1' =$ 

$= \begin{pmatrix} 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}$

A quantum walk on G then consists of repeating the following steps. First, perform the measurement M to determine whether the walker is in R_1 or R_2. If the measurement outcome is M_1, perform one step of the walk W_1 on the graph R_1'. Otherwise, perform one step of the walk W_2 on the graph R_2. Note that, if the walk is begun with a superposition corresponding to being at vertices 0 and 2, and outcome M_2 is measured after one step, this superposition is translated to a superposition of vertices 1 and 3 in R_2: so in this case quantum coherence is preserved.

2.7 The Reachability problem for directed graphs

Reachability (also known as s-t Connectivity or Path) is the problem of deciding whether, for two vertices s and t in a directed graph, there is a path from s to t. In the context of classical algorithms, the problem is suspected to be more difficult than its undirected variant; in fact, it is NL-complete [110], whereas undirected Reachability is in L [115]. On reversible directed graphs, the problem reduces to undirected connectivity. This is clear from the following lemma:

Lemma 2.7.1. *In any connected reversible graph G, there is a path from every vertex a to every other vertex b.*

Proof. Immediate from the definition of strong connectivity in [37]. To see this explicitly, note that since G is connected, any vertex a may be linked to any other vertex b by a sequence of arcs $v_i \to v_j$ whose directions may vary along the sequence. Since G is reversible, v_i is also reachable from v_j for each arc, so a and b are reachable from each other in either direction. □

Thus, every vertex within each connected component of a reversible graph is reachable from every other vertex in that component, exactly as in undirected graphs. This implies that Reachability can be solved for reversible graphs by ignoring the direction of arcs and treating the graph as undirected. Theorem 2.3.2 therefore suggests that quantum walk algorithms of the type discussed in this chapter may not be much help in solving Reachability, because the only graphs on which such algorithms may be defined are exactly those for which the problem is already easy.

It is also worth noting that there are many other classical random walk algorithms which perform a search on directed graphs (an example being Schöning's random walk algorithm for SAT [117]). These often work by traversing a directed graph randomly until they reach a sink, which represents a previously unknown solution. Since such graphs are not reversible, the main result of this chapter shows that quantum walk algorithms for such problems may not be merely straightforward generalisations of their classical counterparts.

Chapter 3

The quantum chromatic number

3.1 Introduction

From the fundamental question of whether a graph is connected, we now turn to another basic property of a graph: its chromatic number. A *colouring* of an undirected graph G is an assignment of colours to the vertices such that adjacent vertices have different colours. The chromatic number $\chi(G)$ is defined as the minimum number of colours required to properly colour G. We consider here a quantum generalisation of a distributed graph colouring problem.

Namely, Alice and Bob want to convince a referee with probability 1 that they have a c-colouring of a graph $G = (V, E)$ in the interrogation model: they each get asked a vertex v, w of the graph, respectively, and have to report back a colour α, β (resp.) to the referee (i.e. a number in $[c] = \{1, 2, \ldots, c\}$). If $v = w$, then to pass they have to reply the same: $\alpha = \beta$; if $(v, w) \in E$, then to pass they have to reply differently: $\alpha \neq \beta$.

If they are not allowed to talk to each other during the interrogation but may agree on a strategy beforehand, then it is straightforward to see that they can win with probability 1 if and only if $c \geq \chi(G)$ – that is, in a classical world where Alice and Bob may share randomness and an otherwise deterministic strategy. However, if Alice and Bob share an entangled state (possibly depending on the graph), there are graphs for which Alice and Bob can win this game with probability 1 for $c < \chi(G)$. Based on a suggestion of Patrick Hayden (see [14]) we call the smallest c such that Alice and Bob can win the graph colouring scheme the *quantum chromatic number*.

Such a problem was first considered in [31, 27], and generalised in [128], Theorems 8.5.1-3, and [30], for Hadamard graphs: the vertices are n-bit strings, and two of them are joined by an edge if and only if their Hamming distance is $n/2$. In these references it is shown that the game can be won with $c = n$ colours. This line of investigation was carried further under the heading "pseudo-telepathy" in [58, 57, 26, 14]. Earlier work of Frankl and Rödl [56] in extremal combinatorics established that the chromatic

number of the Hadamard graphs grows exponentially in n. In [106] it is shown that the chromatic number is equal to n if and only if $n \in \{1, 2, 4, 8\}$.

The rest of the chapter is structured as follows: in Section 3.2 we present the model (strictly speaking, an infinite hierarchy of models) for the quantum chromatic number. Then we go on to general properties of the quantum chromatic number in Section 3.3, bounds via orthogonal representations (Section 3.4), restrictions on the quantum chromatic number when it is low (Section 3.5), and finish with an example of a small graph that demonstrates a separation between classical and quantum chromatic number (Section 3.6).

The results in this chapter are joint work with Peter Cameron, Mike Newman, Simone Severini and Andreas Winter, and have been published previously as "On the quantum chromatic number of a graph", *Electronic Journal of Combinatorics* vol. 14 no. 1 (`quant-ph/0608016`).

3.2 Model(s)

The most general strategy for Alice and Bob to win the graph colouring game with probability 1 with c colours for a graph $G = (V, E)$ consists of an entangled state $|\psi\rangle_{AB} \in \mathbb{C}^{d\times d}$ shared between them, and two families of POVMs $(E_{v\alpha})_{\alpha=0,\ldots,c-1}$ and $(F_{v\beta})_{\beta=0,\ldots,c-1}$, indexed by the vertices $v \in V$ of the graph. The fact that they win with probability 1 is expressed by the consistency condition

$$\forall v \in V\ \forall \alpha \neq \beta \qquad \langle\psi|E_{v\alpha} \otimes F_{v\beta}|\psi\rangle = 0, \tag{3.1}$$

$$\forall (v, w) \in E\ \forall \alpha \qquad \langle\psi|E_{v\alpha} \otimes F_{w\alpha}|\psi\rangle = 0. \tag{3.2}$$

Note that the dimension d bears no relationship to c, that the entangled state $|\psi\rangle$ can be anything (it may even be mixed but it is immediate that w.l.o.g. we may assume it to be pure), and the POVMs may have operators of arbitrary rank.

The smallest possible c for which Alice and Bob can convince the referee with certainty, i.e. such that eq. (3.1) holds, is called the *quantum chromatic number* of G, which will be denoted by $\chi_q(G)$.

Theorem 3.2.1. *To win the graph colouring game in the above setting, w.l.o.g. the state is maximally entangled, and the POVM elements are all projectors, all w.l.o.g. of the same rank.*

Proof. Without loss of generality we can assume that $|\psi\rangle$ has full Schmidt rank d since otherwise we restrict all POVMs to the supports of the respective reduced states. From eq. (3.1) we get, for any $v \in V$, any α and $\beta \neq \alpha$, that $E_{v\alpha} \perp \mathrm{tr}_B\big((I \otimes F_{v\beta})|\psi\rangle\langle\psi|\big)$,

hence

$$E_{v\alpha} \perp \sum_{\beta\neq\alpha} \mathrm{tr}_B\big((I\otimes F_{v\beta})|\psi\rangle\langle\psi|\big) = \mathrm{tr}_B\big((I\otimes I - I\otimes F_{v\alpha})|\psi\rangle\langle\psi|\big).$$

From this, and because Alice needs to get outcome α with certainty if Bob gets α, we must have

$$E_{v\alpha} = \mathrm{supp}\ \mathrm{tr}_B\big((I\otimes F_{v\alpha})|\psi\rangle\langle\psi|\big).$$

By the same argument all $F_{v\beta}$ are projectors.

Now we argue that the consistency requirement for state $|\psi\rangle$ implies that it is also true when we substitute the maximally entangled state $|\Phi_d\rangle$: in its Schmidt basis, $|\psi\rangle = \sum_i \sqrt{\lambda_i}|i\rangle|i\rangle$, and denoting $\rho = \mathrm{tr}_B\,|\psi\rangle\langle\psi| = \sum_i \lambda_i|i\rangle\langle i| = \mathrm{tr}_A\,|\psi\rangle\langle\psi|$, the finding of the previous paragraph can be cast as

$$E_{v\alpha} = \mathrm{supp}\,\sqrt{\rho}\big(\overline{F_{v\alpha}}\big)\sqrt{\rho}, \quad F_{w\beta} = \mathrm{supp}\,\sqrt{\rho}\big(\overline{E_{w\beta}}\big)\sqrt{\rho}. \tag{3.3}$$

This implies however

$$E_{v\alpha}\rho E_{v\beta} = 0$$

for all v and $\alpha \neq \beta$ (where we cancelled $\sqrt{\rho}$'s left and right), and likewise for $F_{w\alpha}$. But with the fact that each $E_{v\alpha}$ is a projector and that summed over α they yield the identity, this gives (for arbitrary v)

$$\rho = \sum_{\alpha,\beta} E_{v\alpha}\rho E_{v\beta} = \sum_{\alpha} E_{v\alpha}\rho E_{v\alpha},$$

from which it follows that ρ commutes with all the operators $E_{v\alpha}$, and likewise $F_{w\beta}$ [93]. So we might as well make ρ a multiple of the identity, and hence a maximally entangled state. Additionally, we find

$$E_{v\alpha} = \overline{F_{v\alpha}}, \quad F_{w\beta} = \overline{E_{w\beta}}. \tag{3.4}$$

Finally, we show how to make the operators all the same rank: let $|\psi'\rangle = |\psi\rangle \otimes |\Phi_c\rangle$, and

$$E'_{v\alpha} = \sum_{i=0}^{c-1} E_{v,\alpha+i} \otimes |i\rangle\langle i|, \quad F'_{w\beta} = \sum_{i=0}^{c-1} F_{w,\beta+i} \otimes |i\rangle\langle i|, \tag{3.5}$$

where the colours are w.l.o.g. $\{0,\ldots,c-1\}$ and the additions above are modulo c. These states and operators evidently still make for a valid quantum colouring, and also clearly all operators have now the same rank. □

This proposition motivates us to introduce rank-r versions of the quantum chromatic number: we define $\chi_q^{(r)}(G)$ as the minimum c such that Alice and Bob can win the graph colouring game for G with a maximally entangled state of rank rc, and POVMs with operators of rank r (exactly).

The special case of the rank-1 model is the case where Alice and Bob share a c-dimensional maximally entangled state

$$|\Phi_c\rangle = \frac{1}{\sqrt{c}} \sum_i |i\rangle_A |i\rangle_B.$$

To make their choices, they both use rank 1 von Neumann measurements, which are ordered bases $(|e_{v\alpha}\rangle)_\alpha$ and $(|f_{v\beta}\rangle)_\beta$ for all vertices v, for Alice and Bob, respectively. At this point we can argue easily that $\chi_q^{(1)}(G) \leq \chi(G)$, as follows. Take a colouring $\gamma : G \longrightarrow \{0, \ldots, c-1\}$ of G with $c = \chi(G)$ colours, and let Alice and Bob share the maximally entangled state $|\Phi_c\rangle$. Their measurements are simply permutations of the standard basis:

$$|e_{v\alpha}\rangle = |\alpha + \gamma(v) \mod c\rangle, \quad |e_{w\beta}\rangle = |\beta + \gamma(w) \mod c\rangle.$$

We now make several observations regarding Alice and Bob's selection of bases. First, Bob's bases are tied to Alice's by the demand of consistency: we need, for all v and α,

$$\langle e_{v\alpha}| \langle f_{v\alpha}|\Phi_c\rangle = 1/\sqrt{c}, \tag{3.6}$$

which enforces the condition of (3.4) that

$$|f_{v\alpha}\rangle = \overline{|e_{v\alpha}\rangle}. \tag{3.7}$$

This means that we can translate the colouring condition into something that only concerns Alice's bases: we need, for all $(v, w) \in E$ and all α,

$$\langle e_{v\alpha}| \langle f_{w\alpha}|\Phi_c\rangle = 0.$$

Because of eqn. (3.6) this can be rewritten as

$$\forall (v, w) \in E \text{ and } \forall \alpha \quad \langle e_{v\alpha}|e_{w\alpha}\rangle = 0. \tag{3.8}$$

It is convenient to introduce unitary matrices U_v for each vertex v, whose columns are just the vectors $|e_{v\alpha}\rangle$, $\alpha = 1, \ldots, c$. Then we can reformulate Alice's strategy as follows: on receiving the request for vertex v, she performs the unitary $U_v^\dagger$ on her quantum system and measures in the standard basis to get a number $\alpha \in [c]$. By eqn. (3.6) above, Bob, for vertex w, performs the unitary $\overline{U_w}^\dagger = U_w^\top$ and measures in the standard basis to obtain $\beta \in [c]$. So we can rewrite the colouring condition expressed in eqn. (3.8) as:

$$\forall (v, w) \in E \quad U_v^\dagger U_w \text{ has only zeroes on the diagonal.} \tag{3.9}$$

By a similar chain of arguments we can show, for the POVM constructed in the proof

of Theorem 3.2.1, that $F_{v\alpha} = \overline{E_{v\alpha}}$ for all vertices v and all colours α, and that hence the colouring condition can be phrased entirely in terms of Alice's operators:

$$\forall (v,w) \in E \text{ and } \forall \alpha \quad E_{v\alpha} E_{w\alpha} = 0, \tag{3.10}$$

i.e. $E_{v\alpha}$ and $E_{w\alpha}$ are orthogonal.

We note here that the following question remains open: whether $\chi_q^{(1)}(G) = \chi_q(G)$ for all graphs G. This has bearing on the decidability of the quantum chromatic number. Determining whether $\chi_q^{(r)}(G) \leq c$ is decidable[1] because it boils down to solving the set of quadratic equations (3.1) over the reals in a space of dimension cr, for which there exist exact algorithms based on extensions of the Gröbner basis technique [18]. However, $\chi_q(G) = \inf_r \chi_q^{(r)}(G)$ is not decidable in such an easy way. It should be possible to prove at least an upper bound on r that is sufficient to attain the limit. In that case, it would make sense to ask about the complexity of computing $\chi_q(G)$, in particular whether it is NP-hard, as is computing the chromatic number $\chi(G)$.

3.3 General properties

We look at some basic properties of the quantum chromatic number as a graph parameter. None of these are particularly surprising; indeed, the point of this section is to show that the quantum chromatic number "does the right thing", and merits being considered as a generalisation of the (classical) chromatic number.

A *homomorphism* is a mapping from one graph to another that preserves edges. That is, a homomorphism ϕ from G to H maps vertices of G to vertices of H such that if x and y are adjacent in G then $\phi(x)$ and $\phi(y)$ are adjacent in H. We write $G \to H$ to indicate that there exists a homomorphism from G to H.

The following easy observation is a useful tool.

Lemma 3.3.1. *If $G \to H$, then $\chi_q^{(r)}(G) \leq \chi_q^{(r)}(H)$ for all r and hence $\chi_q(G) \leq \chi_q(H)$.*

Proof. Let ϕ be a homomorphism from G to H. Then any quantum colouring of H gives a quantum colouring of G by colouring the vertex x of G with the colour assigned to $\phi(x)$ in H. □

It is trivial to see that if (and only if) G has no edges then $\chi_q^{(r)}(G) = \chi_q(G) = 1$. With a little more effort, one sees that if $G = K_n$ then $\chi_q^{(r)}(G) = \chi_q(G) = n$, where K_n is the complete graph on n vertices. For, using Theorem 3.2.1 and eqn (3.10), we have a set of n rank-r pairwise orthogonal operators in a space of dimension cr. We can say a little more.

[1] Not necessarily efficiently; it is also open whether an efficient algorithm could be found (perhaps using semidefinite programming) to determine $\chi_q^{(r)}(G)$.

Theorem 3.3.2. *$\chi_q(G) = 2$ if and only if $\chi(G) = 2$.*

Proof. If $\chi(G) = 2$, then $G \to K_2$ and $K_2 \to G$, and so by Lemma 3.3.1 $\chi_q(G)$ is at most and at least 2. On the other hand, consider any quantum colouring of G with 2 colours, with orthogonal projectors $E_{v\alpha}$ for Alice, $\alpha = 0, 1$. By eq. (3.10), however, $E_{v\alpha} = I - E_{w\alpha}$ for adjacent vertices v and w. That means, looking at a fixed colour α^*, we encounter only two different operators as we traverse the graph – these can serve as colours in a colouring as adjacent vertices will have different $E_{v\alpha^*}$. □

The *clique number* of G, denoted by $\omega(G)$, is the size of the largest complete subgraph of G.

Theorem 3.3.3. $\omega(G) \leq \chi_q(G) \leq \chi(G)$

Proof. Any graph G contains $K_{\omega(G)}$ as a subgraph, so $K_{\omega(G)} \to G$. Also $G \to K_{\chi(G)}$, by mapping each vertex to the vertex of $K_{\chi(G)}$ corresponding to its colour. The result follows by Lemma 3.3.1. □

(Of course, Theorems 3.3.2 and 3.3.3 remain valid if we replace χ_q with $\chi_q^{(r)}$ for any r.) Let G and H be two graphs on the same vertex set. We define the graph $G \cup H$ to be the graph whose edge set is the union of the edge sets of G and H. It is easy to see that $\chi(G \cup H) \leq \chi(G)\chi(H)$: colour each vertex in $G \cup H$ with the ordered pair of colours it received in colourings of G and H, respectively. This idea can be extended to quantum colourings:

Theorem 3.3.4. *For any r, s, we have $\chi_q^{(rs)}(G \cup H) \leq \chi_q^{(r)}(G)\chi_q^{(s)}(H)$.*

Proof. Given rank-r and rank-s quantum colourings for G and H respectively, we obtain a rank-rs quantum colouring of $G \cup H$ by taking the tensor products of the individual POVM operators associated to the vertices. □

As a corollary, we obtain the following, showing that a graph and its complement cannot both have small quantum chromatic number.

Theorem 3.3.5. $\chi_q(G)\chi_q(\overline{G}) \geq n$.

Proof. Apply Theorem 3.3.4 with $H = \overline{G}$, the complement of G. □

3.4 Orthogonal representations

The origin of the quantum chromatic number is in Hadamard graphs [31, 27], which are a special case of orthogonality graphs, so it is natural to consider the larger family.

An *orthogonal representation* of a graph G is a mapping ϕ from the vertices of G to the non-zero vectors of some vector space, such that if two vertices x and y are

adjacent, then $\phi(x)$ and $\phi(y)$ are orthogonal. Conversely, given a set of vectors, we define their *orthogonality graph* to be the graph having the vectors as vertices, with two vectors adjacent if and only if they are orthogonal.

Let $\xi(G)$ be the smallest integer c such that G has an orthogonal representation in the vector space $\mathbb{C}^c$. Furthermore, let $\xi'(G)$ be the smallest integer c such that G has an orthogonal representation in the vector space $\mathbb{C}^c$ with the added restriction that the entries of each vector must have modulus one. (In fact, we really only need the entries in any particular vector to have constant modulus.)

Theorem 3.4.1. $\omega(G) \leq \xi(G) \leq \chi_q^{(1)}(G) \leq \xi'(G) \leq \chi(G)$

Proof. For each integer c, let F_c be the quantum Fourier transform of order c, i.e., $[F_c]_{j,k} = \frac{1}{\sqrt{c}} e^{2\pi ijk/c}$. Then:

- Given a graph with $\chi(G) = c$, colour the vertices with the rows of F. Adjacent vertices have distinct colours and hence orthogonal vectors, and thus $\xi'(G) \leq \chi(G)$.
- Given a graph with $\chi_q^{(1)}(G) = c$, map each vertex to the first column of its corresponding unitary matrix. By eqn. (3.8) adjacent vertices will get mapped to orthogonal vectors, and thus $\xi(G) \leq \chi_q^{(1)}(G)$.
- Given a graph with $\omega(G) = c$, any orthogonal representation of it must contain c pairwise orthogonal vectors and thus $\omega(G) \leq \xi(G)$.
- Finally, given a graph with $\xi'(G) = c$, map each vertex x to $\Delta_x F_c$, where Δ_x is the diagonal (unitary) matrix whose diagonal entries are the entries of x. Then $\langle x|y\rangle = 0$ implies that $(\Delta_v F_c)^\dagger (\Delta_w F_c)$ has only zeroes on the diagonal. Thus $\chi_q^{(1)}(G) \leq \xi'(G)$.

□

Finally, we derive an upper bound on the chromatic number of the orthogonality graph on $\mathbb{C}^k$ in terms of k, which gives an upper bound on $\chi(G)$ in terms of $\xi(G)$ for any graph G by converting any orthogonal representation of G into a colouring of G. This allows us to bound the largest possible gap between $\chi(G)$ and $\chi_q^{(1)}(G)$.

Theorem 3.4.2. *For a graph G,*

$$\chi(G) \leq (1 + 2\sqrt{2})^{2\,\xi(G)} \leq (1 + 2\sqrt{2})^{2\chi_q^{(1)}(G)}.$$

Proof. To show the first inequality, we give a colouring of the orthogonality graph on $\mathbb{C}^k$, where $k = \xi(G)$. This can be produced from a set of unit vectors $V = \{|v_i\rangle\}$ such that for all unit vectors $|w\rangle \in \mathbb{C}^k$, $\||w\rangle - |v_i\rangle\|_2 < 1/\sqrt{2}$ for some i, by assigning colour i to $|w\rangle$

(if there are two or more vectors in V satisfying this inequality, picking one arbitrarily). This works because, for any two vectors $|x\rangle$, $|y\rangle$, $\langle x|y\rangle = 0 \Rightarrow 2(1 - \mathrm{Re}(\langle x|y\rangle)) = \||x\rangle - |y\rangle\|_2^2 = 2$, so no two orthogonal vectors will receive the same colour. We use the argument of [69] to bound the size of such a set (which [69] calls a $1/\sqrt{2}$-net). Let $M = \{|v_i\rangle\}$ be a maximal set of unit vectors such that $\||v_i\rangle - |v_j\rangle\|_2 \geq 1/\sqrt{2}$ for all i and j and set $m = |M|$. Then M is a $1/\sqrt{2}$-net giving a m-colouring of the orthogonality graph of $\mathbb{C}^k$. Observe that, as subsets of $\mathbb{R}^{2k}$, the open balls of radius $1/(2\sqrt{2})$ about each $|v_i\rangle$ are disjoint and contained in the overall ball of radius $1 + 1/(2\sqrt{2})$. Thus $m(1/(2\sqrt{2}))^{2k} \leq (1 + 1/(2\sqrt{2}))^{2k}$.

The second inequality follows from Theorem 3.4.1. □

The above result shows that the separation between $\chi(G)$ and $\chi_q^{(1)}(G)$ can be at most exponential; the results of [31, 128], on the other hand, demonstrate that exponential gaps can occur, showing that this inequality is tight up to constant factors.

3.5 Few colours

Here we investigate properties of graphs with small quantum chromatic number or small orthogonal rank. We already saw that for two colours, classical and quantum chromatic numbers coincide. It turns out that for three this is also the case, and for numbers up to 8 the quantum chromatic number stays close to the orthogonal rank.

Theorem 3.5.1. *Given a graph G, $\chi_q^{(1)}(G) = 3$ if and only if $\chi(G) = 3$.*

Proof. If $\chi(G) = 3$, we cannot have $\chi_q(G) = 2$ (nor 1 because the graph is not empty) as this would mean $\chi(G) = 2$. On the other hand, consider a rank-1 quantum colouring with 3 colours. We use the analysis in Section 3.2 and in particular the observation that we can view the quantum colouring as a family of 3×3 unitaries U_v such that eqn (3.9) holds. The columns of the unitaries are just the basis vectors $|e_{v0}\rangle$, $|e_{v1}\rangle$, $|e_{v2}\rangle$. W.l.o.g. the graph is connected and for one distinguished vertex v_0 we may assume $U_{v_0} = I$.

The crucial observation is that there are essentially only two unitary matrices $U_v^\dagger U_w$ with zeroes on the diagonal [119]: they can only be

$$\begin{pmatrix} 0 & 0 & * \\ * & 0 & 0 \\ 0 & * & 0 \end{pmatrix} \quad \text{or} \quad \begin{pmatrix} 0 & * & 0 \\ 0 & 0 & * \\ * & 0 & 0 \end{pmatrix},$$

where the starred entries must be roots of unity. Starting from v_0 we hence find inductively that all U_v are, up to phase factors, permutation matrices. Just looking at the first column, we now obtain a 3-colouring of G, choosing the colour according to the row in which the nonzero entry of the column vector is. □

We now show that, in small dimension, having a small-dimensional orthogonal representation is sufficient for a graph to have a low quantum chromatic number.

Theorem 3.5.2. *Let G be a graph with an orthogonal representation in $\mathbb{R}^c$. If $c = 3, 4$ then $\chi_q^{(1)}(G) \leq 4$; if $4 < c \leq 8$ then $\chi_q^{(1)}(G) \leq 8$.*

Proof. If $c = 4, 8$ then associate every vector $v \in \mathbb{R}^4$ and $w \in \mathbb{R}^8$ to real orthogonal designs V and W of the form $OD\,(4; 1, \ldots, 1)$ and $OD\,(8; 1, \ldots, 1)$, respectively [44]. For example, every vector $v \in \mathbb{R}^4$ is associated to a real orthogonal matrix

$$V = \begin{pmatrix} v_1 & v_2 & v_3 & v_4 \\ -v_2 & v_1 & -v_4 & v_3 \\ -v_3 & v_4 & v_1 & -v_2 \\ -v_4 & -v_3 & v_2 & v_1 \end{pmatrix}.$$

If $v \in \mathbb{R}^c$ and $c = 3$ or $4 < c \leq 8$ then concatenate a zero-vector of length 1 or $8 - c$ to v, respectively, and proceed as above. □

The above construction works based on the fact that in dimensions 4 and 8 there exist division algebras (Hamilton quaternions and Cayley octonions). That is, if we associate a quaternion $v_1 + v_2 i + v_3 j + v_4 k$ with a vector (v_1, v_2, v_3, v_4), multiplying that quaternion by i, j, or k will give an orthogonal vector (and similarly for octonions). Unfortunately division algebras exist only in dimensions 1, 2, 4 and 8 [50].

3.6 A graph with a small quantum chromatic number

We now give an example of a fairly small graph G (18 vertices and 44 edges) which has quantum chromatic number – in fact, even $\chi_q^{(1)}(G)$ – equal to 4, but chromatic number 5. Label the vertices with integers $1 \ldots 18$; then

$$\begin{aligned} E = \{&(1,2),(1,3),(1,11),(1,12),(1,16),(2,3),(2,4),\\ &(2,13),(3,4),(3,13),(4,5),(4,6),(4,10),(4,17),\\ &(5,6),(5,7),(5,14),(6,7),(6,14),(7,8),(7,9),\\ &(7,16),(8,9),(8,10),(8,13),(9,10),(9,13),(10,11),\\ &(10,12),(10,17),(11,12),(11,14),(12,14),(13,14),\\ &(13,15),(13,18),(14,15),(14,18),(15,16),(15,17),\\ &(15,18),(16,17),(16,18),(17,18)\} \end{aligned}$$

The graph may be visualised as consisting of two components connected to each other by 8 additional edges: a 4-regular graph on vertices $1 - 14$ [augmented by two edges $(4, 10)$ and $(13, 14)$], and a 4-clique on vertices $15 - 18$, see Fig. 3.1. The following list

of vectors gives an orthogonal representation of G in $\mathbb{R}^4$, which by Proposition 3.5.2 gives a quantum colouring with 4 colours:

$$\{(0,0,1,-1),(1,0,0,0),(0,1,1,1),(0,1,0,-1),(0,0,1,0),$$
$$(1,1,0,1),(1,-1,0,0),(0,0,0,1),(1,1,1,0),(1,0,-1,0),$$
$$(0,1,0,0),(1,0,1,1),(0,1,-1,0),(1,0,0,-1),(1,1,1,1),$$
$$(1,1,-1,-1),(1,-1,1,-1),(1,-1,-1,1)\}$$

Because G contains a 4-clique, $\chi_q(G)$ cannot, on the other hand, be smaller than 4.

It may be verified as follows that G cannot be 4-coloured. Assume w.l.o.g. that vertices 15-18 are coloured $1, 2, 3, 4$ respectively. Then vertices 13 and 14 must divide colours 2 and 3 between them; and for a valid 4-colouring, none of the triplets $(1, 4, 13)$, $(1, 10, 14)$, $(4, 7, 14)$, $(7, 10, 13)$ may consist of 3 distinct colours. Using this, it is straightforward to try all the possible colourings of vertex 7 and see that each leads to vertices 4 and 10 being assigned the same colour.

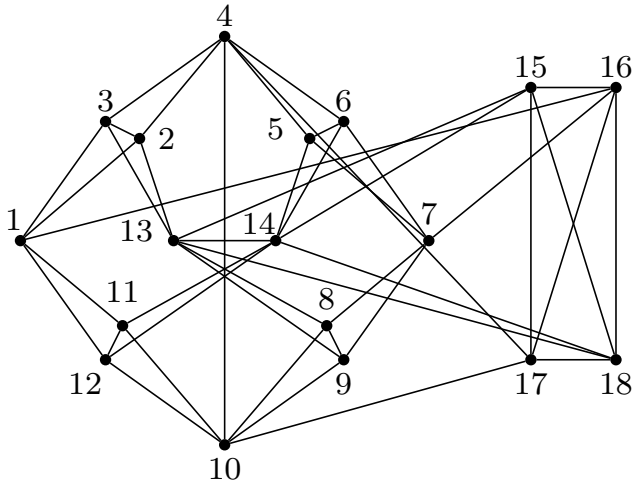


Figure 3.1: A graph G with $\chi_q^{(1)}(G) = 4$, but $\chi(G) = 5$.

This graph is much smaller and uses fewer colours than the previously smallest specimen exhibiting a separation between classical and quantum chromatic numbers: in [14] a graph on 1609 vertices is described with $\chi(G) \geq 13$ and $\chi_q(G) = 12$.

As Theorem 3.5.1 shows that if $\chi_q^{(1)}(G) = 3$ then $\chi(G) = 3$, this is the minimum value of $\chi_q^{(1)}(G)$ for which we can achieve a separation from $\chi(G)$. However, a graph showing a separation with a smaller number of vertices might exist, as might a graph with $\chi_q(G) = 3$, $\chi(G) > 3$. Also, in the general setting of pseudo-telepathy games, an even smaller game is known where Alice and Bob gain a quantum advantage from a 3×3-dimensional entangled state rather than the 4×4-dimensional state used here [43].

Chapter 4

A lower bound on entanglement-assisted quantum communication complexity

4.1 Introduction

From the model of the previous chapter, where sharing entanglement allowed two players to win a game with no communication at all, we turn to a more modest goal: using quantum mechanics to *reduce* the amount of communication required for Alice and Bob to complete some task. The study of this problem belongs to the field of quantum communication complexity. Classical communication complexity was first introduced by Yao [131] in 1979, and has been found to have many important links to other areas of computer science. The same applies to its quantum generalisation. (See [129] and [91] for excellent introductions to quantum and classical communication complexity, respectively.)

Specifically, consider a total Boolean function $f : \{0,1\}^n \times \{0,1\}^n \mapsto \{0,1\}$. The quantum communication complexity of f is defined to be the minimum number of qubits required to be transmitted between two parties (Alice and Bob) for them to compute $f(x,y)$ for any two n-bit inputs x, y, given that Alice starts out with x and Bob with y. This number is clearly upper-bounded by n, but for some functions may be considerably lower. Alice and Bob may be allowed some probability of error ϵ, and may be allowed to share an entangled state before they start their protocol. We will assume that Bob has to output the result.

Some functions are known to have a quantum communication complexity lower than their classical communication complexity (for example, a bounded-error protocol for the disjointness function $f(x,y) = 1 \Leftrightarrow |x \wedge y| = 0$ requires $\Omega(n)$ bits of classical communication, but only $\Theta(\sqrt{n})$ qubits of quantum communication [31, 1, 114]), but

the extent of the possible reduction in communication is unknown. In particular, it is still open whether the quantum communication complexity of total functions can ever be exponentially smaller than the classical communication complexity, although it has been shown [113, 62] that quantum communication can exponentially reduce the cost of computing a partial function (where there is a promise on the input). It is therefore of interest to produce lower bounds on quantum communication complexity.

In this context, the model with prior entanglement is less well understood; although there are strong bounds known for some classes of functions [42, 114], there are few general lower bounds [32]. In the 1-way and simultaneous message passing[1] models of communication complexity, sharing entanglement can reduce the communication cost of a partial function exponentially [62, 60], but it is unknown whether the same might hold for a total function.

In this chapter, we develop an elegant result of Cleve et al. that relates computation to communication. Cleve et al. showed [42] that, if Alice and Bob have access to a protocol to exactly compute the inner product function $IP(x,y) = \sum_i x_i y_i \pmod 2$, then this can be used to produce a quantum protocol that communicates Alice's input x to Bob. They used this to show that IP cannot be computed (exactly and without prior entanglement) by sending fewer than n qubits from Alice to Bob. Similar results hold for the bounded-error case and with prior entanglement.

We show that a weaker form of this result can be extended to *all* Boolean functions. That is, for almost any Boolean function f, the ability for Alice and Bob to compute f implies the ability for Alice to send some arbitrary information to Bob. The extension leads to the development of a new complexity measure for Boolean functions: *communication capacity*. Given a Boolean function $f(x,y)$, we define the communication capacity of f as the maximum number of bits which the execution of a protocol to compute f allows Alice to communicate to Bob (in an asymptotic sense). This is a concept which has no classical analogue and which, as we will show, gives a lower bound on the quantum communication complexity of f, with or without entanglement.

Some comments on notation: we will use M to denote the square communication matrix of f (where M_{xy} is equal to $(-1)^{f(x,y)}$). We will use the standard notation $Q_E(f)$ to denote the quantum communication complexity of f in the case where the protocol must be exact, $Q_\epsilon(f)$ the complexity where Alice and Bob are allowed to err with probability $\epsilon < 1/2$, and $Q_2(f)$ the complexity in the case where $\epsilon = 1/3$. In all three cases, Alice and Bob's initial state is separable; $Q^*_E(f)$, $Q^*_\epsilon(f)$ and $Q^*_2(f)$ will represent the equivalent quantities in the case where they are allowed to share an arbitrary initial entangled state.

Then the main result of this chapter can be stated as follows (a more precise statement is given as Theorem 4.2.1 below).

[1] This is a model where Alice and Bob communicate not with each other but with a referee.

Theorem 4.1.1. *Let $f : \{0,1\}^n \times \{0,1\}^n \mapsto \{0,1\}$ be a total Boolean function with communication matrix M. Then, for any non-negative diagonal matrices A and B with $\|A\|_2 = \|B\|_2 = 1$,*

$$Q_2^*(f) = \Omega(H(\sigma^2(AMB))/\log n) \tag{4.1}$$

where $\sigma^2(M)$ is the vector of squared singular values of a matrix M and $H(v)$ is the Shannon entropy of v, i.e. $H(v) = -\sum_i v_i \log_2 v_i$.

We use this result to show that the quantum communication complexity of a random function is linear in n, even if Alice and Bob are allowed to share an arbitrary entangled state.

This chapter is joint work with Andreas Winter and has been published previously as "A lower bound on entanglement-assisted quantum communication complexity", in the proceedings of ICALP 2007, pp. 122–133 (`quant-ph/0610085`).

4.1.1 Related work

This work is part of a large and growing body of research on quantum communication complexity. The concept of quantum communication complexity was introduced by Yao [132] more than a decade ago, and then studied extensively in Kremer's thesis [89]. The study of communication complexity with prior entanglement (in a three-party scenario using only classical communication) was initiated by Cleve and Buhrman [41].

The lower bound of Theorem 4.1.1 is a generalisation of a bound obtained by Klauck [87] on quantum communication complexity in the model without entanglement. The result here can thus be seen as extending Klauck's bound to the model of entanglement-assisted quantum communication, and giving it a satisfying operational interpretation. As our bound also holds for classical communication complexity, it fits into the framework of results using ideas from quantum information to say something about classical computation.

Entropic lower bounds for quantum communication complexity have been studied previously by van Dam and Hayden [46]. Their work starts from a different perspective to this chapter: considering the communication required to perform a quite general state transformation task. They produce a lower bound on communication complexity based on Rényi entropy (q.v.) which is similar to the bound given here, although they only consider the uniform distribution on Alice and Bob's inputs. Their bound has the advantage that it removes the $\log n$ factor from Theorem 4.1.1; however, an unfortunate side-effect is that it only holds in a scenario where Alice and Bob are allowed to share only maximally entangled states (rather than arbitrary entanglement).

Following the completion of this work, Linial and Shraibman have shown [97] that the minimum γ_2 norm of matrices that approximate the communication matrix M gives a lower bound on entanglement-assisted quantum communication complexity.

This norm is defined as

$$\gamma_2(M) = \min_{XY=M} \|X\|_{\ell_2 \to \ell_\infty} \|Y\|_{\ell_1 \to \ell_2} \tag{4.2}$$

where $\|X\|_{\ell_2 \to \ell_\infty}$ is the largest ℓ_2 norm of a row of X, and $\|Y\|_{\ell_1 \to \ell_2}$ is the largest ℓ_2 norm of a column of Y. Among other results, Linial and Shraibman use this lower bound to extend the bound of Klauck [87] to the model of quantum communication with entanglement. Their work thus proves the special case of Theorem 4.1.1 where $A_{ii} = B_{ii} = 1/\sqrt{2^n}$.

As is usual in computational complexity, we would expect most functions to have "high" quantum communication complexity. Kremer showed [89] by a counting argument that a random function f has $Q_2(f) \geq n/2$ (and thus $Q_E(f) \geq n/2$). Buhrman and de Wolf extended Kremer's methods to show that, for all f, it holds that $Q_E^*(f) \geq (\log \operatorname{rank}(M))/2$ [32] (an equivalent result is shown in Section 6.4.2 of [107]). As almost all Boolean matrices have full rank, this shows that for almost all f, $Q_E^*(f) \geq n/2$. Very recently, Gavinsky, Kempe and de Wolf [61] have shown the final remaining case: for almost all f, $Q_2^*(f) = \Omega(n)$. Their technique was to relate quantum communication protocols to quantum fingerprinting protocols, and then to show a relationship between quantum fingerprinting and some well-studied concepts from classical computational learning theory. This result was shown independently by Linial and Shraibman [96]; their paper also extends the well-known discrepancy lower bound to the model of quantum communication with entanglement.

As an application of our communication capacity technique, we reprove the result that for almost all f, $Q_2^*(f) = \Omega(n)$. The proof is of a quite different character and of (arguably) a more "quantum" nature, as it is based on showing that the entropy of almost all density matrices produced in a certain random way is high.

4.2 Turning any distributed function into a communication protocol

In this section, we will describe a protocol (which is a simple extension of the protocol in [42] for IP) that allows any protocol for evaluating a distributed function to be turned into a communication protocol. However, for some functions, the communication will be considerably more inefficient than IP allows (Alice may only be able to send $\ll n$ bits to Bob).

4.2.1 Exact protocols

Say Alice and Bob have access to a classical or quantum protocol that computes $f(x, y)$ exactly. We express this as a unitary P that performs the following action.

$$P|x\rangle_A|y\rangle_B|0\rangle_B|a\rangle_{AB} = |x\rangle_A|y\rangle_B|f(x,y)\rangle_B|a'\rangle_{AB} \tag{4.3}$$

where $|a\rangle$, $|a'\rangle$ are arbitrary (and possibly entangled) ancilla states shared by Alice and Bob. Note that, as P does not modify the first two registers, we may decompose it as follows:

$$P = \sum_{x,y} |x\rangle\langle x|_A \otimes |y\rangle\langle y|_B \otimes U_{xy} \tag{4.4}$$

for some unitary U_{xy} acting only on the last two registers. Following [42], we will turn this into a "clean" protocol P' by giving Bob an additional qubit to copy the answer into, then running the protocol backwards to uncompute the "junk" $|a'\rangle$. The steps of the clean protocol are thus

$$\begin{aligned}
&\text{(i)} && |x\rangle_A|y\rangle_B|0\rangle_B|0\rangle_B|a\rangle_{AB} \\
&\text{(ii)} && \to\ |x\rangle_A|y\rangle_B|f(x,y)\rangle_B|0\rangle_B|a'\rangle_{AB} \\
&\text{(iii)} && \to\ |x\rangle_A|y\rangle_B|f(x,y)\rangle_B|f(x,y)\rangle_B|a'\rangle_{AB} \\
&\text{(iv)} && \to\ |x\rangle_A|y\rangle_B|0\rangle_B|f(x,y)\rangle_B|a\rangle_{AB}
\end{aligned}$$

where now the fourth register contains the answer. Ignoring the third and fifth registers, which are the same at the beginning and the end of the protocol, we are left with the map

$$P'|x\rangle_A|y\rangle_B|0\rangle_B = |x\rangle_A|y\rangle_B|f(x,y)\rangle_B \tag{4.5}$$

Note that, if the original protocol P communicated a qubits from Alice to Bob and b qubits from Bob to Alice, the protocol P' requires $a + b$ qubits to be communicated in each direction. That is, P' sends as many qubits in the "forward" direction as the original protocol P sends in total. Now say Alice wants to communicate her input x to Bob using this protocol. They start with the following state, where (b_y) is an arbitrary probability distribution on Bob's inputs:

$$|\psi\rangle = |x\rangle_A \left(\sum_{y\in\{0,1\}^n} \sqrt{b_y}|y\rangle_B \right) \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)_B \tag{4.6}$$

Note that this state is separable (so we do not *require* entanglement to execute the communication protocol). After executing the clean protocol for f, they are left with

$$P'|\psi\rangle = |x\rangle_A \left(\frac{1}{\sqrt{2}} \sum_{y\in\{0,1\}^n} \sqrt{b_y}|y\rangle_B \left(|f(x,y)\rangle - |1-f(x,y)\rangle\right)_B \right) \tag{4.7}$$

$$= |x\rangle_A \left(\sum_{y\in\{0,1\}^n} (-1)^{f(x,y)} \sqrt{b_y}|y\rangle_B \right) \frac{1}{\sqrt{2}} \left(|0\rangle - |1\rangle\right)_B \tag{4.8}$$

Ignoring the registers that remain the same throughout, Bob has the following state at the end of the protocol.

$$|\psi_x\rangle = \sum_{y\in\{0,1\}^n} (-1)^{f(x,y)} \sqrt{b_y}|y\rangle \tag{4.9}$$

This state provides some information about Alice's bit string x. If $\langle\psi_x|\psi_{x'}\rangle = 0$ for all $x' \neq x$ (as is the case with the protocol of [42] for IP, where Bob uses the uniform distribution on his inputs) then Bob can determine x with certainty and hence has received n bits from Alice. If this is not the case, then we can still quantify precisely how much information can be transmitted. The protocol is equivalent to Alice encoding the classical bit-string x as a state $|\psi_x\rangle$, and co-operating with Bob to send it to him. Say Alice uses a distribution (a_x) on her inputs. Then the ensemble representing what Bob eventually receives is

$$\rho = \sum_{x\in\{0,1\}^n} a_x|\psi_x\rangle\langle\psi_x| \tag{4.10}$$

By Holevo's theorem [74], the entropy $S(\rho)$ describes the maximum number of bits of classical information about x available to Bob by measuring ρ. And, by the Holevo-Schumacher-Westmoreland channel coding theorem for a channel with pure signal states [67], Alice and Bob can achieve this bound (in an asymptotic sense) using block coding!

Therefore, the ability to compute f exactly can be used to transmit $S(\rho)$ bits of information through a quantum channel, even though this does not hold if Alice and Bob are restricted to a classical channel. We thus define the *communication capacity* of a Boolean function f as the maximum over all probability distributions (a_x) (on Alice's inputs) and (b_y) (on Bob's inputs) of

$$S\left(\sum_{x\in\{0,1\}^n} a_x|\psi_x\rangle\langle\psi_x| \right) \text{ , where } |\psi_x\rangle = \sum_{y\in\{0,1\}^n} (-1)^{f(x,y)} \sqrt{b_y}|y\rangle \tag{4.11}$$

4.2.2 Bounded error protocols

In the case where Alice and Bob have access to a protocol computing f with some probability of error, Bob will not have the state $|\psi_x\rangle$ at the end of the protocol, but

rather some approximation $|\psi_x^\epsilon\rangle$. We will now show that, if the error probability is small, this is in fact still sufficient to communicate a significant amount of information from Alice to Bob. As before, Alice will use a distribution (a_x) on her inputs, and Bob a distribution (b_y).

Say Alice and Bob are using a protocol P^ϵ that computes f with probability of error ϵ, where $\epsilon < 1/2$. As before, the $|x\rangle$ and $|y\rangle$ registers will be unchanged by this protocol, so we can write

$$P^\epsilon = \sum_{x,y} |x\rangle\langle x|_A \otimes |y\rangle\langle y|_B \otimes U_{xy}^\epsilon \tag{4.12}$$

Now let us run the protocol on the same starting state $|\psi\rangle$ as in the previous section.

$$\begin{aligned}
&\text{(i)} \qquad |x\rangle_A \left(\frac{1}{\sqrt{2}} \sum_{y\in\{0,1\}^n} \sqrt{b_y}|y\rangle_B |0\rangle_B (|0\rangle - |1\rangle)_B \right) |a\rangle_{AB} \\
&\text{(ii)} \quad \rightarrow \quad |x\rangle_A \left(\frac{1}{\sqrt{2}} \sum_{y\in\{0,1\}^n} \sqrt{b_y}|y\rangle_B (\alpha_{xy}|0\rangle + \beta_{xy}|1\rangle)_B (|0\rangle - |1\rangle)_B \right) |a'\rangle_{AB}
\end{aligned}$$

where the effect of U_{xy}^ϵ on the "answer" qubit has been decomposed into α_{xy} and β_{xy} components. If $f(x,y) = 0$, then $|\alpha_{xy}|^2 \geq 1 - \epsilon$, and thus (by unitarity) $|\beta_{xy}|^2 \leq \epsilon$; if $f(x,y) = 1$, $|\beta_{xy}|^2 \geq 1 - \epsilon$ and $|\alpha_{xy}|^2 \leq \epsilon$. The ancilla register is still completely arbitrary, and in particular may be entangled with any of the other registers. Continuing the protocol, we have

$$\begin{aligned}
&\text{(iii)} \rightarrow |x\rangle_A \left(\frac{1}{\sqrt{2}} \sum_{y\in\{0,1\}^n} \sqrt{b_y}|y\rangle_B (\alpha_{xy}|0\rangle(|0\rangle - |1\rangle) - \beta_{xy}|1\rangle(|0\rangle - |1\rangle))_B \right) |a'\rangle_{AB} \\
&\text{(iv)} \rightarrow |x\rangle_A \Bigg(\frac{1}{\sqrt{2}} \sum_{y\in\{0,1\}^n} \sqrt{b_y}|y\rangle_B (\alpha_{xy}(\alpha_{xy}^*|0\rangle + \gamma_{xy}^*|1\rangle)|0\rangle - \alpha_{xy}(\alpha_{xy}^*|0\rangle + \\
&+\gamma_{xy}^*|1\rangle)|1\rangle - \beta_{xy}(\beta_{xy}^*|0\rangle + \delta_{xy}^*|1\rangle)|0\rangle + \beta_{xy}(\beta_{xy}^*|0\rangle + \delta_{xy}^*|1\rangle)|1\rangle)_B \Bigg) |a\rangle_{AB}
\end{aligned}$$

where we introduce γ_{xy}^* and δ_{xy}^* as arbitrary elements of $(U_{xy}^\epsilon)^\dagger$, subject only to the constraint that U_{xy}^ϵ be unitary. We may now remove registers that end the protocol unchanged and rewrite Bob's final state as

$$|\psi_x^\epsilon\rangle = \sum_{y\in\{0,1\}^n} \sqrt{b_y}|y\rangle \left((|\alpha_{xy}|^2 - |\beta_{xy}|^2)|0\rangle + (\alpha_{xy}\gamma_{xy}^* - \beta_{xy}\delta_{xy}^*)|1\rangle \right) \tag{4.13}$$

Now, if $f(x,y) = 0$, then $|\alpha_{xy}|^2 - |\beta_{xy}|^2 \geq 1 - 2\epsilon > 0$, whereas if $f(x,y) = 1$, $|\alpha_{xy}|^2 -$

$|\beta_{xy}|^2 \leq 2\epsilon - 1 < 0$. We may therefore write

$$|\psi_x^\epsilon\rangle = \sum_{y\in\{0,1\}^n} \sqrt{b_y}|y\rangle \left((-1)^{f(x,y)} \cos\theta_{xy}|0\rangle + e^{i\phi_{xy}} \sin\theta_{xy}|1\rangle\right) \tag{4.14}$$

where θ_{xy} is real with $\cos\theta_{xy} \geq 1 - 2\epsilon$, and ϕ_{xy} is an arbitrary phase. Crucially, the form of these states is quite restricted and close to the original $|\psi_x\rangle$. In fact, it is clear that

$$|(\langle\psi_x|\langle 0|)|\psi_x^\epsilon\rangle|^2 \geq (1-2\epsilon)^2 \tag{4.15}$$

Set $\rho^\epsilon = \sum_{x\in\{0,1\}^n} a_x|\psi_x^\epsilon\rangle\langle\psi_x^\epsilon|$. We will compare this to the state

$$\rho' = \sum_{x\in\{0,1\}^n} a_x|\psi_x\rangle|0\rangle\langle\psi_x|\langle 0|$$

(where of course $S(\rho') = S(\rho)$). We have

$$\|\rho' - \rho^\epsilon\|_1 \leq 2\sqrt{1-(1-2\epsilon)^2} \leq 4\sqrt{\epsilon} \tag{4.16}$$

We will use Fannes' inequality [53] to show that $S(\rho^\epsilon) \approx S(\rho)$. Define the function

$$\eta_0(x) = \begin{cases} -x\log x & \text{for } x \leq 1/e \\ 1/e\log e & \text{for } x > 1/e \end{cases} \tag{4.17}$$

Then Fannes' inequality gives that

$$S(\rho^\epsilon) \geq S(\rho) - 4\sqrt{\epsilon}n - \log\eta_0(4\sqrt{\epsilon}) \tag{4.18}$$

4.2.3 Communication complexity lower bounds from communication capacity

A lower bound for the communication capacity of a function f can be written down in terms of its communication matrix M as follows. As before, set

$$\rho = \sum_{x\in\{0,1\}^n} a_x|\psi_x\rangle\langle\psi_x| \text{ for } |\psi_x\rangle = \sum_{y\in\{0,1\}^n} (-1)^{f(x,y)}\sqrt{b_y}|y\rangle \tag{4.19}$$

for arbitrary probability distributions (a_x), (b_y) on Alice and Bob's inputs. Define the rescaled Gram matrix G as $G_{ij} = \sqrt{a_i}\sqrt{a_j}\langle\psi_i|\psi_j\rangle$. Now it is known [83] that G will have the same eigenvalues as ρ, and thus the same entropy. But it can easily be verified that

$$G = (AMB)(AMB)^\dagger \tag{4.20}$$

where A and B are diagonal matrices with $A_{ii} = \sqrt{a_i}$, $B_{ii} = \sqrt{b_i}$. So the eigenvalues of G are simply the singular values squared of AMB. We may thus write

$$S(\rho) = H(\sigma^2(AMB)) \tag{4.21}$$

where $\sigma^2(M)$ denotes the vector containing the squared singular values of a matrix M. We can now produce lower bounds on the quantum communication complexity of f by appealing to the result of Nayak and Salzman [105] which states that, if Alice wishes to transmit n bits to Bob over a quantum channel with probability of success p, Alice must send $m \geq \frac{1}{2}\left(n - \log\frac{1}{p}\right)$ qubits to Bob. If they are not allowed to share prior entanglement, the factor of $1/2$ vanishes. This immediately gives a lower bound on the exact quantum communication complexity of f, as lower bounds on the forward communication required for the "clean" protocols that we use translate into lower bounds on the total amount of communication needed for any communication protocol.

In the bounded-error case, we can still use the Nayak-Salzman result. Consider a block coding scheme with blocks of length k where each letter $|\psi_x^\epsilon\rangle$ is produced by one use of f, as in the previous section. By [67] there exists such a scheme that transmits $kS(\rho^\epsilon) - o(k)$ bits of information with k uses of f, as $k \to \infty$, and probability of success $p \to 1$. A lower bound on the bounded-error quantum communication complexity of f follows immediately:

$$mk \geq \frac{1}{2}(kS(\rho^\epsilon) - o(k) - o(1)), \tag{4.22}$$

hence, after taking the limit $k \to \infty$, $p \to 1$, we find $m \geq \frac{1}{2}S(\rho^\epsilon)$.

In order to reduce the error probability ϵ to $O(1/n^2)$ (to remove the additive term linear in n in inequality (4.18)), it is sufficient to repeat the original protocol $O(\log n)$ times and take a majority vote [89]. Alternatively, using (4.18) directly gives a better bound for functions for which $S(\rho)$ is linear in n. We thus have the following theorem.

Theorem 4.2.1. *Let $f : \{0,1\}^n \times \{0,1\}^n \mapsto \{0,1\}$ be a total Boolean function with communication matrix M. Then, for any non-negative diagonal matrices A and B with $\|A\|_2 = \|B\|_2 = 1$,*

$$Q_E(f) \geq H(\sigma^2(AMB)) \tag{4.23}$$

$$Q_E^*(f) \geq \frac{1}{2}H(\sigma^2(AMB)) \tag{4.24}$$

$$Q_\epsilon(f) \geq \begin{cases} \Omega(H(\sigma^2(AMB))/\log n) \\ H(\sigma^2(AMB)) - 4\sqrt{\epsilon}n - \log\eta_0(4\sqrt{\epsilon}) \end{cases} \tag{4.25}$$

$$Q_\epsilon^*(f) \geq \begin{cases} \Omega(H(\sigma^2(AMB))/\log n) \\ \frac{1}{2}(H(\sigma^2(AMB)) - 4\sqrt{\epsilon}n - \log\eta_0(4\sqrt{\epsilon})) \end{cases} \tag{4.26}$$

where $\eta_0(x)$ is defined as in equation (4.17).

If we use the uniform distribution on Alice and Bob's inputs, then $AMB = M/2^n$. In the case of the models without entanglement, Klauck obtained this specialised result via a different method [87]. This theorem can thus be seen as simultaneously extending Klauck's work to the model with entanglement, generalising it, and giving it an operational interpretation. The special case of the uniform distribution was also used by Cleve et al. [42] to prove their lower bound on the communication complexity of IP.

It has to be noted that this bound is not always tight: an example is provided by the disjointness problem, where Alice and Bob want to determine if their strings x and y have a position where they are both 1. It is known that the quantum communication complexity of this function is $\Theta(\sqrt{n})$ [114, 1]. On the other hand, an implicit upper bound on the entropy in Theorem 4.2.1 was already given for this case in [12], and it is only $O(\log n)$. Thus, not quite surprisingly, the ability of a function to let Alice communicate to Bob is not the same as the communication cost of implementing this computation.

4.3 Rényi entropic bounds on communication capacity

A disadvantage of the von Neumann entropy $S(\rho)$ is the difficulty involved in its computation. The *second Rényi entropy* $S_2(\rho)$ [116] provides an easily computable lower bound on $S(\rho)$. The Rényi entropy of order α is defined as

$$S_\alpha(\rho) = \frac{1}{1-\alpha} \log \operatorname{tr}(\rho^\alpha) \tag{4.27}$$

so we have

$$S_2(\rho) = -\log \operatorname{tr}(\rho^2) = -\log \sum_{i,j} |\rho_{ij}|^2 \tag{4.28}$$

and there is the fundamental property that $S_\alpha(\rho) \leq S_\beta(\rho)$ if $\alpha \geq \beta$. The Rényi entropies also obey the bounds $0 \leq S_\alpha(\rho) \leq n$. As with the von Neumann entropy, the Rényi entropy is a function only of the eigenvalues of ρ, so the Rényi entropy of the density matrix corresponding to an ensemble of equiprobable states is the same as that of the rescaled Gram matrix corresponding to these states. We can use this to write down a formula for the second Rényi entropy of a density matrix ρ corresponding to the communication matrix M of a function (as in the previous section, specialising to the uniform distribution on Alice and Bob's inputs), which gives a lower bound on its

communication capacity and thus its entanglement-assisted communication complexity.

$$\begin{aligned} S_2(\rho) &= -\log \operatorname{tr}\left(\frac{1}{2^{4n}}(MM^\dagger)^2\right) && (4.29)\\ &= 4n - \log\left(\sum_{i,j}\left(\sum_k M_{ik}M_{jk}\right)^2\right) && (4.30)\\ &= 4n - \log\left(\sum_{i,j,k,l} M_{ik}M_{jk}M_{il}M_{jl}\right) && (4.31)\end{aligned}$$

As discussed in Section 4.1.1, Rényi entropic arguments have previously been used by van Dam and Hayden [46] to put lower bounds on quantum communication complexity.

4.4 The quantum communication complexity of a random function

In this section, we will show a lower bound on the communication capacity – and thus the quantum communication complexity – of a random function (one which takes the value 0 or 1 on each possible input with equal probability). Define the state ρ as

$$\rho = \frac{1}{2^n}\sum_{k\in\{0,1\}^n} |\psi_k\rangle\langle\psi_k|, \text{ where } |\psi_k\rangle = \frac{1}{\sqrt{2^n}}\sum_{i\in\{0,1\}^n}(-1)^{a^k_{i+1}}|i\rangle \tag{4.32}$$

where a^k is a randomly generated 2^n-bit string, and a^k_i represents the i'th bit of a^k. We will show that the Rényi entropy $S_2(\rho)$ is high for almost all ρ.

Theorem 4.4.1. $\Pr\left[S_2(\rho) < (1-\delta)n\right] \leq e^{-(2^{\delta n}-1)^2/2}$.

Proof. We have

$$\begin{aligned} S_2(\rho) &= 4n - \log\left(\sum_{i,j}\left(\sum_k M_{ik}M_{jk}\right)^2\right) && (4.33)\\ &= 4n - \log\left(\sum_i\left(\sum_k (M_{ik})^2\right)^2 + \sum_{i\neq j}\left(\sum_k M_{ik}M_{jk}\right)^2\right) && (4.34)\\ &= 4n - \log\left(N^3 + T\right) && (4.35)\end{aligned}$$

where we define $N = 2^n$ and $T = \sum_{i\neq j}\left(\sum_k M_{ik}M_{jk}\right)^2$. It is then clear that

$$\Pr\left[S_2(\rho) < (1-\delta)n\right] = \Pr\left[T > N^3(N^\delta - 1)\right] \tag{4.36}$$

Each term in the inner sum in T (the sum over k) is independent and picked uniformly at random from $\{-1, 1\}$. We will now produce a tail bound for T using "Bernstein's

trick" (see Appendix A of [5]): from Markov's inequality we have

$$\Pr[T > a] < \mathbb{E}(e^{\lambda T})/e^{\lambda a} < \mathbb{E}(e^{\lambda X_{11}})^{N^2}/e^{\lambda a} \tag{4.37}$$

where we define $X_{ij} = (\sum_k M_{ik}M_{jk})^2$: each X_{ij} is independent and identically distributed, so T is the sum of $N(N-1) < N^2$ copies of X_{11}. It remains to calculate $\mathbb{E}(e^{\lambda X_{11}})$. This can be written out explicitly as follows.

$$\mathbb{E}(e^{\lambda X_{11}}) = \frac{1}{2^N}\sum_{k=0}^{N}\binom{N}{k}e^{\lambda(N-2k)^2} \tag{4.38}$$

It is then straightforward to see (using an inequality from [5]) that the following series of inequalities holds.

$$\begin{aligned}
\mathbb{E}(e^{\lambda X_{11}}) &\leq \frac{1}{2^N}\sum_{k=0}^{N}\binom{N}{k}\left(e^{\lambda(N-2k)^2} + e^{-\lambda(N-2k)^2}\right) && (4.39)\\
&\leq \frac{1}{2^{N-1}}\sum_{k=0}^{N}\binom{N}{k}e^{\lambda^2(N-2k)^4/2} && (4.40)\\
&\leq \frac{1}{2^{N-1}}\sum_{k=0}^{N}\binom{N}{k}e^{\lambda^2 N^4/2} = 2e^{\lambda^2 N^4/2} && (4.41)
\end{aligned}$$

Inserting this in eqn. (4.37), and minimising over λ, gives

$$\Pr[T > a] < 2e^{-a^2/2N^6} \tag{4.42}$$

and substituting $a = N^3(N^\delta - 1)$ gives the required result. □

In particular, putting $\delta = 1/2$ gives that $\Pr[S_2(\rho) < n/2] \leq 2e^{-(\sqrt{N}-1)^2/2}$, which is doubly exponentially small in n. As ρ corresponds to the communication matrix of a random function, Theorem 4.2.1 immediately gives the result that the entanglement-assisted quantum communication complexity of almost all functions is $\Omega(n)$.

Chapter 5

The distinguishability of random quantum states

5.1 Introduction

The fact that non-orthogonal pure quantum states may not be distinguished perfectly is a fundamental property of quantum mechanics. This leads to the following *quantum detection problem* (also known as *quantum state discrimination*): given an unknown quantum state $|\psi_?\rangle$, picked from a known set $\mathcal{E}$ with known a priori probabilities, find the "optimal" measurement $M^{opt}(\mathcal{E})$ to determine $|\psi_?\rangle$. First studied by Helstrøm [70] and Holevo [74] in the 1970s, there is now a vast literature related to this problem (see [36] for a survey). Several different criteria for optimality may been considered [70, 48, 51]; in this chapter we only concern ourselves with optimising the probability of success $P^{opt}(\mathcal{E})$, and in particular the related *state distinguishability problem* of finding $P^{opt}(\mathcal{E})$ without necessarily finding $M^{opt}(\mathcal{E})$. Efficient optimisation techniques can be used to estimate $P^{opt}(\mathcal{E})$ numerically [52]; however, the problem of finding an analytic expression for $P^{opt}(\mathcal{E})$ seems intractable. We are therefore led to attempting to produce bounds on this quantity.

In this chapter, two lower bounds on this optimal probability are derived; one based on the pairwise distinguishability of the states in $\mathcal{E}$, and one based on the eigenvalues of their Gram matrix. These bounds are derived for pure states but have an extension to mixed states. We also mention an upper bound on the probability of success based on pairwise distinguishability. As showing that a set of quantum states are quite distinguishable, or otherwise, forms an essential part of proofs in many areas of quantum information theory, we hope that these results will find application elsewhere.

The bounds are first applied to an illustrative example: a set of pure states with constant inner product. We then turn to the main subject of the chapter, which is the distinguishability of *random* quantum states. In particular, we derive a strong lower bound on the probability of success of distinguishing n random quantum states in d

dimensions, where n and d are large. In order to study the inner products of a set of random states, we use a powerful result from random matrix theory: the Marčenko-Pastur law [101]. This law turns out to give the distribution of the eigenvalues of the Gram matrix of n states in d dimensions, where n and d approach infinity and their ratio approaches a constant. In finite dimension, one can lower bound the rate of convergence to the law.

See Section 6.10 for an application of these results to the oracle identification problem in quantum computation.

The majority of this chapter has been published previously as "On the distinguishability of random quantum states", *Communications in Mathematical Physics* vol. 273 no. 3, pp. 619-636 (`quant-ph/0607011`).

5.2 Bounds on the distinguishability of quantum states

We consider an ensemble $\mathcal{E}$ containing n d-dimensional pure states $|\psi_i\rangle$ with their a priori probabilities p_i. We will use $\{|\psi'_i\rangle\}$ to denote the set containing the same states, renormalised to reflect their probabilities (i.e. $|\psi'_i\rangle = \sqrt{p_i}|\psi_i\rangle$). Given an unknown state $|\psi_?\rangle$, picked in accordance with these probabilities, the quantity we are interested in is the average probability of success for a given generalised measurement (POVM) to distinguish which state we were given. For a measurement M (given by a set of positive operators $\{M_i\}$ summing to the identity), let this probability be denoted by $P^M(\mathcal{E})$. Then we have

$$P^M(\mathcal{E}) = \sum_i \langle\psi'_i|M_i|\psi'_i\rangle = \sum_i p_i \langle\psi_i|M_i|\psi_i\rangle \tag{5.1}$$

$M^{opt}(\mathcal{E})$ will denote the measurement with the optimal probability of success, and in an abuse of notation $P^{opt}(\mathcal{E})$ will denote this optimal probability. We call this the optimal probability of distinguishing the states in $\mathcal{E}$.

We use three matrix norms: the Euclidean (Frobenius) norm $\|A\|_2 = \sqrt{\mathrm{tr}A^\dagger A} = \sqrt{\sum_{i,j} |A_{ij}|^2}$, the trace norm $\|A\|_1 = \mathrm{tr}\sqrt{A^\dagger A} = \sum_i \sigma_i(A)$, where $\sigma_i(A)$ denotes the i'th singular value of A, and the l_1 norm $\sum_{i,j} |A_{ij}|$. We will often use the $d \times n$ *state matrix* $S = S(\mathcal{E}) = (|\psi'_1\rangle, \ldots, |\psi'_n\rangle)$ whose i'th column is the state $|\psi'_i\rangle$. Then $G = S^\dagger S$ gives the $n \times n$ Gram matrix [75] encoding all the inner products between the renormalised states in $\mathcal{E}$. If $n < d$, G will have $d - n$ zero eigenvalues. Note that every rectangular matrix M with $\|M\|_2 = 1$ is a state matrix. ρ will represent the density matrix of the ensemble:

$$\rho = \sum_{i=1}^{n} |\psi'_i\rangle\langle\psi'_i| \tag{5.2}$$

It is well-known [83] that G and ρ have the same non-zero eigenvalues.

5.2.1 Use of the "pretty good measurement"

We will use a specific measurement to provide bounds on $P^{opt}(\mathcal{E})$, which is "canonical" in the sense that it performs reasonably well for any ensemble $\mathcal{E}$. This is the so-called *pretty good measurement* (PGM), which was independently identified by several authors (e.g. see [67, 68]) and has a number of useful properties. It is usually defined as a set of projectors $\{|\nu_i\rangle\langle\nu_i|\}$ onto "measurement vectors" $|\nu_i\rangle$, where $|\nu_i\rangle = \rho^{-1/2}|\psi'_i\rangle$ (the inverse only being taken on the support of ρ). However, it may also be defined implicitly, which brings out its "canonical" nature.

To this end, consider an arbitrary measurement M for $\mathcal{E}$ that consists of a set of n rank 1 projectors onto unnormalised measurement vectors $|\mu_i\rangle$, where each measurement vector corresponds to a state $|\psi'_i\rangle$ in the ensemble. (In fact, it turns out that the optimal measurement for an ensemble of pure states always falls into this category [51].) The probability of getting measurement outcome i and receiving state j is then $|\langle\mu_i|\psi'_j\rangle|^2$, and the overall probability of success of this measurement is $\sum_{i=1}^{n}|\langle\mu_i|\psi'_i\rangle|^2$. We may thus encode all the inner products (and hence the probabilities) in a matrix P, where $P_{ij} = \langle\mu_i|\psi'_j\rangle$; and rather than looking for an optimal measurement M, we can rephrase our task as looking for an optimal matrix P that corresponds to a valid measurement.

We have the following requirement on P, from the fact that M must be a valid POVM.

$$(P^\dagger P)_{ij} = \sum_{k=1}^{n}\langle\psi'_i|\mu_k\rangle\langle\mu_k|\psi'_j\rangle = \langle\psi'_i|\left(\sum_{k=1}^{n}|\mu_k\rangle\langle\mu_k|\right)|\psi'_j\rangle = G_{ij} = (S^\dagger S)_{ij} \tag{5.3}$$

A natural way to produce a matrix P that satisfies this condition from any given S is to take $P = \sqrt{G}$, the positive semidefinite square root of G. The PGM turns out to be a measurement corresponding to this matrix P, for, if $P_{ij} = \langle\nu_i|\psi'_j\rangle$, then

$$(P^2)_{ij} = \sum_{k=1}^{n}\langle\psi'_i|\rho^{-1/2}|\psi'_k\rangle\langle\psi'_k|\rho^{-1/2}|\psi'_j\rangle = \langle\psi'_i|\left(\rho^{-1/2}\sum_{k=1}^{n}|\psi'_k\rangle\langle\psi'_k|\rho^{-1/2}\right)|\psi'_j\rangle = G_{ij} \tag{5.4}$$

The probability of success for the PGM is thus given by $P^{pgm}(\mathcal{E}) = \sum_{i=1}^{n}(\sqrt{G})^2_{ii}$. Barnum and Knill have proved [17] that the PGM has the further property that it is almost optimal in the following sense.

Theorem 5.2.1. (Barnum, Knill) [17] $P^{pgm}(\mathcal{E}) \geq P^{opt}(\mathcal{E})^2$.

For completeness, we now give a simplified proof of Barnum and Knill's result in the case of pure states.

Proof. Consider an arbitrary POVM R consisting of measurement operators $\{R_i\}$, and an arbitrary ensemble $\mathcal{E}$ of renormalised states $\{|\psi'_i\rangle\}$, with a priori probabilities p_i,

where as before $|\psi'_i\rangle = \sqrt{p_i}|\psi_i\rangle$ and $\rho = \sum_{i=1}^n |\psi'_i\rangle\langle\psi'_i|$. Assume w.l.o.g. that $R_i = |\mu_i\rangle\langle\mu_i|$ for some vectors $|\mu_i\rangle$, as the optimal measurement will always be of this form [52]. Then

$$\begin{aligned} P^R(\mathcal{E}) &= \sum_{i=1}^n \langle\psi'_i|R_i|\psi'_i\rangle = \sum_{i=1}^n |\langle\psi'_i|\mu_i\rangle|^2 = \sum_{i=1}^n |\langle\psi'_i|\rho^{-1/4}\rho^{1/4}|\mu_i\rangle|^2 && (5.5)\\ &\leq \sum_{i=1}^n \langle\psi'_i|\rho^{-1/2}|\psi'_i\rangle\langle\mu_i|\rho^{1/2}|\mu_i\rangle && (5.6)\\ &\leq \sqrt{\left(\sum_{i=1}^n \langle\psi'_i|\rho^{-1/2}|\psi'_i\rangle^2\right)\left(\sum_{j=1}^n \langle\mu_j|\rho^{1/2}|\mu_j\rangle^2\right)} && (5.7)\\ &\leq \sqrt{\sum_{i=1}^n \langle\psi'_i|\rho^{-1/2}|\psi'_i\rangle^2} = \sqrt{P^{pgm}(\mathcal{E})} && (5.8) \end{aligned}$$

The first and second inequalities are Cauchy-Schwarz inequalities, and the third follows because the vectors $\{\rho^{1/2}|\mu_i\rangle\}$ can easily be seen to define an ensemble with density matrix ρ:

$$\sum_{i=1}^n \rho^{1/2}|\mu_i\rangle\langle\mu_i|\rho^{1/2} = \rho^{1/2}\left(\sum_{i=1}^n |\mu_i\rangle\langle\mu_i|\right)\rho^{1/2} = \rho \qquad (5.9)$$

and we therefore have $\sum_{i=1}^n \langle\mu_i|\rho^{1/2}|\mu_i\rangle^2 \leq 1$, as this is the probability of success of the measurement R applied to this ensemble. □

We have thus shown the overall relationship $P^{opt}(\mathcal{E})^2 \leq P^{pgm}(\mathcal{E}) \leq P^{opt}(\mathcal{E})$.

5.2.2 Bounds from the pairwise inner products

A set of states that are pairwise almost orthogonal are pairwise highly distinguishable. It thus seems intuitively clear that, given such a set, the probability of success in distinguishing one state from *all* the others must also be high. However, this intuition is wrong. This was noted by Jozsa and Schlienz [83], who showed that the inner products of an ensemble of states may all be reduced, while simultaneously reducing the von Neumann entropy of the ensemble (which gives a measure of overall distinguishability). This effect also manifests itself in quantum fingerprinting [30]. Here, d-dimensional states are "compressed" to $O(\log d)$-dimensional "fingerprint" states that can be distinguished pairwise. However, given such a fingerprint the corresponding original state may not be identified, as this would violate Holevo's theorem [74].

Nevertheless, for certain ensembles the pairwise inner products can give a good lower bound on the overall distinguishability, as noted by several authors [67, 17]. In this section, we derive such a bound. Our approach is based on that of Hausladen et al. [67], who found a parabola giving a lower bound on the square root function, which

is useful because of the following lemma.

Lemma 5.2.2. *If the function $\sqrt{x}$ is bounded below by $f(x) = ax + bx^2$ for $x \geq 0$, then $(\sqrt{G})_{ii} \geq aG_{ii} + b\sum_{j=1}^{n} |G_{ij}|^2$.*

Proof. G is a positive semidefinite matrix and thus may be diagonalised: $G = UDU^\dagger$, where $D = diag(\{\lambda_i\})$ and $U = (u_{ij})$ is unitary. Working out the matrix algebra shows that $(\sqrt{G})_{ii} = \sum_{k=1}^{n} \sqrt{\lambda_k}|u_{ik}|^2$, so $(\sqrt{G})_{ii} \geq \sum_{k=1}^{n} f(\lambda_k)|u_{ik}|^2 = f(G)_{ii}$. But $f(G)_{ii} = (aG + bG^2)_{ii} = aG_{ii} + b\sum_{j=1}^{n} G_{ij}G_{ji} = aG_{ii} + b\sum_{j=1}^{n} |G_{ij}|^2$. □

Our goal will be to find a and b to parametrise f such that $aG_{ii} + b\sum_{j=1}^{n} |G_{ij}|^2$ is maximised. It is clear that, for this to be maximised, $f(r)$ must equal $\sqrt{r}$ for some r (or we could just increase a or b). So we will pick a and b such that $f(r) = \sqrt{r}$ and $f'(r) = \frac{1}{2\sqrt{r}}$ (i.e. the curves are tangent at this point). This leads to the simultaneous equations

$$ar + br^2 = \sqrt{r},\ a + 2br = \frac{1}{2\sqrt{r}} \tag{5.10}$$

Solving for a and b gives the optimal values

$$a = \frac{3}{2\sqrt{r}},\ b = -\frac{1}{2r^{3/2}} \tag{5.11}$$

To see that $f(x)$ actually is a lower bound for $\sqrt{x}$ for any positive value of r (with these values for a and b), note that the only solutions to the related equation $f(x)^2 = x$ are $x = 0$, $x = r$, or $x = 4r$. As $f(4r)$ is negative, we have that $f(x) = \sqrt{x}$ if and only if $x = 0$ or $x = r$. So the only remaining possibility is that $f(x) > \sqrt{x}$ for all $0 < x < r$. Plugging in a suitable value of x (e.g. $r/2$) shows that this is not the case. The expression $aG_{ii} + b\sum_{j=1}^{n} |G_{ij}|^2$ may now be expressed solely in terms of r. Optimising this for r gives that the maximum is found at the point

$$r = \frac{\sum_{j=1}^{n} |G_{ij}|^2}{G_{ii}} \tag{5.12}$$

Returning to the original inequality, we have

$$(\sqrt{G})_{ii} \geq \frac{3}{2\sqrt{r}}G_{ii} - \frac{1}{2r^{3/2}}\sum_{j=1}^{n} |G_{ij}|^2 \Rightarrow (\sqrt{G})_{ii}^2 \geq \frac{G_{ii}^3}{\sum_{j=1}^{n} |G_{ij}|^2} \tag{5.13}$$

We thus have the following bound on the probability of distinguishing the states in $\mathcal{E}$.

$$P^{pgm}(\mathcal{E}) \geq \sum_{i=1}^{n} \frac{\langle\psi_i'|\psi_i'\rangle^3}{\sum_{j=1}^{n} |\langle\psi_i'|\psi_j'\rangle|^2} = \sum_{i=1}^{n} \frac{p_i^2}{\sum_{j=1}^{n} p_j|\langle\psi_i|\psi_j\rangle|^2} \tag{5.14}$$

If all the states have equal a priori probabilities, the bound simplifies further to

$$P^{pgm}(\mathcal{E}) \geq \frac{1}{n}\sum_{i=1}^{n}\frac{1}{\sum_{j=1}^{n}|\langle\psi_i|\psi_j\rangle|^2} \tag{5.15}$$

The bound (5.14) is always positive and greater than or equal to $\sum_{i=1}^{n} p_i^2$, thus showing that the PGM always does at least as well as the "non-measurement" of guessing which state was received in accordance with their a priori probabilities. For comparison, the bound of [67], obtained for the uniform distribution, used the parabola $f(x) = \frac{3}{2}x - \frac{1}{2}x^2$ to give the simpler expression

$$P^{pgm}(\mathcal{E}) \geq 1 - \frac{1}{n}\sum_{i\neq j}|\langle\psi_i|\psi_j\rangle|^2 \tag{5.16}$$

5.2.3 Bounds from eigenvalues

The eigenvalues of a Hermitian matrix are closely related to its diagonal elements; indeed, the former majorises the latter [75]. With this in mind, we look for a bound on the unknown diagonal elements of $\sqrt{G}$ in terms of the known eigenvalues $\{\lambda_i\}$ of G.

Lemma 5.2.3.

$$P^{pgm}(\mathcal{E}) \geq \frac{1}{n}\left(\sum_{i=1}^{n}\sqrt{\lambda_i}\right)^2 = \frac{1}{n}\|S\|_1^2$$

Proof. By the fact that the trace of a matrix is the sum of its eigenvalues, we have

$$\sum_{i=1}^{n}(\sqrt{G})_{ii} = \sum_{i=1}^{n}\sqrt{\lambda_i} \tag{5.17}$$

$$\Rightarrow \quad \left(\sum_{i=1}^{n}(\sqrt{G})_{ii}\right)^2 = \left(\sum_{i=1}^{n}\sqrt{\lambda_i}\right)^2 \tag{5.18}$$

$$\Rightarrow \quad n\sum_{i=1}^{n}(\sqrt{G})_{ii}^2 \geq \left(\sum_{i=1}^{n}\sqrt{\lambda_i}\right)^2 \tag{5.19}$$

$$\Rightarrow \quad P^{pgm}(\mathcal{E}) \geq \frac{1}{n}\left(\sum_{i=1}^{n}\sqrt{\lambda_i}\right)^2 \tag{5.20}$$

where in (5.19) we used a Cauchy-Schwarz inequality, showing that equality can only be attained in step (5.19) when all the $(\sqrt{G})_{ii}$ are equal. □

Interestingly, this bound is the same as the fidelity of G with the maximally mixed state I/n, where the fidelity $F(\rho,\sigma)$ is defined as $\left(\mathrm{tr}\,\sqrt{\rho^{1/2}\,\sigma\,\rho^{1/2}}\right)^2$ [126, 82].

5.2.4 Distinguishability of unitary operators

We briefly mention a related problem: distinguishing between unitary operators. In this scenario, we are given an unknown d-dimensional unitary operator $U_?$ picked at random from some set $\mathcal{E} = \{U_k\}$ with a priori probabilities $\{p_k\}$, and must input some state to $U_?$ (perhaps appending an ancilla), perform a measurement, then output a guess as to which operator we were given. This is clearly possible with certainty if and only if there exists some $|\psi\rangle$ such that $\langle\psi|(U_k^\dagger \otimes I)(U_{k'} \otimes I)|\psi\rangle = 0$ for all $k \neq k'$; when this is not the case, the goal is to find $|\psi\rangle$ to optimise the success probability. Interestingly (and in contrast to the problem of distinguishing quantum states), for any pair of unitary operators U_1, U_2, there exists some finite number of copies n and a state $|\psi\rangle$ such that $\langle\psi|(U_1^\dagger)^{\otimes n} U_2^{\otimes n}|\psi\rangle = 0$ [3].

We can obtain lower bounds on the distinguishability of a set of unitary operators using the following reduction to the problem of distinguishing quantum states. Append a d-dimensional ancilla and input the maximally entangled state $|\psi\rangle = \sum_{i=0}^{d-1} |i\rangle|i\rangle$ to $U_?$. We then have

$$\begin{aligned} \langle\psi|(U_k^\dagger \otimes I)(U_{k'} \otimes I)|\psi\rangle &= \sum_{i,j=0}^{d-1} \langle i|\langle i|(U_k^\dagger U_{k'}) \otimes I|j\rangle|j\rangle & (5.21) \\ &= \sum_{i,j=0}^{d-1} \langle i|U_k^\dagger U_{k'}|j\rangle\langle i|j\rangle = \sum_{i=0}^{d-1} \langle i|U_k^\dagger U_{k'}|i\rangle = \mathrm{tr}(U_k^\dagger U_{k'}) & (5.22) \end{aligned}$$

which is precisely the Hilbert-Schmidt inner product between U_k and $U_{k'}$ [75]. This implies that the results of the previous two sections can be applied to the Gram matrix $G_{xy} = \mathrm{tr}(U_x^\dagger U_y)$ of a set of unitary operators to give lower bounds on the distinguishability of that set.

5.2.5 Distinguishing mixed states

It is natural to ask to what extent the preceding lower bounds hold for the generalised problem of distinguishing an ensemble $\mathcal{E}$ consisting of mixed states $\{\rho_i\}$. The following lemma allows the problem to be related to that of distinguishing pure states.

Lemma 5.2.4. *Let $\mathcal{E}$ be an ensemble of n d-dimensional mixed states $\{\rho_i\}$ with a priori probabilities $\{p_i\}$, and having spectral decompositions $\rho_i = \sum_{k=1}^{d} \lambda_{ik}|v_{ik}\rangle\langle v_{ik}|$. Let $\mathcal{F}$ be an ensemble of the nd pure states given by the eigenvectors $\{|v_{ik}\rangle\}$ with a priori probabilities $\{p_i\lambda_{ik}\}$. Then $P^{pgm}(\mathcal{E}) \geq P^{pgm}(\mathcal{F})$.*

Proof. For mixed states, the PGM is defined by the following measurement operators $\{M_i\}$:

$$M_i = \rho^{-1/2}\rho_i'\rho^{-1/2}, \text{ where } \rho_i' = p_i\rho_i \text{ and } \rho = \sum_{i=1}^{n} \rho_i' \qquad (5.23)$$

So the probability of success can be bounded as follows, where we use the renormalised eigenvectors $|v'_{ik}\rangle = \sqrt{p_i}\sqrt{\lambda_{ik}}|v_{ik}\rangle$.

$$\begin{aligned}
P^{pgm}(\mathcal{E}) &= \sum_{i=1}^{n} \mathrm{tr}\left(\rho^{-1/2}\rho'_i\rho^{-1/2}\rho'_i\right) && (5.24)\\
&= \sum_{i=1}^{n} \mathrm{tr}\left(\rho^{-1/2}\left(\sum_{k=1}^{d}|v'_{ik}\rangle\langle v'_{ik}|\right)\rho^{-1/2}\left(\sum_{l=1}^{d}|v'_{il}\rangle\langle v'_{il}|\right)\right) && (5.25)\\
&= \sum_{i=1}^{n}\sum_{k,l=1}^{d} \mathrm{tr}\left(\rho^{-1/2}|v'_{ik}\rangle\langle v'_{ik}|\rho^{-1/2}|v'_{il}\rangle\langle v'_{il}|\right) && (5.26)\\
&= \sum_{i=1}^{n}\sum_{k,l=1}^{d} |\langle v'_{ik}|\rho^{-1/2}|v'_{il}\rangle|^2 && (5.27)\\
&\geq \sum_{i=1}^{n}\sum_{k=1}^{d} |\langle v'_{ik}|\rho^{-1/2}|v'_{ik}\rangle|^2 = P^{pgm}(\mathcal{F}) && (5.28)
\end{aligned}$$

□

Therefore, if the eigenvalues and eigenvectors of the states $\{\rho_i\}$ are known, the lower bounds given previously may be applied. If not, a weaker lower bound based only on the pairwise fidelities of the states may be given (where, as before, we set $F(\rho,\sigma) = \left(\mathrm{tr}\ \sqrt{\rho^{1/2}\,\sigma\,\rho^{1/2}}\right)^2$).

Theorem 5.2.5. *Let $\mathcal{E}$ be an ensemble of n mixed states $\{\rho_i\}$ with a priori probabilities $\{p_i\}$. Then*

$$P^{pgm}(\mathcal{E}) \geq \sum_{i=1}^{n} \frac{p_i^2\,\mathrm{tr}(\rho_i^2)}{\sum_{j=1}^{n} p_j F(\rho_i,\rho_j)} \tag{5.29}$$

Proof. From the bound (5.14) and Lemma 5.2.4, we have

$$\begin{aligned}
P^{pgm}(\mathcal{E}) &\geq \sum_{i=1}^{n}\sum_{k=1}^{d} \frac{p_i^2\lambda_{ik}^2}{\sum_{j=1}^{n}\sum_{l=1}^{d} p_j\lambda_{jl}|\langle v_{ik}|v_{jl}\rangle|^2} && (5.30)\\
&= \sum_{i=1}^{n}\sum_{k=1}^{d} \frac{p_i^2\lambda_{ik}^2}{\sum_{j=1}^{n} p_j\langle v_{ik}|\left(\sum_{l=1}^{d}\lambda_{jl}|v_{jl}\rangle\langle v_{jl}|\right)|v_{ik}\rangle} && (5.31)\\
&= \sum_{i=1}^{n}\sum_{k=1}^{d} \frac{p_i^2\lambda_{ik}^2}{\sum_{j=1}^{n} p_j\langle v_{ik}|\rho_j|v_{ik}\rangle} && (5.32)\\
&\geq \sum_{i=1}^{n}\sum_{k=1}^{d} \frac{p_i^2\lambda_{ik}^2}{\sum_{j=1}^{n} p_j F(\rho_i,\rho_j)} = \sum_{i=1}^{n} \frac{p_i^2\,\mathrm{tr}(\rho_i^2)}{\sum_{j=1}^{n} p_j F(\rho_i,\rho_j)} && (5.33)
\end{aligned}$$

□

This bound gets progressively worse as the states in $\mathcal{E}$ get more mixed. One might expect the following lower bound to hold for mixed states, as it is the obvious extension

of the bound (5.14) for pure states, but interestingly it does not.

$$P^{pgm}(\mathcal{E}) \not\geq \sum_{i=1}^{n} \frac{p_i^2}{\sum_{j=1}^{n} p_j F(\rho_i, \rho_j)} \tag{5.34}$$

A simple counterexample is given by the equiprobable ensemble consisting of the following two three-dimensional states.

$$\rho_1 = \begin{pmatrix} \frac{1}{2} & 0 & 0 \\ 0 & \frac{1}{2} & 0 \\ 0 & 0 & 0 \end{pmatrix}, \; \rho_2 = \begin{pmatrix} \frac{1}{2} & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & \frac{1}{2} \end{pmatrix} \tag{5.35}$$

5.2.6 Upper bounds on distinguishability

The final question in quantum measurement theory that we mention is whether converses of the inner product and eigenvalue bounds can be found that would give *upper* bounds on the success probability of distinguishing between a set of quantum states. Such bounds might be useful in the fields of quantum cryptography or quantum query complexity.

Firstly, it is worth noting that no upper bound on the success probability in terms of the eigenvalues alone can be found[1], for the following reason. Any set of eigenvalues $\{\lambda_i\}$ summing to 1 can give rise to a Gram matrix G where $G_{ii} = \lambda_i$, and $G_{ij} = 0$ (for $i \neq j$). Such matrices correspond to an ensemble $\mathcal{E}$ of perfectly distinguishable states where $P^{pgm}(\mathcal{E}) = 1$.

However, following the completion of this thesis, an upper bound has been found that is given in terms of the pairwise fidelities of the states in $\mathcal{E}$ [103]; the bound also extends to mixed states (recall that the fidelity $F(\rho, \sigma)$ is defined as $\left(\text{tr } \sqrt{\rho^{1/2}\, \sigma\, \rho^{1/2}}\right)^2$ [126, 82]). We content ourselves with stating it below.

Theorem 5.2.6. *Let $\mathcal{E}$ be an ensemble of quantum states $\{\rho_i\}$ with a priori probabilities $\{p_i\}$. Then, for any measurement M,*

$$P^M(\mathcal{E}) \leq 1 - \sum_{i>j} p_i p_j F(\rho_i, \rho_j)$$

5.3 The distinguishability of states with constant inner product

An illustrative case to which the above lower bounds can be applied is that of equiprobable states where the pairwise inner products are all equal, so the states are all equally distinguishable from each other. Consider an ensemble $\mathcal{E}$ with Gram matrix G, where $G_{ii} = 1/n$ and $G_{ij} = p/n$ for $i \neq j$ (and p is a positive real constant). In this case, the

[1] As future work, it would be interesting to determine whether an upper bound (or an improved lower bound) could be produced by considering the diagonal entries of G as well as its eigenvalues.

inner product bound of Section 5.2.2 gives the bound

$$P^{pgm}(\mathcal{E}) \geq \frac{1}{1+p^2(n-1)} = O(1/n) \tag{5.36}$$

The eigenvalue bound, however, gives much better results. The symmetry of G shows immediately that it has an eigenvector $(1,1,\ldots,1)$; the corresponding eigenvalue is $\lambda_1 = p+(1-p)/n$. The set of eigenvectors may be completed by taking any $n-1$ vectors orthogonal to $(1,1,\ldots,1)$, which will be eigenvectors with eigenvalues $\lambda_{2\ldots n} = (1-p)/n$. We therefore have

$$\begin{aligned} P^{pgm}(\mathcal{E}) &\geq \frac{1}{n}\left(\sqrt{p+\frac{1-p}{n}} + (n-1)\sqrt{\frac{1-p}{n}}\right)^2 & (5.37)\\ &\geq \frac{1}{n}\left((n-1)^2\frac{(1-p)}{n}\right) \geq (1-p) - \frac{2(1-p)}{n} & (5.38) \end{aligned}$$

so the probability of distinguishing these states approaches a constant as $n \to \infty$. In fact, one can show that inequality (5.37) is actually an equality giving the precise probability of success $P^{pgm}(\mathcal{E})$ (this follows from showing that the diagonal entries of $\sqrt{G}$ are all equal).

Such an ensemble therefore provides a kind of converse to the ensemble of states used in quantum fingerprinting [30]: in this case, no matter how many states there are in the ensemble, their joint distinguishability is of the same order as their pairwise distinguishability. We will see below that this behaviour is not typical; however, it is perhaps not surprising, because $\mathcal{E}$ can only be realised in n dimensions. To see this, note that G is non-singular, so the states in $\mathcal{E}$ must be linearly independent.

5.4 The overlap of random quantum states

We now turn to the distinguishability of random quantum states. As a preamble, this section contains a derivation of the precise distribution of the overlap between random pure quantum states, i.e. their pairwise distinguishability[2].

The standard definition of a random d-dimensional quantum state is a vector picked uniformly at random from the complex unit sphere. This may be produced by writing down a d-dimensional vector v, each of whose components are complex Gaussians (say $v_i \sim \widetilde{N}(0,1/d)$, i.e. v_i has probability density function $\frac{d}{\pi}e^{-d|v_i|^2}$), and normalising the result. To see that this gives a vector uniformly distributed on the sphere, note that the joint probability distribution of the entries of v depends only on the norm of v, and is thus uniform on the sphere when v is normalised.

Before discussing the standard complex case, we first calculate the overlap between

[2]This is probably well-known but is hard to find in the literature. The fidelity of random mixed states is considered in [135].

random real quantum states, where $v_i \sim N(0, 1/d)$, i.e. v_i has probability density function $\frac{\sqrt{d}}{\sqrt{2\pi}} e^{-dv_i^2/2}$. See [81] for definitions and properties of the probability distributions used below.

Lemma 5.4.1. *Let $|r\rangle$, $|s\rangle$ be random unit vectors in $\mathbb{R}^d$. The overlap $|\langle r|s\rangle|^2$ follows a beta distribution with parameters $\alpha = 1/2$, $\beta = (d-1)/2$ – i.e. $|\langle r|s\rangle|^2 \sim \beta(1/2, (d-1)/2)$.*

Proof. Without loss of generality, we will show that $|\langle 0|r\rangle|^2 \sim \beta(1/2, (d-1)/2)$. This is simply equal to the square of the first component of $|r\rangle$. We will use the random variable r_i to denote the i'th component of $|r\rangle$, and w_i to denote a normally distributed random variable. Then, by the parametrisation discussed above,

$$r_1^2 = \frac{w_1^2}{\sum_{i=1}^d w_i^2} \tag{5.39}$$

Now $w_i^2 \sim \chi^2(1)$, so $\sum_{i=2}^d w_i^2 \sim \chi^2(d-1)$. A standard result relating statistical distributions states that, if $X_1 \sim \chi^2(a)$ and $X_2 \sim \chi^2(b)$, $X_1/(X_1 + X_2) \sim \beta(a/2, b/2)$. So we can split the bottom part of the sum into two halves and produce

$$r_1^2 = \frac{w_1^2}{w_1^2 + \sum_{i=2}^d w_i^2} \sim \beta(1/2, (d-1)/2) \tag{5.40}$$

□

Lemma 5.4.2. *Let $|a\rangle$, $|b\rangle$ be random unit vectors in $\mathbb{C}^d$. The overlap $|\langle a|b\rangle|^2$ follows a beta distribution with parameters $\alpha = 1$, $\beta = d - 1$ – i.e. $|\langle a|b\rangle|^2 \sim \beta(1, d-1)$.*

Proof. The proof follows the same lines as that of the previous lemma. The real and complex parts of each component of $|a\rangle$ are normally distributed, so if w_i, x_i represent independent, normally distributed random variables, we have

$$|a_1|^2 = \frac{w_1^2 + x_1^2}{\sum_{i=1}^d w_i^2 + x_i^2} \tag{5.41}$$

so, using the same argument as before, we have that $|\langle a|b\rangle|^2$ is distributed as $|a_1|^2 \sim \beta(1, d-1)$. □

It is worth noting that both cases have the same expectation $(1/d)$, but the real case has approximately double the variance:

$$\mathrm{Var}(|\langle r|s\rangle|^2) = \frac{d-1}{d^2(d/2+1)}, \; \mathrm{Var}(|\langle a|b\rangle|^2) = \frac{d-1}{d^2(d+1)} \tag{5.42}$$

5.5 The distinguishability of random quantum states

We will use Lemma 5.2.3 and some results from the theory of random matrices to put a lower bound on the probability of distinguishing multiple random quantum states. The expected value of this lower bound will be obtained for a quite general notion of "randomness", but in order to get measure concentration results we will specialise to states distributed uniformly at random (according to the Haar measure). The bound holds in the asymptotic regime where the number of states n and the dimension d approach a constant ratio; we also give lower bounds on the rate of convergence.

5.5.1 A little random matrix theory

In this section, we will calculate the expected value of the trace norm of a random matrix, which (by Lemma 5.2.3) is related to the distinguishability of a set of random states. The distribution of the trace norm (i.e. the sum of singular values) of a matrix M is closely related to that of the eigenvalues of the matrix $MM^\dagger$, which is known to statisticians as a (complex) *Wishart matrix*. The distribution of the eigenvalues of a Wishart matrix is given by the Marčenko-Pastur law [101], which is stated in the form we need in [16].

Theorem 5.5.1. (Marčenko-Pastur law) [101]
Let R_r be a family of $d \times n$ matrices with $n \geq d$ and $d/n \to r \in (0,1]$ as $n, d \to \infty$, where the entries of R_r are i.i.d. complex random variables with mean 0 and variance 1. Then, as $n, d \to \infty$, the eigenvalues of the rescaled matrix $\frac{1}{n} R_r R_r^\dagger$ tend to a limiting distribution with density

$$p_r(x) = \frac{\sqrt{(x - A^2)(B^2 - x)}}{2\pi r x} \tag{5.43}$$

for $A^2 \leq x \leq B^2$ (where $A = 1 - \sqrt{r}$, $B = 1 + \sqrt{r}$), and density 0 elsewhere.

We will translate this to a similar statement about the singular values of R_r. The following lemma is straightforward.

Lemma 5.5.2. *Let R_r be a family of $d \times n$ matrices with $k/m \to r \in (0,1]$ as $n, d \to \infty$, where $k = \min(n, d)$ and $m = \max(n, d)$, and the entries of R_r are i.i.d. complex random variables with mean 0 and variance 1. Then, as $n, d \to \infty$, the singular values of $R_r/\sqrt{m}$ tend to a limiting distribution with density*

$$p_r(y) = \frac{\sqrt{(y^2 - A^2)(B^2 - y^2)}}{\pi r y} \tag{5.44}$$

for $A \leq y \leq B$ (where $A = 1 - \sqrt{r}$, $B = 1 + \sqrt{r}$), and density 0 elsewhere.

Proof. The lemma follows from Theorem 5.5.1 for $n \geq d$ by substituting $y = \sqrt{x}$. For $n \leq d$, note that the singular values of R are the same as those of R^T, so the roles of n and d need merely be interchanged. □

Lemma 5.5.3. *Let R_r be a family of $d \times n$ matrices with $k/m \to r \in (0, 1]$ as $n, d \to \infty$, where $k = \min(n, d)$ and $m = \max(n, d)$, and the entries of R_r are i.i.d. complex random variables with mean 0 and variance 1. Then, as $n, d \to \infty$, the expected trace norm of R_r is*

$$\mathbb{E}(\|R_r\|_1) = \frac{m^{3/2}}{\pi} \int_A^B \sqrt{(y^2 - A^2)(B^2 - y^2)}\, dy \tag{5.45}$$

where $A = 1 - \sqrt{r}$, $B = 1 + \sqrt{r}$.

Proof. With probability 1, R_r will have k non-zero singular values. Let $\sigma_i(R_r)$ denote the value of the i'th (unsorted) singular value of R_r, for arbitrary i between 1 and k. We have

$$\mathbb{E}(\|R_r\|_1) = (k\sqrt{m})\, \mathbb{E}(\sigma_i(R_r/\sqrt{m})) = k\sqrt{m} \int_A^B y\, p_r(y)\, dy \tag{5.46}$$

and using Lemma 5.5.2 gives the desired result. □

This turns out to be an elliptic integral which cannot be expressed in terms of elementary functions [63]. However, it is possible to produce a good lower bound, which is tight in the case $r = 1$:

Lemma 5.5.4.

$$\mathbb{E}(\|R_r\|_1) \geq k\sqrt{m}\sqrt{1 - r\left(1 - \frac{64}{9\pi^2}\right)} \tag{5.47}$$

with equality when $r = 1$.

Proof. Deferred to Section 5.7. □

As these are asymptotic results, it is important to bound the rate of convergence of this expected value to that given by Lemma 5.5.4. This can be done using a theorem of Bai [15], who has shown that the Kolmogorov distance between the (rescaled) expected empirical spectral distribution of an $m \times k$ matrix (with $m \geq k$) and the asymptotic distribution given by the Marčenko-Pastur law is $O(m^{-5/48})$. After some algebra, this may be used with Lemma 5.5.4 to give

$$\mathbb{E}(\|R_r\|_1) \geq k\sqrt{m}\left(\sqrt{1 - r\left(1 - \frac{64}{9\pi^2}\right)} - O(m^{-5/48})\right) \tag{5.48}$$

for a finite-dimensional $m \times k$ matrix R_r.

5.5.2 Random quantum states

We can apply this result, and the lower bound of Lemma 5.2.3, to estimate the distinguishability of random quantum states uniformly distributed on the complex unit sphere in d dimensions. In fact, we may exploit the concentration of measure effects

characteristic of high-dimensional spaces to show lower bounds on the distinguishability of *almost all* ensembles of quantum states.

As discussed in Section 5.4, uniformly random quantum state may be produced by creating a vector v, each of whose components are complex Gaussians (say $v_i \sim \widetilde{N}(0, 1/d)$), and normalising the result. The intuition that this normalisation step is "almost unnecessary" [130] can be formalised as follows. It is straightforward to see that $\mathbb{E}(\|v\|) = 1$. In order to get an explicit expression for the concentration around this expectation the following result from Appendix A of [22] can be used.

Lemma 5.5.5. (Norm concentration of Gaussian vectors) [22]
Let v be a d-dimensional random vector, each of whose components $v_i \sim \widetilde{N}(0, 1/d)$. Then, for any ϵ,

$$\Pr[|\|v\|_2^2 - 1| \geq \epsilon] < 2e^{-d\epsilon^2/12} \tag{5.49}$$

Similarly, the state matrix of an ensemble $\mathcal{E}$ of n equiprobable d-dimensional uniformly random quantum states is given by a $d \times n$ matrix S whose columns are uniformly random quantum states renormalised so that each column has norm $1/\sqrt{n}$. Let S' denote the matrix produced by rescaling each column by $1/\sqrt{n}$, rather than normalising them. We will show that S and S' are close with high probability. Consider an arbitrary column of S and the same column in S', denoted v and v' respectively. Lemma 5.5.5 allows a bound to be put on the probability of v and v' being far apart, as

$$\|v - v'\|_2^2 = \|(\sqrt{n}\|v'\|_2 - 1)v\|_2^2 = \frac{1}{n}(\sqrt{n}\|v'\|_2 - 1)^2 \tag{5.50}$$

We may therefore obtain

$$\begin{aligned} \Pr[\|v - v'\|_2^2 \geq \epsilon] &= \Pr[(\sqrt{n}\|v'\|_2 - 1)^2 \geq n\epsilon] \\ &\leq \Pr[|n\|v'\|_2^2 - 1| \geq n\epsilon] \leq 2e^{-n^2 d\epsilon^2/12} \end{aligned} \tag{5.51}$$

Considering all the columns in the matrices S and S', and using the union bound, we have

$$\Pr[\|S - S'\|_2^2 \geq \epsilon] \leq 2ne^{-d\epsilon^2/12} \tag{5.52}$$

In order to convert this to a statement about the "distinguishability" function $f(S) = \frac{1}{n}\|S\|_1^2$ that we are interested in, we need the following lemma, which is proved in Section 5.8.

Lemma 5.5.6. *Let S be an $n \times d$ matrix with $\|S\|_2 \leq l$, and define $f(S) = \frac{1}{n}\|S\|_1^2$. Then the Lipschitz constant η of f, $\eta = \sup_{x,y} |f(x) - f(y)|/\|x - y\|_2$, satisfies $\eta \leq 2l$.*

Lemma 5.5.6 implies the following relationship, for any $l > 0$.

$$\begin{aligned} \Pr\left[|(\|S'\|_1^2 - \|S\|_1^2)/n| \geq 2l\sqrt{\epsilon}\right] &\leq \Pr[\|S'\|_2 \geq l] + \Pr[\|S - S'\|_2^2 \geq \epsilon] \\ &\leq 2e^{-nd(l^2-1)^2/12} + 2ne^{-d\epsilon^2/12} \end{aligned} \tag{5.53}$$

The final result we will need is the following concentration lemma.

Lemma 5.5.7. (Concentration of Gaussian measure) [92]
Let p be a point in $\mathbb{R}^d$ picked in accordance with standard Gaussian measure. Then

$$\Pr[|f(p) - \mathbb{E}(f)| \geq \epsilon] \leq 2e^{-\epsilon^2/2\eta^2} \tag{5.54}$$

where η is the Lipschitz constant of f, $\eta = \sup_{x,y} |f(x) - f(y)|/\|x - y\|_2$.

We now have all the required ingredients to prove a lower bound on the distinguishability of almost all quantum states.

Theorem 5.5.8. *Let $\mathcal{E}$ be an ensemble of n equiprobable d-dimensional quantum states picked uniformly at random. Set $p = \frac{1}{r}\left(1 - \frac{1}{r}\left(1 - \frac{64}{9\pi^2}\right)\right) - O(n^{-5/48})$ if $n \geq d$, and $p = 1 - r\left(1 - \frac{64}{9\pi^2}\right) - O(d^{-5/48})$ otherwise. Then, for any $\epsilon \leq p/2$,*

$$\Pr[P^{pgm}(\mathcal{E}) \leq p - 2\epsilon] \leq 2\left((n+1)e^{-d\epsilon^4/K} + e^{-nd\epsilon^2/5}\right) \tag{5.55}$$

where K is a constant ≤ 300.

Proof. As before, let S be the state matrix of $\mathcal{E}$, and let S' be the matrix produced by rescaling the vectors of Gaussians which would produce S if they were normalised. The matrix $R = \sqrt{nd}\, S'$ fulfils the criteria for the Marčenko-Pastur law (Theorem 5.5.1), as its entries are complex random variables with mean 0 and variance 1. We therefore have

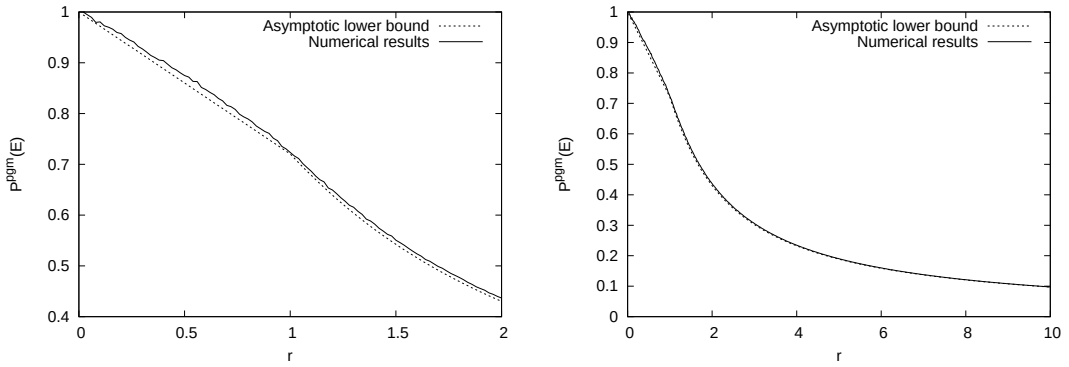
$$\mathbb{E}\left(\frac{1}{n}\|S'\|_1^2\right) \geq \frac{1}{n}\mathbb{E}(\|S'\|_1)^2 = \frac{1}{n^2 d}\mathbb{E}(\|R\|_1)^2 \geq p \tag{5.56}$$

using the lower bound on the expected trace norm of R from Lemma 5.5.4 and the convergence result of Bai [15]. We will show that this implies a bound on $\frac{1}{n}\|S\|_1^2$, and hence (by Lemma 5.2.3) a bound on $P^{pgm}(\mathcal{E})$. From Lemma 5.5.7 (identifying $\mathbb{C}^d$ with $\mathbb{R}^{2d}$) and eqn. (5.53), we have for any l

$$\begin{aligned}
&\Pr\left[\|S\|_1^2/n \leq p - 2\epsilon\right] && (5.57)\\
&\quad \leq \Pr\left[|(\|S'\|_1^2 - \|S\|_1^2)/n| \geq \epsilon\right] + \Pr\left[|\|S'\|_1^2/n - \mathbb{E}\left(\|S'\|_1^2/n\right)| \geq \epsilon\right] && (5.58)\\
&\quad \leq 2\left(\exp\left(-\frac{nd(l^2-1)^2}{12}\right) + n\exp\left(-\frac{d\epsilon^4}{192\, l^4}\right) + \exp\left(-\frac{nd\epsilon^2}{4\, l^2}\right)\right) && (5.59)\\
&\quad \leq 2\left(\exp\left(-\frac{nd\epsilon^4}{12}\right) + n\exp\left(-\frac{d\epsilon^4}{300}\right) + \exp\left(-\frac{nd\epsilon^2}{5}\right)\right) && (5.60)
\end{aligned}$$

where, in the last line, we pick $l^2 = 1 + \epsilon^2$ and note that $\epsilon \leq 1/2$. □

Despite the large constants that appear in these expressions, Figure 5.1 shows numerical evidence that ensembles $\mathcal{E}$ of quantum states picked uniformly at random in fact appear to have a value of $P^{pgm}(\mathcal{E})$ close to the asymptotic lower bound, even when the states are (relatively) low-dimensional.



(a) $0 \leq r \leq 2$ (b) $0 \leq r \leq 10$

Figure 5.1: Asymptotic bound on $P^{pgm}(\mathcal{E})$ vs. numerical results (averaged over 10 runs) for ensembles of $n = 50r$ 50-dimensional uniformly random states.

5.6 Discussion

This work can be seen as part of an overall programme of understanding the behaviour of random quantum states [69, 112, 130, 135].

There is a fundamental correspondence between the mixed state obtained from an equal mixture of uniformly random pure states, and that produced by starting with a larger system in a uniformly random pure state, and tracing out part of the system. Consider a d-dimensional state

$$\rho_{n,d} = \frac{1}{n} \sum_{i=1}^{n} |\psi_i\rangle\langle\psi_i| \tag{5.61}$$

where each state in the set $\mathcal{E} = \{|\psi_i\rangle\}$ is picked uniformly at random. We can think of $\rho_{n,d}$ as being produced from the following dn-dimensional state (which we consider to live in a Hilbert space $\mathcal{H}_d \otimes \mathcal{H}_n$) by tracing out the second subsystem:

$$|\upsilon\rangle = \frac{1}{\sqrt{n}} \sum_{k=0}^{n-1} |\upsilon_k\rangle|k\rangle = \frac{1}{\sqrt{n}} \sum_{k=0}^{n-1} \sum_{l=0}^{d-1} \alpha_{kl}|l\rangle|k\rangle \tag{5.62}$$

for some coefficients α_{kl}. As mentioned previously, the α_{kl} will be approximately normally distributed as $\widetilde{N}(0, 1/d)$. So, because of the normalisation factor at the front of the sum, the overall state $|\upsilon\rangle$ has coefficients which are approximately normally distributed and scaled as $\widetilde{N}(0, 1/dn)$. Therefore, this state is approximately picked from the uniform distribution on the unit sphere in $\mathbb{C}^{dn}$. Popescu, Short and Winter [112] obtained an upper bound on the expected trace distance of such a state $\rho_{n,d}$ from the maximally mixed state I/d, and used this to show, among other results, that for $n \gg d$, $\rho \approx I/d$.

Because the non-zero eigenvalues of the Gram matrix of (rescaled) states in $\mathcal{E}$ are

the same as the eigenvalues of $\rho_{n,d}$ [83], this chapter can be seen as obtaining a similar result to [112] for the *fidelity* of $\rho_{n,d}$ with the maximally mixed state, via quite different methods. However, the bound is tighter for n close to d, and the notion of "randomness" of the states $\{|\psi_i\rangle\}$ is more general (which is simply a side-effect of relying on the powerful Marčenko-Pastur law).

5.7 Proof of Lemma 5.5.4

In this section we will prove a lemma which immediately implies Lemma 5.5.4. See [63] for the facts used about elliptic integrals and hypergeometric series.

Lemma 5.7.1. *Let $0 \leq r \leq 1$ and $A = 1 - \sqrt{r}$, $B = 1 + \sqrt{r}$. Then*

$$\int_A^B \sqrt{(y^2 - A^2)(B^2 - y^2)}\, dy \geq r\pi \sqrt{1 - r\left(1 - \frac{64}{9\pi^2}\right)} \tag{5.63}$$

with equality at $r = 0$, $r = 1$.

Proof. We have

$$\begin{aligned} f(r) &= \int_A^B \sqrt{(y^2 - A^2)(B^2 - y^2)}\, dy & (5.64)\\ &= \frac{B}{3}\left((A^2 + B^2)E\left(\frac{\sqrt{B^2 - A^2}}{B^2}\right) - 2A^2 K\left(\frac{\sqrt{B^2 - A^2}}{B^2}\right)\right) & (5.65)\\ &= \frac{2(1+\sqrt{r})}{3}\left((1+r)E\left(\frac{2r^{1/4}}{1+\sqrt{r}}\right) - (1-\sqrt{r})^2 K\left(\frac{2r^{1/4}}{1+\sqrt{r}}\right)\right) & (5.66) \end{aligned}$$

where $K(r)$ and $E(r)$ are the complete elliptic integrals of the first and second kind, respectively:

$$K(r) = \int_0^1 \frac{dx}{\sqrt{(1-x^2)(1-r^2x^2)}}\,,\; E(r) = \int_0^1 \frac{\sqrt{1-r^2x^2}}{\sqrt{1-x^2}} dx \tag{5.67}$$

Note that $f(r)$ may be evaluated explicitly for $r = 0$ and $r = 1$, giving 0 and 8/3 respectively. Now we may apply a standard change of variables (Landen's transformation) to both elliptic integrals, giving

$$\begin{aligned} f(r) &= \frac{2(1+\sqrt{r})}{3}\left(\frac{1+r}{1+\sqrt{r}}\left(2E(\sqrt{r}) - (1-r)K(\sqrt{r})\right) - (1-\sqrt{r})^2(1+\sqrt{r})K(\sqrt{r})\right)\\ &= \frac{4}{3}\left((1+r)E(\sqrt{r}) - (1-r)K(\sqrt{r})\right) & (5.68) \end{aligned}$$

We now move to the representation of $K(r)$ and $E(r)$ as hypergeometric series, which

are defined as follows (using the notation $a^{\bar{n}} = a(a+1)\cdots(a+n-1)$).

$$ {}_2F_1(a,b;c;r) = \sum_{n=0}^{\infty} \frac{a^{\bar{n}} b^{\bar{n}}}{c^{\bar{n}} n!} r^n \tag{5.69} $$

$$ K(r) = (\pi/2)\, {}_2F_1(1/2,1/2;1;r^2)\,,\; E(r) = (\pi/2)\, {}_2F_1(-1/2,1/2;1;r^2) \tag{5.70} $$

This has the advantage that, by a transformation rule due to Gauss, we can rewrite $f(r)$ as a single hypergeometric series.

$$ \begin{aligned} f(r) &= \frac{2\pi}{3}\left((1+r)\, {}_2F_1(-1/2,1/2;1;r) - (1-r)\, {}_2F_1(1/2,1/2;1;r)\right) && (5.71)\\ &= \pi r\, {}_2F_1(-1/2,1/2;2;r) && (5.72) \end{aligned} $$

Returning to the original inequality, our task has been simplified to showing that

$$ g(r) = {}_2F_1(-1/2,1/2;2;r)^2 \geq 1 - r\left(1 - \frac{64}{9\pi^2}\right) \tag{5.73} $$

Evaluating $g(r)$ at 0 and 1 makes it clear that this is equivalent to showing that $g(r)$ is concave for $0 \leq r \leq 1$, which would follow from showing the second derivative $g''(r)$ to be negative in this region. From the rules governing differentiation of hypergeometric series, it is easy to show that

$$ g''(r) = \frac{1}{32}\left({}_2F_1(1/2,3/2;3;r)^2 - 2\, {}_2F_1(-1/2,1/2;2;r)_2F_1(3/2,5/2;4;r)\right) \tag{5.74} $$

The following hypergeometric transformation allows this to be simplified.

$$ \begin{aligned} {}_2F_1(a,b;c;r) &= (1-r)^{c-a-b}{}_2F_1(c-a,c-b;c;r)\\ \Rightarrow g''(r) &= \frac{1}{32}\big((1-r)^2{}_2F_1(5/2,3/2;3;r)^2\\ &- 2(1-r)^2\, {}_2F_1(5/2,3/2;2;r)\, {}_2F_1(3/2,5/2;4;r)\big) \end{aligned} $$

We will show that ${}_2F_1(5/2,3/2;3;r)^2 \leq {}_2F_1(5/2,3/2;2;r)\, {}_2F_1(5/2,3/2;4;r)$ for all positive r, implying that $g''(r)$ is negative in this region. We write out the two hypergeometric series explicitly, setting $k_n = r^n(5/2)^{\bar{n}}(3/2)^{\bar{n}}/n!$.

$$ {}_2F_1(5/2,3/2;3;r)^2 = \sum_{m,n=0}^{\infty} \frac{k_m k_n}{3^{\bar{m}} 3^{\bar{n}}} \tag{5.75} $$

$$ \begin{aligned} {}_2F_1(5/2,3/2;2;r)\, {}_2F_1(5/2,3/2;4;r) &= \sum_{m,n=0}^{\infty} \frac{k_m k_n}{4^{\bar{m}} 2^{\bar{n}}} && (5.76)\\ &= \sum_{m,n=0}^{\infty} \frac{k_m k_n}{3^{\bar{m}} 3^{\bar{n}}} \left(\frac{3}{3+m}\right)\left(\frac{2+n}{2}\right) && (5.77) \end{aligned} $$

$$= \sum_{m=0}^{\infty} \frac{k_m^2}{3^{\bar{m}}3^{\bar{m}}} \left(\frac{6+3m}{6+2m} \right) + \sum_{\substack{m,n=0 \\ m>n}}^{\infty} \frac{k_m k_n}{3^{\bar{m}}3^{\bar{n}}} \left(\frac{3(2+n)}{2(3+m)} + \frac{3(2+m)}{2(3+n)} \right) \tag{5.78}$$

$$\geq \sum_{m=0}^{\infty} \frac{k_m^2}{3^{\bar{m}}3^{\bar{m}}} + \sum_{\substack{m,n=0 \\ m>n}}^{\infty} \frac{2k_m k_n}{3^{\bar{m}}3^{\bar{n}}} = {}_2F_1(5/2, 3/2; 3; r)^2 \tag{5.79}$$

where elementary methods can be used to show that the bracketed last term in eqn. (5.78) is at least 2 for any non-negative m and n. This completes the proof of the lemma. □

The difference between the approximation (5.63) and the actual value of the integral (evaluated numerically) is plotted in Figure 5.2.

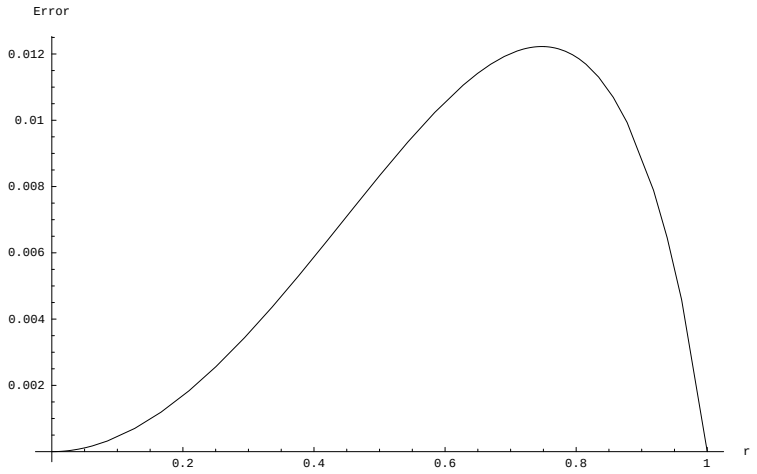


Figure 5.2: Error in approximation to elliptic integral (5.63) for $0 \leq r \leq 1$.

5.8 Lipschitz constants

This section contains derivations of the Lipschitz constants of the functions used for concentration of measure results elsewhere in this thesis (Lemmas 5.5.6 and 6.10.4).

Lemma 5.8.1. *Let S be an $n \times d$ matrix with $\|S\|_2 = 1$, and define $f(S) = \frac{1}{n}\|S\|_1^2$. Then the Lipschitz constant η of f satisfies $\eta \leq 2$.*

Proof. Let $k = \min(n, d)$. We have

$$\begin{aligned}
\eta &= \sup_{S,T} \frac{|f(S) - f(T)|}{\|S - T\|_2} = \sup_{S,T} \frac{|\,\|S\|_1^2 - \|S\|_1^2\,|}{n\|S - T\|_2} && (5.80)\\
&= \sup_{S,T} \left(\frac{\|S\|_1 + \|T\|_1}{n} \right) \frac{|\,\|S\|_1 - \|S\|_1\,|}{\|S - T\|_2} && (5.81)\\
&\leq \sup_{S,T} \left(\frac{\|S\|_1 + \|T\|_1}{n} \right) \frac{\|S - T\|_1}{\|S - T\|_2} && (5.82)\\
&\leq \sup_{S,T} \frac{\sqrt{k}\,(\|S\|_1 + \|T\|_1)}{n} \leq 2k/n \leq 2 && (5.83)
\end{aligned}$$

The first inequality is a triangle inequality, and the second two are derived from

$$\|S\|_1 = \sum_{i=1}^{k} \sigma_i(S) \leq \sqrt{k \sum_{i=1}^{k} \sigma_i^2(S)} \leq \sqrt{k}\|S\|_2 \tag{5.84}$$

which in turn uses a Cauchy-Schwarz inequality. □

Lemma 5.8.2. *Let S be a point on the nd-dimensional hypercube written down as an $n \times d$ $\{-1, 1\}$-matrix, and let $f(S) = \frac{1}{n^2 d}\|S\|_1^2$. Then the Lipschitz constant η of f (with respect to the Hamming distance) satisfies $\eta \leq 4/nd$.*

Proof. The proof is very similar to that of Lemma 5.5.6. As before, let $k = \min(n, d)$. We have

$$\begin{aligned}
\eta &= \sup_{S,T} \frac{|f(S) - f(T)|}{d(S, T)} = \sup_{S,T} \frac{1}{n^2 d} \frac{|\,\|S\|_1^2 - \|S\|_1^2\,|}{d(S, T)} && (5.85)\\
&\leq \sup_{S,T} \left(\frac{\|S\|_1 + \|T\|_1}{n^2 d} \right) \frac{\|S - T\|_1}{\frac{1}{2}\|S - T\|_1} && (5.86)\\
&\leq \sup_{S,T} \frac{2\sqrt{k}\,(\|S\|_1 + \|T\|_1)}{n^2 d} \leq 4k/n^2 d \leq 4/nd && (5.87)
\end{aligned}$$

where, extending inequality (5.84), we use $\|S\|_1 \leq \sqrt{k}\|S\|_2 \leq \sqrt{k}\|S\|_1$. □

Chapter 6

Quantum query complexity and oracle identification

6.1 Introduction

In the study of quantum computation we are often concerned with finding efficient quantum circuits for solving some computational problem, and conversely with showing that a given problem admits no efficient quantum circuit. It is an unfortunate fact that interesting measures of complexity (such as the minimum circuit depth or minimum circuit size to solve a given problem) are often hard to compute.

We therefore lower our sights to a model which does not capture physical efficiency, but which does give lower bounds on circuit size: the model of *query complexity*. This chapter contains a brief introduction to the topic of quantum query complexity, followed by a derivation of the fundamental lower bound on the quantum query complexity of unstructured search, which has a simple application to bounding the query complexity of the Boolean satisfiability problem (SAT). We then specialise (Section 6.5 onwards) to a specific problem in this framework – the oracle identification problem.

In the most general version of the query complexity model, a problem P is defined by a set of problem instances A, a set of *oracle* functions $\{f_a : a \in A\}$ corresponding to these problem instances, and a function g. Our goal is to compute $g(a)$, where $a \in A$ is a specific problem instance. We do not know a at the start of the algorithm, but can find out information about a by making calls to f_a. Our quantum algorithm alternates "expensive" oracle query operations with "free" unitary operations which are arbitrary but not problem-dependent, terminating with a measurement, each of whose outcomes is associated with a different value that $g(a)$ may take. The final state of such an algorithm (before this measurement) can thus be written as

$$|\psi_{a,t}\rangle = O_a\, U_t\, O_a\, U_{t-1}\, \cdots O_a\, U_1 |\psi_0\rangle \tag{6.1}$$

where O_a denotes a unitary operator that encodes f_a somehow (see Section 6.2 for a discussion of this point), and the algorithm uses t queries to O_a. We say that the algorithm succeeds with probability p if, for any problem instance a, the algorithm outputs $g(a)$ with probability at least p. Define the bounded-error quantum query complexity of P, $Q_2(P)$, as the minimum number of queries required for any quantum algorithm to compute $g(a)$ with success probability 2/3. Similarly, define the exact quantum query complexity $Q_E(P)$ as the minimum number of queries necessary to succeed with certainty. The deterministic classical decision tree complexity $D(P)$ is the equivalent classical quantity, where we are restricted to classical queries to f_a and must succeed with certainty.

Decision tree complexity has long been a topic of study in classical computer science. The generalisation to quantum query complexity was first made precise by Beals et al. [19], but was implicitly studied previously by other authors (e.g. [64], [21]). See the review [33] for a good introduction to these complexity measures.

How can one prove lower bounds on quantum query complexity? There is essentially only one known "meta-technique": define some quantity that measures the progress of a quantum algorithm solving a problem, show that it must be low at the beginning and high at the end, and show that it cannot increase too much with each query to the oracle (and cannot increase at all with a non-query transformation). This technique can be split into two sub-families: the *polynomial method* [19], where the quantity of interest is the degree of a polynomial representing the function to be computed, and the *adversary method* (e.g. [21, 7]), where we use some measure of "separation" of states corresponding to inputs for which different outputs are required. These two methods have both been successful in lower bounding the quantum query complexity of a variety of problems, but appear (in general) to be incomparable [9].

One situation where quite strong general lower bounds can be put on quantum query complexity is the case of computing a total Boolean function $g : \{0,1\}^n \mapsto \{0,1\}$. Here, the set of problem instances is just all n-bit strings, and the oracle function queries individual bits ($A = \{0,1\}^n$ and $f_a(x) = a_x$). In this case, the quantum query complexity can be at most polynomially smaller than the deterministic classical query complexity. Indeed, Beals et al. showed [19] that the deterministic query complexity $D(g) = O(Q_2(g)^6)$ and $D(g) = O(Q_E(g)^4)$; the latter result was later improved by Midrijānis to $D(g) = O(Q_E(g)^3)$ [102]. See [78] for a good review of known lower bounds on quantum query complexity.

6.2 Oracles

Consider a classical oracle function $f_a(x) : \{0,1\}^n \mapsto \{0,1\}^m$ that takes an n-bit input to an m-bit output and is parametrised by a problem instance a. If we wish to use this function in a quantum algorithm, we need a unitary operator to play the role of the

oracle, i.e. a unitary operator that encodes f_a in some way. There are several possible encodings that have been considered in the literature, two of which are the *phase oracle*

$$P_a|x\rangle = \omega_{2^m}^{f_a(x)}|x\rangle \tag{6.2}$$

(where $\omega_k = e^{\frac{2\pi i}{k}}$) and the *bit oracle*

$$B_a|x\rangle|y\rangle = |x\rangle|f_a(x) \oplus y\rangle \tag{6.3}$$

where $|y\rangle$ is an additional m-qubit output register. The phase oracle can be obtained from the bit oracle with one query. However, obtaining the bit oracle from the phase oracle may require more queries, or even be impossible. An example is given by the function $f_a(x) = a$: encoding this function by the phase oracle results in a being returned as an unobservable global phase, so no information about a can be retrieved.

The bit oracle can be diagonalised by the use of the inverse quantum Fourier transform (QFT) operating on the $|y\rangle$ register [84]. Call the 2^m-dimensional QFT Q, where Q is defined by its action on basis states: $Q|x\rangle = \sum_k \omega_{2^m}^{kx}|k\rangle$. Then direct calculation shows that

$$(I \otimes Q)B_a(I \otimes Q^{-1})|x\rangle|y\rangle = \omega_{2^m}^{f_a(x)y}|x\rangle|y\rangle \tag{6.4}$$

There is a third, and even simpler, way of encoding an oracle function as a unitary operator: the *minimal* or *erasing* oracle [84]

$$M_a|x\rangle = |f_a(x)\rangle \tag{6.5}$$

For this operator to be unitary, it is clear that $f_a(x)$ must be a permutation.

6.3 Unstructured search

Perhaps the most basic problem in computer science is *unstructured search*. One formulation of this problem is as follows. Consider a set S containing $N = 2^n$ elements, each labelled by an n-bit binary string. S either contains one "marked" element a or no marked elements. The function $f_a(x) : \{0,1\}^n \mapsto \{0,1\}$ returns 1 if $x = a$, and 0 otherwise. The problem is to find a, or output that it does not exist, with the smallest possible number of evaluations of f_a.

On a classical computer, it is clear that N oracle queries are required to succeed in the worst case (where there is no marked element in the set). However, on a quantum computer, Grover's celebrated search algorithm [64] can be used to find the marked element with constant probability, or show that it does not exist, using only $O(\sqrt{N})$ queries.

It turns out that this is optimal: even on a quantum computer, unstructured search

requires $\Omega(\sqrt{N})$ oracle queries[1]. This lower bound was in fact shown before the development of Grover's search algorithm [21], and can be proven in a variety of ways (e.g. [7, 19]). In view of its importance, and as it is used in several places later in this thesis, we sketch a proof here. The proof is based on the "geometric" adversary arguments originally introduced by [21], with some minor changes.

Consider $N+1$ different instances of the unstructured search problem, corresponding to element number $1 \ldots N$ being marked, or there being no marked element. We will then have N different oracle operators $\{O_a\}$, each corresponding to a different element being marked, and an additional oracle operator corresponding to there being no marked element. We do not fix the way that these oracles encode the function f_a, and place only a single restriction on them: that there exists a decomposition of the overall system into a direct sum of subspaces $\mathcal{H}_0 + \bigoplus_{i=1}^{N} \mathcal{H}_i$ such that oracle O_a acts as the identity everywhere other than $\mathcal{H}_a$ (the oracle corresponding to there being no marked element must therefore be the identity operator I). The meaning of this constraint is that, given an oracle O_a, there exist N possible subspaces in which amplitude can be "invested", but we should only get an oracle response if we guess the right subspace $\mathcal{H}_a$. It is easy to see that both the phase and bit oracles satisfy this constraint.

The basic idea of the proof is simple: in order to distinguish the case where there is a marked element in the set from the case where there is not, the state produced by running the algorithm with any oracle O_a must be "far" from the state corresponding to use of the identity oracle operator. We will require the following lemma.

Lemma 6.3.1. (Bernstein and Vazirani [23])
Given a state $|\psi_?\rangle$ promised to be either $|\psi_1\rangle$ or $|\psi_2\rangle$, there exists a measurement that determines which is the case with error probability at most ϵ if and only if $|\langle\psi_1|\psi_2\rangle|^2 \leq 4\epsilon(1-\epsilon)$.

Call $|\psi_{i,t}\rangle$ the state of the system after applying t oracle queries interspersed with t arbitary unitaries U_t, where the marked element is stored in position i, and let $|\phi_t\rangle$ be the state at time t if there is no marked element. Assume that the algorithm finishes at time $t = T$ and fails with probability ϵ. Then, if the marked element is stored at position i, the final state of the algorithm is

$$|\psi_{i,T}\rangle = U_T\, O_i\, U_{T-1} \cdots O_i\, U_1 |\psi_0\rangle \tag{6.6}$$

We will compare this with the result of applying the unitaries $\{U_t\}$ without the oracle queries, corresponding to there being no marked element:

$$|\phi_T\rangle = U_T\, U_{T-1} \cdots U_1 |\psi_0\rangle \tag{6.7}$$

[1] In fact, Grover's algorithm is exactly optimal and cannot be improved by even one query [134].

Using the triangle inequality, we have

$$\||\psi_{i,T}\rangle - |\phi_T\rangle\| \leq \sum_{t=1}^{T} \||\psi_{i,t}\rangle - |\phi_t\rangle\| \tag{6.8}$$

i.e. the total distance between $|\psi_{i,T}\rangle$ and $|\phi_T\rangle$ cannot be more than the sum of the distances between them at each step of the algorithm. Then, via a Cauchy-Schwarz inequality, we have

$$\||\psi_{i,T}\rangle - |\phi_T\rangle\|^2 \leq T \sum_{t=1}^{T} \||\psi_{i,t}\rangle - |\phi_t\rangle\|^2 \tag{6.9}$$

so we can lower bound the average distance as

$$\frac{1}{N}\sum_{i=1}^{N} \||\psi_{i,T}\rangle - |\phi_T\rangle\|^2 \leq \frac{T}{N}\sum_{i=1}^{N}\sum_{t=1}^{T} \||\psi_{i,t}\rangle - |\phi_t\rangle\|^2 \tag{6.10}$$

We can now find a bound on each side of this inequality. As the algorithm fails with probability $\leq \epsilon$, we can use Lemma 6.3.1 to bound the left-hand side as follows.

$$\begin{aligned} \frac{1}{N}\sum_{i=1}^{N} \||\psi_{i,T}\rangle - |\phi_T\rangle\|^2 &= \frac{1}{N}\sum_{i=1}^{N} 2 - 2\,\mathrm{Re}(\langle\psi_{i,T}|\phi_T\rangle) & (6.11)\\ &\geq 2\left(1 - 2\sqrt{\epsilon(1-\epsilon)}\right) & (6.12) \end{aligned}$$

In order to bound the right-hand side, we define a "deviation" vector $|D_{i,t}\rangle$ that, intuitively, measures the effect of an oracle call at time t:

$$|D_{i,t}\rangle = O_i|\phi_t\rangle - |\phi_t\rangle \tag{6.13}$$

Now we have

$$\begin{aligned} |\psi_{i,1}\rangle &= |\phi_1\rangle + |D_{i,1}\rangle \\ |\psi_{i,2}\rangle &= |\phi_2\rangle + |D_{i,2}\rangle + O_iU_2|D_{i,1}\rangle \\ &\vdots \\ |\psi_{i,t}\rangle &= |\phi_t\rangle + |D_{i,t}\rangle + O_iU_t|D_{i,t-1}\rangle + \ldots + O_iU_t\cdots O_iU_2|D_{i,1}\rangle \end{aligned}$$

Thus

$$\||\psi_{i,t}\rangle - |\phi_t\rangle\| \leq \||D_{i,t}\rangle\| + \||D_{i,t-1}\rangle\| + \ldots + \||D_{i,1}\rangle\| \leq t \max_{|\phi\rangle} \|O_i|\phi\rangle - |\phi\rangle\| \tag{6.14}$$

so we have the following inequality:

$$\frac{T}{N}\sum_{i=1}^{N}\sum_{t=1}^{T}\||\psi_{i,t}\rangle - |\phi_t\rangle\|^2 \leq \frac{T^2}{N}\max_{|\phi\rangle}\sum_{i=1}^{N}\|O_i|\phi\rangle - |\phi\rangle\|^2 \tag{6.15}$$

Now we can decompose $|\phi\rangle$ as $|\phi\rangle = |\phi^{\perp}\rangle + \sum_{i=1}^{N}|\phi_i\rangle$, where $|\phi_i\rangle$ is the (unnormalised) projection onto the subspace $\mathcal{H}_i$ on which oracle O_i acts, and $|\phi^{\perp}\rangle$ is the projection onto the remainder. We then have

$$\sum_{i=1}^{N}\|O_i|\phi\rangle - |\phi\rangle\|^2 = \sum_{i=1}^{N}\|O_i|\phi_i\rangle - |\phi_i\rangle\|^2 \tag{6.16}$$

which will clearly be maximised when $O_i|\phi_i\rangle = -|\phi_i\rangle$, giving

$$\frac{T}{N}\sum_{i=1}^{N}\sum_{t=1}^{T}\||\psi_{i,t}\rangle - |\phi_t\rangle\|^2 \leq \frac{4T^2}{N}\sum_{i=1}^{N}\||\phi_i\rangle\|^2 \leq \frac{4T^2}{N} \tag{6.17}$$

which immediately gives the overall inequality

$$T \geq \sqrt{\frac{N\left(1 - 2\sqrt{\epsilon(1-\epsilon)}\right)}{2}} \tag{6.18}$$

Setting $\epsilon = 1/3$, we have proven the well-known $\Omega(\sqrt{N})$ bound on unstructured search. Note that we did not rely on knowing anything about any additional workspace used by the algorithm, and put very weak restrictions on the way that the oracle was encoded.

6.4 Boolean satisfiability

Boolean satisfiability (SAT) is a fundamental NP-complete problem which, aside from its central theoretical role, finds direct applications in fields ranging from computer vision to hardware design [65]. The problem can be defined as follows: given a Boolean expression E in conjunctive normal form (CNF), find an assignment to the variables in E such that E evaluates to 1. For example, $(a \vee b) \wedge (a \vee \neg b) \wedge (\neg a \vee b)$ has a satisfying assignment $a = b = 1$. n will denote the number of distinct literals in E, m the number of clauses (disjunctions) in E, and k the maximum number of literals in each clause. The k-SAT problem results from fixing k to be a constant; even 3-SAT is still NP-complete (although 2-SAT is in P [110]).

There are numerous classical algorithms for k-SAT that achieve improvements on the trivial algorithm of trying all potential solutions (but still have exponential time complexity), an example being Schöning's $O((4/3)^n)$ randomised algorithm for 3-SAT [117]. However, in the case of the general SAT problem, no classical algorithm is known that achieves a time complexity of better than the trivial $O(2^n)$, although algorithms

exist that achieve non-trivial complexities in terms of m [71].

Some proposed quantum algorithms to solve SAT [55, 73] use the following, quite general, oracle model. The oracle is queried with an assignment to the variables in the expression, and returns the number of unsatisfied clauses in the expression. The goal, of course, is to find an assignment for which the oracle returns 0, or to show that none exists. In this section, we reduce unstructured search on 2^n elements to solving an instance of SAT using this oracle, and thus show a $\Omega(2^{n/2})$ lower bound on SAT's query complexity.

This implies that quantum algorithms for SAT will need to use a more powerful oracle model to achieve non-trivial speed-ups over classical algorithms, in contrast to the result of van Dam et al. [47] that even the *classical* query complexity of the 3-SAT problem using this oracle is only $O(n^3)$.

6.4.1 Lower bound on query complexity

Consider a set $\{E_i\}$, indexed by $0 \leq i < 2^n$, of CNF expressions over the variables $x_1, \ldots, x_n$. We will choose these expressions so each has a different and unique satisfying assignment (the binary representation of i), and all non-satisfying assignments fail to satisfy exactly 1 clause. This can be done using no more than n clauses as follows.

Say i has binary representation $(i_1, \ldots, i_n)$. Then define clause k of E_i as the disjunction of k variables x_1 to x_k, where variables x_1 through x_{k-1} are negated if and only if the corresponding bit in i is 1, and variable x_k is negated if and only if $i_k = 0$. For example, if $n = 3$, we have $E_1 = \neg x_1 \wedge (x_1 \vee \neg x_2) \wedge (x_1 \vee x_2 \vee x_3)$. Thus the first clause of E_i disallows half the possible assignments to the variables, the second clause disallows half of the remaining assignments, and so on; it is easy to see that $(i_1, \ldots, i_n)$ is the only satisfying assignment, and that an assignment that does not satisfy one clause will satisfy all the others.

But what we have defined is exactly the unstructured search problem on 2^n elements: on problem instance i, the oracle that returns how many clauses are unsatisfied by a given assignment will return 0 if queried with $(i_1, \ldots, i_n)$, and 1 otherwise. The previously given lower bound on unstructured search then shows that computing SAT, using this oracle, requires $\Omega(2^{n/2})$ oracle queries. Note that this technique requires the maximum number of variables in a clause k to be allowed to approach n. If we fix k, this technique will not be able to produce a unique satisfying assignment for each expression without also causing the number of conflicts in other assignments to increase.

6.5 Oracle identification with a single query

Consider the following problem. We are given oracular access to a function $f_a(x) : \{0,1\}^n \mapsto \{0,1\}^m$ picked from a known set of N functions S. Our goal is to identify the value of a using the minimum number of calls to the oracle. This problem clearly fits into the query complexity framework, and is known as the *oracle identification problem* (OIP).

It turns out that a number of fundamental problems can be expressed in this form. A straightforward example of such a problem is unstructured search on N elements (where $f_a(x) = 1$ if $x = a$, and 0 otherwise). As discussed previously, this problem is well-known to have a lower bound of N classical queries, and $\Omega(\sqrt{N})$ quantum queries [21]. However, search problems with more structure may require considerably fewer queries. Here, we will consider conditions that determine whether the oracle can be identified with a *single* query.

In the classical case, this will be possible if and only if $f_a(x)$ is one-to-one for some choice of x, thus returning all the information about a immediately. In the quantum case, the situation is more interesting. It is clear that the oracles will be perfectly distinguishable if and only if there exists a state $|\psi\rangle$ such that, for some unitary operator U_a encoding the function $f_a(x)$, the states $\{U_a|\psi\rangle\}$ are orthogonal for all a. We start by obtaining necessary and sufficient conditions on certain sets of oracle unitaries for them to allow a to be determined with certainty using one query.

We then specialise to problems where the oracle functions are encoded by the bit and phase oracles defined in Section 6.2, obtain some necessary and sufficient conditions for exact single-query oracle identification, and then restrict still further, to the subclass of problems where $N = 2^n$, the oracle functions are Boolean and encoded by the phase oracle, and a can be identified with a single query. In this situation, the best known separation from classical query complexity is obtained by the Bernstein-Vazirani parity problem [23], for which a classical computer can find a with n queries. It is natural to ask whether a significantly better separation than this can be produced. We answer this question strongly in the negative by showing that every oracle identification problem of this type that can be solved with one quantum query can be solved with at most $\approx 1.71n$ classical queries.

Servedio and Gortler obtained [118] a much broader result than this, showing that any oracle identification problem using q quantum queries needs only $O(nq^3)$ classical queries[2]. The result here has a smaller constant and the proof is very different, being based on structural properties of Hadamard matrices.

Finally, we move to considering a scenario where we are allowed a constant probability of error to identify the oracle. In the classical world, this ability is almost useless.

[2]As noted by Servedio and Gortler, this does not imply that there can be no exponential *time* separation between quantum and classical computation for such problems.

However, it will turn out that quantum computers can make better use of it: when $f_a(x)$ is a Boolean function encoded by the phase oracle, and if a uniform probability distribution is fixed on the choice of oracle, we will use the results of Chapter 5 to show that *almost all* sets of oracles that are not too large (e.g. $N = 2^n$) allow a to be probabilistically determined with one query.

6.5.1 Previous work

The oracle identification problem lends itself to several natural interpretations, some of which have been considered by other workers. It appears that study of the problem, under the name of "binary identification", was initiated by Garey [59] in 1972. Ambainis et al. [10] were the first to study the general OIP in the context of quantum computation, and obtained upper bounds on query complexity for a range of oracles. These results were extended in [11], which also considered the case of a "noisy" oracle that sometimes gives a wrong answer. Servedio and Gortler [118], and Hunziker et al. [80], have considered the problem from the perspective of computational learning theory.

We now review two known examples of quantum oracle identification by a single query. In both cases, we have $N = 2^n$, so each oracle can be identified with an n-bit string a. The oracle function $f_a(x)$ is encoded by the phase oracle, i.e. as a unitary operator U_a, where $U_a|x\rangle = \omega^{f_a(x)}|x\rangle$, where ω is some root of unity (see Section 6.2). In both cases, the search algorithm simply consists of applying this oracle to the uniform superposition $|\psi\rangle = \frac{1}{\sqrt{2^n}} \sum_i |i\rangle$, and noting that the possible outcomes for different strings are all orthogonal, and hence can be distinguished immediately.

Bernstein-Vazirani parity problem

This straightforward example illustrates the power of quantum search. The scenario is that each oracle performs the parity function $f_a(x) = a \cdot x \pmod 2$, where $a \cdot x = \sum_k a_k x_k$, x_k denoting the k'th bit of x. A classical algorithm requires n queries to determine a with certainty (see Section 6.9 for details). However, Bernstein and Vazirani showed [23] that a quantum algorithm can determine a with only a single query to U_a, where $U_a|x\rangle = (-1)^{a \cdot x}|x\rangle$. This can be shown by considering the inner product between the outcomes for two oracles a, b:

$$\langle\psi|U_b^\dagger U_a|\psi\rangle = \frac{1}{2^n}\sum_i (-1)^{i\cdot(a\oplus b)} = \frac{1}{2^n}\sum_i (-1)^{\sum_k i_k(a_k \oplus b_k)} \tag{6.19}$$

This is easily seen to be zero for all $a \neq b$.

Hamming distance function

The following set of oracles can also be distinguished with a single query. Let $f_a(x) = d(a,x) \pmod 4$, where $d(a,x)$ represents the Hamming distance between a and x – i.e. the number of bits where a and x differ. Hunziker and Meyer [79] (and independently Hogg [72]) have shown that this function enables a to be found with a single query to U_a. This oracle can be used to solve the 1-SAT problem[3] in a single query [72], whereas classically it requires n queries.

The algorithm works as follows. Consider the matrix $B = \frac{1}{\sqrt{2}}\left(\begin{smallmatrix} 1 & i \\ i & 1 \end{smallmatrix}\right)$. We have $B|a_1\rangle = \frac{1}{\sqrt{2}}\sum_{x=0}^{1} i^{d(a_1,x)}|x\rangle$; that is, for some single-bit value a_1, B takes $|a_1\rangle$ to a superposition created from the Hamming distance between a_1 and the two different possible values of x (call this superposition $|\xi_1\rangle$). So, if we can produce the state $|\xi_1\rangle$, then we can obtain a_1 by applying B^{-1}. Further, if we can produce the state $\bigotimes_k |\xi_k\rangle$, then we can obtain the full value of a by applying $(B^{-1})^{\otimes k}$. It turns out that, if f_a is encoded by the phase oracle as a unitary U_a, where $U_a|x\rangle = i^{f_a(x)}|x\rangle$, we have

$$U_a\left(\frac{1}{\sqrt{2^n}}\sum_k |k\rangle\right) = \bigotimes_k |\xi_k\rangle \tag{6.20}$$

The required value a can therefore be obtained with one query.

6.6 Conditions on single-query oracle identification

We now proceed to showing conditions that restrict when we can perform exact oracle identification, first noting that this scenario is just a special case of the unitary operator discrimination problem of Section 5.2.4.

6.6.1 Preliminaries

Consider a set S of N commuting unitary operators $\{U_i\}$ (where $1 \leq i \leq N$), operating on an n qubit space. The $2^n \times N$ matrix Λ whose entries consist of the eigenvalues of the elements of S – i.e. matrix element Λ_{ij} is the i'th eigenvalue of U_j – will be called the *eigenvalue matrix* of S. As every element in S is unitary, every entry in Λ will have unit modulus.

Our goal is to show conditions on the existence of a state $|\psi\rangle$ such that for all $i \neq j$, $\langle\psi|U_j^\dagger U_i|\psi\rangle = 0$. If this is the case, we say that S *allows single-query oracle identification*. As S is simultaneously diagonalisable, we will assume that all of its elements have been diagonalised, as this will not affect the existence of such a $|\psi\rangle$. This can be seen by the following argument: $\langle\psi|U_j^\dagger U_i|\psi\rangle = 0 \Rightarrow \langle\psi|(U_j^\dagger D^\dagger)(DU_i)|\psi\rangle = 0 \Rightarrow \langle\psi'|(DU_j^\dagger D^\dagger)(DU_i D^\dagger)|\psi'\rangle = 0$ for some $|\psi'\rangle$ and any unitary D.

[3] See Section 6.4 for a definition of this problem.

Also, it is worth noting that we only need to consider the case where the dimensions of the state $|\psi\rangle$ and the operators in S are the same: for example, we do not need to consider a setting where each operator in S only acts on part of an entangled state. To see this, consider two arbitrary n-dimensional diagonal unitaries (say A with diagonal elements (a_i) and B with diagonal elements (b_i)). We will extend these by taking the tensor product with an m-dimensional identity operator and calculate the inner product we get when we attempt to distinguish the operators using an arbitrary input state $|\psi\rangle$.

$$\langle\psi|(A^\dagger \otimes I_m)(B \otimes I_m)|\psi\rangle = \sum_{j=1}^{m}\sum_{i=1}^{n} \psi_{ij}^* a_i^* b_i \psi_{ij} = \sum_{i=1}^{n} a_i^* b_i \left(\sum_{j=1}^{m} |\psi_{ij}|^2\right) \tag{6.21}$$

But this is the same as using a state $|\psi'\rangle$ with $\psi'_i = \sum_{j=1}^{m} |\psi_{ij}|^2$ to distinguish the operators A and B directly, so there was no need to add an ancilla. On the other hand, note that if the unitaries in S do not commute, it may be necessary to add an ancilla to distinguish them. For example, the operators $\{\left(\begin{smallmatrix}1&0\\0&1\end{smallmatrix}\right), \left(\begin{smallmatrix}0&1\\1&0\end{smallmatrix}\right), \left(\begin{smallmatrix}1&0\\0&-1\end{smallmatrix}\right)\}$ cannot be distinguished with certainty without an ancilla, but by the results in Section 5.2.4, they can be distinguished using a 2-dimensional ancilla and a maximally entangled state as input.

6.6.2 Single-query oracle identification

The first result in this section is the following.

Theorem 6.6.1. *S allows single-query oracle identification if all the columns of Λ are orthogonal. If $N \geq 2^n$, then this is a necessary condition.*

Proof. We will first show that, if the columns of Λ are orthogonal, S allows single-query oracle identification. Consider the result of applying an arbitrary unitary $U_j \in S$ to the uniform superposition $|+\rangle = \frac{1}{\sqrt{2^n}} \sum_i |i\rangle$. Each result will be equal to the (normalised) j'th column of Λ. If all the columns of Λ are orthogonal, all the results will be orthogonal too, and hence will be distinguishable immediately.

The converse (for $N \geq 2^n$) will be proven as follows. First, it is immediate that for $N > 2^n$ the task is impossible (as we would require N orthogonal states in a smaller than N-dimensional space), so restrict to the case $N = 2^n$. Choose an arbitrary n-qubit input state $|x\rangle = \sum_i x_i |i\rangle$. Define a matrix $M = (U_1|x\rangle \; U_2|x\rangle \; \ldots)$, each column of which is the result of applying a different oracle unitary to $|x\rangle$. As $N = 2^n$, M will be a square matrix. As each oracle is diagonal, we have $U_k|x\rangle = \sum_i \Lambda_{ik} x_i |i\rangle$. If the results of applying each oracle can be distinguished perfectly, it must be possible to pick an $|x\rangle$ such that the columns of M are orthonormal. This implies that the rows of M are

also all orthonormal. It follows that, for all $i \neq j$,

$$\sum_k (x_i \Lambda_{ik}) \left(x_j^* \Lambda_{jk}^*\right) = 0 \Rightarrow x_i x_j^* \left(\sum_k \Lambda_{ik} \Lambda_{jk}^*\right) = 0 \tag{6.22}$$

Thus, for all $i \neq j$, either $x_i = 0$, or $x_j = 0$, or the sum over k is zero. The former two options are not possible, because then the rank of M would be reduced, and the columns could not all be linearly independent (let alone orthogonal). Therefore, the sum must be zero for all $i \neq j$. But this sum is precisely the pairwise inner product of rows i and j of Λ. As Λ is square, this implies that all the columns of Λ must also be orthogonal. □

Note that, unfortunately, the "only if" direction of this result does not hold for $N < 2^n$, where it is easy to produce counterexamples.

6.7 Search with a Boolean oracle function

The simplest possible scenario – one which encompasses the Bernstein-Vazirani parity problem – is where we use a Boolean oracle function, i.e. $f_a(x) : \{0,1\}^n \mapsto \{0,1\}$. In the case where such a function is encoded by the phase oracle, the corresponding eigenvalue matrix Λ will be defined by $\Lambda_{ij} = (-1)^{f_j(i)}$. Given the fact that two vectors whose every entry is ± 1 are orthogonal if and only if they differ in exactly half of their components, by Theorem 6.6.1 we have

Theorem 6.7.1. *A set of N oracle functions $f_a(x) : \{0,1\}^n \mapsto \{0,1\}$, encoded by the phase oracle, can be distinguished with one query if there exists a subset of the inputs such that, for all $a \neq b$, $f_a(x) = f_b(x)$ for precisely half of the inputs x in that subset. If $N \geq 2^n$ then this is a necessary condition.*

In the case where $N = 2^n$, the eigenvalue matrix Λ is a *Hadamard matrix* [66]. Hunziker et al. [80] have considered this specific problem from the perspective of classical computational learning theory, calling it *concept learning* from a single membership query. Their paper shows the "if" direction of this theorem, which still holds in the case where $N < 2^n$. A version of this direction of the theorem can also be applied to the bit oracle. In the case where we use the bit oracle (which we express using the Fourier basis for the output register, see Section 6.2), the eigenvalue matrix Λ will be the same, but augmented with 2^n rows that contain only 1, corresponding to putting 0 in the $|y\rangle$ register. All but one of these rows can be removed without affecting the existence of a submatrix of Λ with orthogonal columns. This leads to the following theorem:

Theorem 6.7.2. *A set of N oracle functions $f_a(x) : \{0,1\}^n \mapsto \{0,1\}$, encoded by the bit oracle, can be distinguished with one query if there exists a k-subset of the inputs*

such that either: (a) for all $a \neq b$, $f_a(x)$ differs from $f_b(x)$ on precisely $k/2$ of the inputs; or (b) for all $a \neq b$, $f_a(x)$ differs from $f_b(x)$ on precisely $(k+1)/2$ of the inputs.

6.8 Search with a higher-dimensional oracle function

An extension of the results in the previous section is to classical oracle functions that return more than one bit. That is, functions $\{0,1\}^n \mapsto \{0,1\}^m$, where $m > 1$. We now give a simple sufficiency condition for oracle identification with the phase oracle.

Theorem 6.8.1. *Consider a set of functions $f_a(x) : \{0,1\}^n \mapsto \{0,1\}^m$ encoded by the phase oracle and an arbitrary subset of the input T. Set $c_{ab}(x) = f_a(x) - f_b(x)$ (where $f_a(x)$ and $f_b(x)$ are considered as integers mod 2^m), and set $c_{ab}^T(k) = |\{x \in T : c_{ab}(x) = k\}|$. Then a can be determined with one query if there exists a T such that, for all $a \neq b$ and for all k, $c_{ab}^T(k) = c_{ab}^T((2^{m-1} + k) \bmod 2^m)$.*

Proof. Use as input state the uniform superposition over the inputs in T, and consider the inner product between the result of applying the oracle corresponding to f_a to this state, and the result of applying f_b. Every pair of outputs $f_a(x)$, $f_b(x)$ of the functions that differ by $c_{ab}(x)$ correspond to a contribution of $\omega_{2^m}^{c_{ab}(x)}$ to the inner product. As $\omega_{2^m}^{c_{ab}(x)} + \omega_{2^m}^{2^{m-1}+c_{ab}(x)} = 0$, if the outputs can be paired up as stated in the theorem, then the inner product between each two distinct columns will be zero. □

6.9 Classical vs. quantum single-query oracle identification

It turns out that the parity and Hamming distance classical oracle functions in Section 6.5.1 can be written in a very similar way. That is, the parity of a and x is $\sum_i a_i \wedge x_i$ (where a_i denotes the i'th bit of a), and the Hamming distance between a and x is $\sum_i a_i \oplus x_i$. Thus, these functions depend only on the individual bits of their inputs, and not on combinations of these bits.

It is notable that, for both of these functions, there exists a classical algorithm to find a using n queries. For the parity function, the algorithm consists of querying the oracle with every possible input of Hamming weight 1; each query reveals the value of one bit of a. For the Hamming distance function, the algorithm is the following. First, query the oracle with the zero string: call the result z. Then query with the string whose first bit is 1, with all other bits set to 0. The result of this query will either be $z - 1$ (if $a_1 = 1$), or $z + 1$ (if $a_1 = 0$). Then set the first bit to a_1, the second bit to 1, and all other bits 0, and query once more to reveal the second bit. Repeating this process, the same principle will apply: each query reveals one more bit of a. This would result in $n + 1$ queries, but the last query can be avoided because, once all bits but one are known, the last bit can be deduced without calling the oracle.

This leads to the following natural question: do any Boolean oracle functions exist which can provide a larger quantum/classical query complexity separation? Servedio and Gortler [118] have answered this question in the general case by showing that, for every oracle identification problem requiring q quantum queries, there exists a classical algorithm using $O(q^3 \log N)$ queries. Here we tighten this bound by showing that, when $q = 1$ and $N = 2^n$, the constant in this expression is extremely small.

Theorem 6.9.1. *Any set of 2^n oracles $f_a(x) : \{0,1\}^n \mapsto \{0,1\}$ that, if encoded by the phase oracle, can be distinguished with one quantum query, can be distinguished using $\approx 1.71n$ classical queries.*

By Theorem 6.7.1, every such $f_a(x)$ can be expressed as a Hadamard matrix H, where the columns are indexed by different values for a, and the rows by different values for x. A classical oracle query consists of obtaining $y = f_a(x)$ for some choice of x, and removing all the columns c of H where $H_{xc} \neq (-1)^y$. When all columns have been removed but one, we have found a. This process will be easiest if there exists a row of H where half of the entries are 1 and half -1, because a query will always remove half of the columns. Conversely, it will be difficult if almost all of the entries in each row are 1 or -1.

Call the maximum number of "-1" entries in a row k. We can assume that $k \leq 2^{n-1}$ – i.e. at most half of the entries in each row are -1 – because, if not, we can always multiply a row by -1 (this corresponds to blindly inverting the result of an oracle call). The following lemma will show that k must be high, and thus classical queries must always be "informative" and reduce the search space efficiently.

Lemma 6.9.2. *Consider a matrix M whose entries are all ± 1, and whose columns are all orthogonal. Say M has n columns, and the maximum number of "-1"s contained in any row of M is k. Then $k \geq \frac{n-1}{3}$.*

Proof. Assume $k \leq n/2$, or the lemma is trivially true. Then consider the inner product between each two columns c and d of M. There are $\binom{n}{2}$ such combinations of different columns. We will look at how the inner product for each combination (c, d) is produced, by considering each row r in turn. (c, d) will be incremented when the two entries c_r and d_r are either both 1 or both -1, and will be decremented otherwise. With each row, therefore, at least $\binom{n-k}{2}$ combinations are incremented (there are at least $n - k$ "1"s in the row, so this is the minimum number of ways of choosing 2 entries that are both 1).

Also, at most $k(n-k)$ combinations are decremented. This can be seen by denoting the number of "-1"s in the row by l, where $l \leq k \leq n/2$. Then the number of ways of picking one 1, and one -1 is $l(n - l)$. For $l \leq n/2$, this is an increasing function, and hence will be maximised by $l = k$. We must thus have $k(n - k) \geq \binom{n-k}{2}$, or it is impossible to reduce all of the combinations which have been incremented back to 0. This immediately leads to the required bound, $k \geq \frac{n-1}{3}$. □

Therefore, each classical query must reduce the size of the search space by approximately 1/3. The total number of queries required to reduce the search space to 1 element is thus at most $\approx \frac{n}{\log(3/2)} \approx 1.71n$ and Theorem 6.9.1 is proven.

6.10 Probabilistic single-query oracle identification

6.10.1 Introduction

This section is concerned with identifying Boolean functions encoded by the phase oracle. In this scenario, the results of the previous sections may suggest that the ability of a quantum computer to perform single-query oracle identification is limited, and that the oracle functions that allow single-query oracle identification (e.g. the Bernstein-Vazirani oracle) are exceptional. However, we will now show, using the results of Chapter 5, that if we switch to a bounded-error model and fix a distribution on the oracles, then quantum computers have a generic advantage over classical computers. In particular, almost all sets of 2^n Boolean oracle functions on n bits can be distinguished by one quantum query with a constant probability of success $> 1/2$.

The formal problem specification is as follows. A set S of N Boolean functions $f_a(x) : \{0,1\}^n \mapsto \{0,1\}$ is fixed, where each function in S is picked uniformly at random from the set of n-bit Boolean functions. A function is picked uniformly at random from S. The goal is to determine, with a constant probability of success and using the minimum number of queries to the function, which function has been picked. Clearly, a classical computer cannot identify a function picked from such a set with probability $> 1/2$ in fewer than $\log N$ queries (as each query may reduce the search space by at most half). In fact, this is almost tight. Consider a set of k random classical queries to the unknown function f. The probability that any two of the set S of random functions agree on all k queries is 2^{-k}, so by the union bound they will all differ on at least one of the queries with probability $\geq 1 - \binom{N}{2}2^{-k}$. Setting $k = 3\log_2 N$ makes this success probability approach 1 for large N, showing that almost all sets of N oracles can be distinguished with $O(\log N)$ classical queries.

In their paper introducing the oracle identification problem, Ambainis et al. [10] developed (among other results) a hybrid quantum-classical algorithm for this "random oracle identification" problem. However, the upper bound they obtained in the case where $N = 2^n$ is only $O(\log N)$ queries, and thus no better than classical computation.

6.10.2 The algorithm

The quantum algorithm for identifying which random oracle function we were given is straightforward: apply the oracle function, encoded as the phase oracle (Section 6.2), once to the uniform superposition $\frac{1}{\sqrt{2^n}}\sum_i |i\rangle$ and use a measurement to attempt to distinguish between the N possible states that may be produced. We now apply the

results of Chapter 5 to determine how distinguishable these states are (see that chapter for the notation used below).

Lemma 6.10.1. *Let $\mathcal{E}$ be an ensemble of N 2^n-dimensional states corresponding to applying the phase oracle encoding of random Boolean functions to the uniform superposition (call these* random oracle states*). Then the rescaled state matrix $\sqrt{N2^n}\, S(\mathcal{E})$ defines a point picked uniformly at random on the $N2^n$-dimensional hypercube $\{-1,1\}^{N2^n}$.*

Proof. Each component of each state will be $\pm 1/\sqrt{N2^n}$, with equal probability of each. □

$\sqrt{N2^n}\, S(\mathcal{E})$ is therefore a random matrix meeting the required conditions for the Marčenko-Pastur law (Theorem 5.5.1), so we may immediately calculate a lower bound on the expected probability of success of a specific measurement (the "pretty good measurement", see Section 5.2.1) applied to this ensemble.

Lemma 6.10.2. *Let $\mathcal{E}$ be an ensemble of N 2^n-dimensional random oracle states, and set $r = N/2^n$. Then*

$$\mathbb{E}(P^{pgm}(\mathcal{E})) \geq \begin{cases} \frac{1}{r}\left(1 - \frac{1}{r}\left(1 - \frac{64}{9\pi^2}\right)\right) - O(N^{-5/48}) & \text{if } N \geq 2^n \\ 1 - r\left(1 - \frac{64}{9\pi^2}\right) - O(2^{-5n/48}) & \text{otherwise} \end{cases} \tag{6.23}$$

and in particular $\mathbb{E}(P^{pgm}(\mathcal{E})) \geq 0.720 - O(2^{-5n/48})$ when $N \leq 2^n$.

Like the sphere, the high-dimensional hypercube exhibits the concentration of measure phenomenon [92], and we can use this to show that almost all sets of oracle states can be distinguished as easily as this expected value would suggest.

Lemma 6.10.3. (Concentration of measure on the cube) [92]
Given a function $f : \{-1,1\}^d \mapsto \mathbb{R}$ defined on a d-dimensional hypercube, and a point p on the hypercube chosen uniformly at random,

$$\Pr[|f(p) - \mathbb{E}(f)| \geq \epsilon] \leq 2\exp\left(\frac{-2\epsilon^2}{d\eta^2}\right) \tag{6.24}$$

where η is the Lipschitz constant of f with respect to the Hamming distance, defined as $\eta = \sup_{x,y} |f(x) - f(y)|/d(x,y)$.

Lemma 6.10.4. *Let H be a point on the nd-dimensional hypercube written down as an $n \times d$ $\{-1,1\}$-matrix, and let $f(H) = \frac{1}{n^2 d}\|H\|_1^2$. Then the Lipschitz constant η of f satisfies $\eta \leq 4/nd$.*

Proof. See Section 5.8. □

Inserting this value of η into Lemma 6.10.3 gives

Theorem 6.10.5. *Let $\mathcal{E}$ be an ensemble of N 2^n-dimensional random oracle states. Set $p = \frac{1}{r}\left(1 - \frac{1}{r}\left(1 - \frac{64}{9\pi^2}\right)\right) - O(N^{-5/48})$ if $N \geq 2^n$, and $p = 1 - r\left(1 - \frac{64}{9\pi^2}\right) - O(2^{-5n/48})$ otherwise, where $r = N/2^n$. Then*

$$\Pr[P^{pgm}(\mathcal{E}) \leq p - \epsilon] \leq 2\exp\left(\frac{-N2^{n+1}\epsilon^2}{16}\right) \tag{6.25}$$

and we have our desired result. For example, picking $N = 2^n$, all but an exponentially small fraction of the possible sets of N oracles on n bits may be distinguished using one query with a probability bounded away from $1/2$ (in fact, to get a probability of success greater than $1/2$, we may take $N/2^n$ to be as high as ~ 1.66). A constant number of repetitions allows this probability to be boosted to be arbitrarily high.

Chapter 7

Quantum search of partially ordered sets

7.1 Introduction

Searching for an object in a set of objects that obey some structure is a fundamental task in computer science. The archetypal example of such a task is finding an integer in a sorted list containing n elements; in this case, binary search can find the marked integer in $O(\log n)$ steps. At the other extreme, any (classical) search algorithm requires $\Omega(n)$ steps to search a completely unsorted n-element list. It is of interest to find a framework for search problems that encompasses both of these structures, and interpolates between them.

One approach is to consider the task of searching a partially ordered set (*poset*). Recall that a partial order on a set S is a relation $\leq$ such that, for $a, b, c \in S$, $a \leq a$, $(a \leq b) \wedge (b \leq a) \Rightarrow a = b$, and $(a \leq b) \wedge (b \leq c) \Rightarrow a \leq c$. We define the relation $<$ in the obvious way: $(a < b) \Leftrightarrow (a \leq b) \wedge (a \neq b)$. For any two elements a, b, either $a \leq b$, $b \leq a$, or a and b are incomparable, $a \not\leq b$. We say that a set is totally ordered if none of its elements are incomparable, and unstructured if all of its elements are incomparable.

There are two natural ways to model poset search. In the first model (introduced by Linial and Saks [95], and called the *concrete* model here), we consider the partial order on S to represent constraints on the structure of an unknown *totally* ordered set, identified with the integers. That is, each element $s \in S$ stores an integer $x = S[s]$, which is returned by a query to the element s. The constraint following from the partial order on S is that if $s \leq t$ for some $s, t \in S$, then $S[s] \leq S[t]$. The goal is to find the location at which a (known) arbitrary integer a is stored, or to output that a is not stored in S, using the minimum number of queries to elements of S. We will usually assume that the integers stored in S are all distinct.

Alternatively, in the second model (introduced by Ben-Asher, Farchi and Newman

[20], and called the *abstract* model here), the goal is to search for an unknown "marked" element $a \in S$, using the minimum number of queries to an oracle, which operates in the following way. On input of an element $x \in S$, the oracle returns one of $\{<, =, \not\leq\}$. The first two possibilities are returned when $a < x$ and $a = x$, respectively, and the third is returned when either $x < a$ or x and a are incomparable.

We sometimes mention an extension of the search problem to a scenario where multiple different answers are permissible. This extension is different for the two models: in the abstract model, we consider there to be multiple marked elements in the set to be searched, with the goal being to output any of these elements. In the concrete model, the analogous scenario is allowing the possibility for the set to store duplicate integers, i.e. allowing there to exist $s, t \neq s$ such that $S[s] = S[t]$.

To sum up, in the concrete model we know what we are searching for, but not where to find it; in the abstract model, we do not know what we are searching for, but we can perform powerful queries that narrow down the search space to find it.

This chapter is concerned with quantum search of posets in both of these models, and in particular with minimising the number of queries to the set required to find the desired element. It is well-known that Grover's algorithm [64] can find the marked element in an unstructured n-element set using $O(\sqrt{n})$ quantum queries, thus gaining a quadratic advantage over classical computation, and that this reduction is optimal. However, no advantage beyond a constant factor may be achieved for quantum search of a totally ordered set [77].

We then have several questions, motivated by these two examples. Can we find interesting quantum algorithms for search of general posets? Could a reduction in queries of more than the quadratic factor given by Grover's algorithm be achieved by such an algorithm, or even an exponential reduction? And what is the structure (or otherwise) of the posets for which a quantum computer can gain an asymptotic advantage over classical computation?

7.1.1 New results

The first result in this chapter is that, in both the abstract and concrete models, quantum algorithms can achieve no more than a quadratic reduction (up to a logarithmic factor) in the number of oracle queries to find a marked element. The lower bounds in the two models seem to need different proof techniques: the bound in the abstract model follows from a reduction to the oracle identification problem, whereas we use structural properties of posets to derive the lower bound in the concrete model.

We give general upper bounds that match these lower bounds up to logarithmic factors. In the abstract model, the upper bound follows from an algorithm of Atici and Servedio [13]. In the concrete model, we give a new and almost optimal quantum algorithm that follows from Dilworth's Theorem [49] on the decomposition of posets

into ordered components.

These general results can be summarised as the following theorem.

Theorem 7.1.1. *Let S be an n-element poset, and let $D(S)$, $Q_E(S)$ and $Q_2(S)$ be the number of queries required for an exact classical, exact quantum, or bounded-error quantum (respectively) algorithm to find the marked element in S. Then, in the abstract model,*

$$\begin{aligned} D(S) &= O(Q_2(S)^2 \log n) \\ Q_2(S) &= O(\sqrt{D(S)} \log n \sqrt{\log \log n}) \end{aligned}$$

and in the concrete model,

$$\begin{aligned} D(S) &= O(Q_2(S)^2 \log n) \\ Q_E(S) &= O(\sqrt{D(S)} \log n) \end{aligned}$$

In both models, we give explicit quantum algorithms for searching specific poset structures. In the abstract model, we give a simple (and nearly optimal) algorithm for searching a class of forest-like posets. For an unstructured set, the algorithm reduces to Grover search, whereas for a totally ordered set it reduces to binary search.

In the concrete model, we give an asymptotically optimal quantum algorithm for searching posets that are derived from 2-dimensional arrays of distinct integers sorted by rows and columns. This gives rise to an optimal quantum algorithm for an apparently unrelated problem: finding the intersection of two sorted lists. Given two lists of at most n integers in increasing order, the algorithm can find an element that appears in both lists in $O(\sqrt{n})$ time, improving on a previous algorithm of Buhrman et al. [34], which achieved a time complexity of $O(\sqrt{n}c^{\log^* n})$ for some constant c.

7.1.2 Previous work

Classically, the question of searching partially ordered sets seems to have first been considered by Linial and Saks [95, 94], who characterised the query complexity of searching posets in their concrete model. They showed that this complexity depends solely (up to constant factors) on the number of ideals of the poset, where an ideal of S is a subset $T \subseteq S$ such that $(x \in T) \wedge (y < x) \Rightarrow (y \in T)$. In particular, they give lower and upper bounds on the complexity of searching for a marked element in an array sorted by rows and columns, and the d-dimensional generalisation thereof.

Ben-Asher, Farchi and Newman [20] introduced the abstract model, and gave an algorithm to find the optimal search strategy in this model for a class of tree-like posets. In this model, it is interesting to note that the problem of determining an optimal search strategy for arbitrary posets is NP-hard, whereas the same question restricted to trees

is soluble in polynomial time [35]. In fact, Onak and Parys have recently obtained an $O(n^3 \log n)$-time algorithm for finding this strategy [109], and also point out that this model is similar to a model of search in graphs, where one queries an edge and is returned the closest endpoint of that edge to the marked element. It was already known that near-optimal search strategies for almost all posets can be produced efficiently [35].

In the case of quantum search, tight upper and lower bounds on query complexity are known for search of unstructured sets [64, 28, 134]. An asymptotically tight lower bound is known for search of totally ordered sets [6, 77]. We will also make use of related results by Aaronson and Ambainis on spatial quantum search [1].

A related problem was previously studied by Yao [133]: given a partially sorted set S of integers (i.e. a set promised to be sorted consistent with some partial order), sort S. Yao obtained strong lower bounds for this problem, showing that almost no quantum advantage can be achieved.

7.2 Preliminaries

7.2.1 Quantum query algorithms

In this chapter, the measure used of the complexity of searching a poset S is usually the number of queries to S required to find the marked element, or report that none exists, rather than the time required for the search (see Section 7.6 for a brief discussion of this point). We will use the standard model of quantum query complexity introduced in Chapter 6; that is, in order to find element a in the poset, we will make calls to an oracle $f_a(x)$, where the input x identifies which poset element to query.

In the abstract model, we require an oracle $f_a(x)$ that returns something from the set $\{<, =, \not\leq\}$, according to whether the unknown marked element $a < x$, $a = x$ or $a \not\leq x$. However, it will be convenient to instead use a Boolean oracle by adding a parameter $z \in \{0, 1\}$ to give an oracle function $f_a(x, z)$, which acts as follows. $f_a(x, 0) = 1$ if $a \leq x$, and 0 otherwise. $f_a(x, 1) = 1$ if $x = a$, and 0 otherwise. It is clear that a query to $f_a(x)$ is sufficient to simulate a query to $f_a(x, z)$, and querying $f_a(x, 0)$ and $f_a(x, 1)$ is sufficient to simulate $f_a(x)$. The query complexity in the two-parameter model may thus only differ by a factor of at most 2 from the one-parameter model. The model can be extended to allowing more than one marked element in an obvious way, by parametrising the oracle with a set of marked elements A; then $f_A(x, 0) = 1$ if there exists $a \in A$ with $a \leq x$.

The concrete model is more straightforward; here, the oracle depends only on the integers stored in the set S, and an oracle query to an element x simply returns the integer stored at the element x, i.e. $S[x]$. We usually assume that, for all $x \neq y$, $S[x] \neq S[y]$.

$D(S)$ will denote the worst-case exact classical decision tree complexity of searching

for a single marked element in the poset S, and $Q_E(S)$ the equivalent quantum query complexity. $Q_2(S)$ is the quantum query complexity where we are allowed to err with probability $\leq 1/3$ (the "2" refers to 2-sided bounded error). Motivated by binary search, our notion of a poset S that allows "efficient" search is one where the marked element can be found using a number of queries that is polylogarithmic in $|S| = n$.

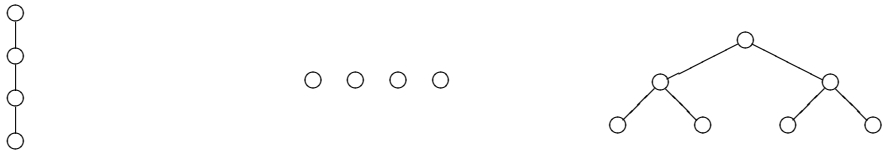
We will make frequent use of an exact variant of Grover's quantum search algorithm [64].

Theorem 7.2.1. (Exact Grover search [e.g. [28], [98]])
Let S be an unstructured set of n elements containing either one marked element, or no marked elements. Then there exists an exact quantum algorithm which outputs the marked element, or that no such element exists, using $O(\sqrt{n})$ queries to the set.

7.2.2 Posets

We will use standard terminology relating to posets. A *chain* in a poset S is a subset $T \subseteq S$, all of whose elements are comparable. Conversely, an *antichain* is a subset whose elements are all incomparable. The *height* $h(S)$ and *width* $w(S)$ of a poset S are the size of the largest chain and antichain in S, respectively. A *subset* of a poset S is a subset of the elements in S that preserves the order relations; conversely, an extension of S preserves the elements but may add new order relations. A *section* of S is a subset $T \subseteq S$ such that $(x \in T) \wedge (z \in T) \wedge (x < y < z) \Rightarrow y \in T$. A *maximal element* of S is an element $x \in S$ such that for all $y \in S$, $y \not> x$.

A poset can be represented graphically by its *Hasse diagram*. A Hasse diagram for S is an undirected graph G whose vertices are labelled by the elements of S. We say that b covers a if $b > a$ and there does not exist $c \in S$ such that $a < c < b$. For each pair of vertices a, b, if a covers b then the vertex corresponding to a in the Hasse diagram is connected to, and positioned higher than, that corresponding to b. Figure 7.1 gives the Hasse diagrams of some example posets.



Totally ordered set Unstructured set Tree-like poset

Figure 7.1: Hasse diagrams of some posets

A poset S is said to be tree-like (forest-like) if its Hasse diagram is a tree (forest) rooted at the maximal element(s) of S.

7.3 The abstract model

In this section, we consider the problem of searching posets in the model studied by Ben-Asher, Farchi and Newman [20], where a query to an element of a poset S returns information about its relationship to the unknown marked element with respect to the partial order on S.

7.3.1 Overall relationships

In this model, we can immediately relate quantum and classical search using a reduction to the oracle identification problem introduced in Chapter 6. In this problem, we are given as an oracle an unknown m-bit Boolean function f picked from a known set of functions S, and we must identify f with the minimum number of queries to the oracle (for the rest of this section, we will borrow terminology from computational learning theory [118], and refer to this as *exactly learning* f).

Servedio and Gortler have shown [118] that the quantum and classical query complexities of this task are closely related, and both depend on a parameter which we call γ^S[1], which is informally defined as the minimum fraction of the functions in S which a classical algorithm can be certain of removing from consideration with a query to f. To be precise, let S' be a subset of S, and let $S'_{a,b}$ be the subset of those functions in S' that take value b on input a. Then

$$\gamma^S = \min_{S' \subseteq S, |S'| \geq 2} \max_{a \in \{0,1\}^m} \min_{b \in \{0,1\}} \frac{|S'_{a,b}|}{|S'|} \tag{7.1}$$

The main result of [118] may be stated as:

Theorem 7.3.1. [118]
Let S be a set of Boolean functions on m bits. Then the quantum query complexity Q of exactly learning a function from S, with a bounded probability of error, obeys the following lower bounds.

$$Q = \Omega\left(\frac{1}{\sqrt{\gamma^S}}\right), Q = \Omega\left(\frac{\log|S|}{m}\right) \tag{7.2}$$

Also, the deterministic classical query complexity C of the same task obeys the following upper bound.

$$C = O\left(\frac{\log|S|}{\gamma^S}\right) \tag{7.3}$$

Quantum and classical query complexities are thus related by $C = O(\log|S|Q^2) = O(mQ^3)$.

The classical algorithm that achieves this query complexity is quite straightforward,

[1] This is Servedio and Gortler's $\hat{\gamma}^C$.

simply consisting of querying the unknown function at the input that, given an adversarial response, reduces the size of the set of remaining possible functions by the largest possible amount.

We now make a connection between the poset search problem and oracle identification. Given a poset, the oracle associated with each possible marked element a is a two-parameter Boolean function $f_a(x, z)$. Distinguishing between these functions is exactly equivalent to finding the hidden a. Thus, in order to find the marked element in an n-element poset, we need to distinguish n Boolean functions on $\lceil \log n + 1 \rceil$ bits. Theorem 7.3.1 immediately gives the following result.

Theorem 7.3.2. *Let S be an n-element poset. Then $D(S) = O(\log n\, Q_2(S)^2)$.*

A quadratic reduction in queries is thus the best that can be obtained using a quantum algorithm, up to a logarithmic factor. We now turn to upper bounds on quantum query complexity. There is a straightforward general upper bound of $O(\sqrt{n})$ oracle queries for any poset. This can be seen by noting that, if the oracle $f_a(x, z)$ is queried only with $z = 1$, the problem reduces to unstructured search, so Grover's algorithm [64] can be used.

Less trivially, Atici and Servedio [13] have given a quantum algorithm for exact learning that can be seen as an analogue of the classical algorithm mentioned in Theorem 7.3.1. This algorithm immediately applies to poset search, and moreover is efficient (runs in time polynomial in n).

Theorem 7.3.3. [13] *Let S be an n-element poset. Then*

$$Q_2(S) = O\left(\frac{\log n \log\log n}{\sqrt{\gamma^S}}\right) \tag{7.4}$$

This upper bound can actually be improved to $Q_2(S) = O\left(\log n\sqrt{\log\log n}/\sqrt{\gamma^S}\right)$. The reason is that the $O(\log\log n)$ factor in Atici and Servedio's algorithm comes from perfoming $O(\log\log n)$ rounds of classical probability amplification, which can be replaced by the use of a quantum algorithm of Buhrman et al. [29] that performs efficient amplitude amplification to small error probabilities.

In summary, it can be seen that the quantum and classical query complexities of this search problem are completely determined (up to logarithmic factors) by this parameter γ^S. However, it is unclear whether the extension to searching for multiple marked elements has a similar reduction to the oracle identification problem, and whether a suitable adaptation of Atici and Servedio's algorithm can be applied in this case.

Finally, note that one might consider a more powerful variant of search in this model, where the oracle $f_a(x)$ is extended to return $>$ if the marked element $a > x$ (so the four possible results are "$<$", "$=$", "$>$" and "incomparable"). The reduction to the oracle identification problem clearly still holds for this variant, so the results in

this section go through unchanged.

7.3.2 Search in forest-like posets

We say a poset is forest-like if every element in the poset is covered by at most one other element (an example of such a poset is shown in Figure 7.1). Classically, forest-like posets have proven to be easier to analyse; indeed, algorithms exist [20, 109] for computing the optimal classical decision tree to search these posets in polynomial time, whereas the same problem is NP-hard for general posets [35]. In this section, we present an exact quantum algorithm for searching a forest-like poset S using $O\left(\log n/\sqrt{\gamma^S}\right)$ queries, improving on the previously mentioned bounded-error $O\left(\log n\sqrt{\log\log n}/\sqrt{\gamma^S}\right)$-query algorithm [13]. Our algorithm improves on Atici and Servedio's in other ways too: firstly, it reduces to an asymptotically optimal algorithm in the case of search of unstructured and totally ordered sets; secondly, it can easily be extended to searching for multiple marked elements, with a small penalty in query complexity.

We first consider the case of a single marked element. The principles behind the algorithm that we will describe are very similar to those underlying Atici and Servedio's. Throughout the algorithm, a subset of possible places that the marked element could be is maintained. We will show that one use of Grover's algorithm over a set G of size at most $1/\gamma^S$ can be used to reduce the size of this subset by at least half, so $\log n$ repetitions suffice to find the marked element. Crucially, for forest-like posets where there is a single marked element, this use of Grover's algorithm can be made exact (Theorem 7.2.1), thus avoiding the need for some number of repetitions to achieve a suitable reduction in the error probability.

The algorithm is explicitly stated as Algorithm 1 below. It uses a subroutine `centralElement` which requires some explanation. Define the weight $wt(v)$ of an element $v \in S$ as $wt(v) = |\{x : (x \in S) \wedge (x \leq v)\}|$. Then `centralElement`$(S)$ returns the element $v \in S$ such that $wt(v)$ is maximised, given that $wt(v) \leq \lceil |S|/2 \rceil$. Such an element will clearly always exist. `siblings`(x) returns the set of elements of S that are covered by the single element that covers x.

We will now prove an upper bound on the query complexity of Algorithm 1, via a couple of preparatory lemmas.

Lemma 7.3.4. *In each iteration of the loop, the total weight of the elements in G is at least $|T|/2$.*

Proof. If x is a maximal element, then the total weight of the elements in G is clearly $|T|$, as every maximal element is added. If x is covered by an element p, then the total weight of the elements in G will be $wt(p) - 1$. But $wt(p) > \lceil |T|/2 \rceil$ (as otherwise p would have been returned by `centralElement` rather than x), so we are done. □

Lemma 7.3.5. *In each iteration of the loop, $|G| \leq 1/\gamma^S$.*

Algorithm 1 Search algorithm for forest-like posets

Input: Forest-like poset S containing n elements
Output: Marked element, or "not found"

```
T ← S;
while |T| > 1 do
  x ← centralElement(T);
  if x is a maximal element of T then
    G = {y : y is a maximal element of T};
  else
    G = {y : y ∈ siblings(x)};
  end if
  y ← result of exact Grover search on G;
  if result is "not found" then
    T ← T \ {z : ∃y′ ∈ G, z ≤ y′};
  else
    T ← {z : z ≤ y};
  end if
end while
if |T|=1 then
  return single element in T;
else
  return not found;
end if
```

Proof. We will show that $\gamma^G = 1/|G|$, implying $\gamma^S \leq 1/|G|$. Restrict the marked element to being an element of G. Then an algorithm can only remove elements of G from consideration by querying within G. This is because, if x is not a maximal element of T, all the members of G are covered by a single element p, so the only queries that can allow us to reject members of G are queries to members of G. Alternatively, if x is a maximal element of T, then it is easy to see that x is actually also a maximal element of S. So G will contain all the maximal elements of S, and again the only queries that can allow us to reject members of G are queries to members of G. □

Theorem 7.3.6. *Algorithm 1 finds the marked element in a forest-like n-element poset S, or outputs that no such element exists, with certainty using at most $O\left(\log n/\sqrt{\gamma^S}\right)$ queries to S.*

Proof. It is immediate that the algorithm is correct, as each iteration of the loop is guaranteed to remove at least one element from T. It remains to prove an upper bound on its query complexity. If the marked element a is in the set T at all, we are guaranteed that either $a \leq x$ for either exactly one element $x \in G$, or for no elements in G. The Grover search step will thus either reduce the search space to the elements $\{z\}$ of T for which $z \leq x$, or will remove all the elements $z \in T$ that are less than any element in G from consideration. Each element of G has weight at most $\lceil |T|/2 \rceil$, and by Lemma 7.3.4, their total weight is at least $|T|/2$. So each iteration of the loop will reduce the

size of T by at least about half. By Lemma 7.3.5, each Grover search uses at most $O(1/\sqrt{\gamma^S})$ queries, so the theorem is proven. □

In some cases, Algorithm 1 may do better than this upper bound suggests. One such example is searching a completely unstructured set (in which case the algorithm reduces to standard unstructured search, and thus achieves an $O(\sqrt{n}) = O\left(1/\sqrt{\gamma^S}\right)$ query complexity). As another example, it is easy to convince oneself that Algorithm 1 finds the marked element in a poset whose Hasse diagram is a complete k-ary tree with l levels using $O(\sqrt{k}l)$ queries, rather than the $O(\sqrt{k}l \log k)$ queries guaranteed by Theorem 7.3.6.

Finally, note that the extension to searching for an unknown number of marked elements is straightforward: in this case, the exact Grover search step is replaced by picking an element y from G uniformly at random. If there exists a marked element a such that $a \leq y'$ for some element $y' \in G$, then the probability that $y = y'$ is at least $1/\sqrt{\gamma^S}$. We need to boost this success probability to $\Omega(1 - 1/\log n)$ in order for the success probability after $O(\log n)$ recursions to be $\Omega(1)$. By a result of Buhrman et al. [29] on amplification of classical probabilistic algorithms with one-sided error, this can be achieved using $O(\sqrt{\log \log n}/\sqrt{\gamma^S})$ iterations of picking $y \in G$ uniformly at random, giving an overall complexity of $O\left(\log n \sqrt{\log \log n}/\sqrt{\gamma^S}\right)$.

7.4 The concrete model

In this section, we consider the problem of poset search in the model studied by Linial and Saks [95], where the poset is thought of as storing partially sorted integers (or elements from any other totally ordered set), and querying an element of the poset returns the integer stored at that element. Note that we redefine $D(S)$, $Q_E(S)$ and $Q_2(S)$ appropriately.

7.4.1 Overall relationships

This model appears more complex to analyse, as the complexity of the search problem now depends not only on the structure of the poset being searched, but also on the integers that are stored in that poset. Also, the classical analysis of Linial and Saks [95] relies on certain properties of classical algorithms for poset search that quantum algorithms seem not to share. For example, at the end of a correct classical algorithm which searched unsuccessfully for the element a in S, every element $x \in S$ must have been classified according to whether $x < a$, $x = a$ or $x > a$. Quantum algorithms appear not to have this property.

However, we can develop a quantum lower bound that is similar to a known classical lower bound based on the size of the largest "unsorted" subset of S, namely the size of the largest antichain, $w(S)$. It turns out that finding an element in such a subset

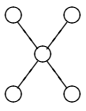
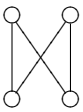
reduces to an unstructured search problem. We begin with a lemma whose classical part was shown by Linial and Saks [95] with a different proof.

Lemma 7.4.1. *Let S be a poset and let T be a section of S. Then $D(S) \geq D(T)$, $Q_E(S) \geq Q_E(T)$ and $Q_2(S) \geq Q_2(T)$.*

Proof. First, note that S can be partitioned into three disjoint subsets (or *layers*): the set T; an "upper" set U where for all $u \in U$, there is no $t \in T$ such that $u \leq t$; and a "lower" set V where for all $v \in V$, there is no $t \in T \cup U$ such that $t \leq v$. Assume S has n elements, identified with the integers. Let V store the integers $\{1, \ldots, |V|\}$ in some manner consistent with its partial order, and similarly let U store the integers $\{|V| + |T| + 1, \ldots, n\}$. By the definition of the partitioning of S, T can store every permutation of the integers $\{|V|+1, \ldots, |V|+|T|\}$ that is consistent with its own partial order, independently of the integers stored in the remainder of S.

Now consider an adversarial strategy where the marked element is guaranteed to be in the set $\{|V|+1, \ldots, |V|+|T|\}$, and thus is stored in T. Any query to elements in U or V will then give no information about the position of the marked element within T, so any classical or quantum algorithm can restrict itself to making queries to elements in T. But any classical [exact quantum, bounded-error quantum] algorithm to find a marked element in T that only makes queries to elements in T must use $D(T)$ [$Q_E(T)$, $Q_2(T)$] queries. □

Note that this property does not hold for arbitrary subsets of posets [95]: for example, the following posets S, $T \subset S$ have $D(S) = 3$ but $D(T) = 4$. The theorem does not hold at all in the abstract model of poset search discussed in the previous section.

$S =$  , $T =$ 

Lemma 7.4.2. *Let S be an n-element unstructured poset. Then $D(S) = n$ and $Q_2(S) = \Omega(\sqrt{n})$.*

Proof. Let S store an arbitrary permutation π of the integers $\{1, \ldots, n\}$, and let the marked element be $a = \pi(1)$. The classical lower bound is obvious [95] (as the only information obtained from a query to an element $x \in S$ is whether $a = x$ or $a \neq x$, every element in S may need to be queried in the worst case). In the quantum case, the lower bound of Ambainis on inverting a permutation [7] may be used to show that any quantum algorithm to find a requires $\Omega(\sqrt{n})$ queries. □

Theorem 7.4.3. *Let S be an n-element poset. Then $D(S) = \Omega(w(S))$ and $Q_2(S) = \Omega(\sqrt{w(S)})$. Also, $Q_2(S) = \Omega(\log n)$.*

Proof. Let T be the largest antichain in S. T is unstructured, T is a section of S and $|T| = w(S)$. The first part of the theorem follows immediately from Lemma 7.4.1 and Lemma 7.4.2. For the second part, note that any quantum algorithm to find a marked element in S could also be used to find a marked element in a totally ordered set of n elements. The lower bound then follows from the lower bound of Ambainis [6] (improved by Høyer, Neerbek, and Shi [77]) on quantum search of an ordered list. □

We now consider the question of upper bounds. It turns out that, up to a logarithmic factor, the width $w(S)$ *completely* characterises the classical and quantum query complexities of searching in this model. To show this, we will need the following powerful combinatorial result, which says something about the decomposition of a poset into chains.

Theorem 7.4.4. (Dilworth's Theorem [49])
Let S be an n-element poset with $w(S) = k$. Then S is the union of k disjoint chains.

In fact, such a decomposition can be found in time $O(n^3)$ [25].

Lemma 7.4.5. *Let S be a poset. Then we have $D(S) = O(w(S) \log h(S))$ and $Q_E(S) = O(\sqrt{w(S)} \log h(S))$.*

Proof. Decompose S into a set C containing $w(S)$ disjoint chains, each of which contains at most $h(S)$ elements. The classical algorithm proceeds by searching each chain in C in turn, using binary search. The total number of queries required is therefore $O(w(S) \log h(S))$.

In the quantum case, our algorithm will nest an exact binary search algorithm within the exact variant of Grover's search algorithm. We produce an oracle P_a which, when given the identifier of a chain in C as input, returns whether the desired element a is contained within that chain; each call to P_a clearly requires at most $O(\log h(S))$ queries to the set. As the chains are disjoint, we are guaranteed that P_a will return 1 on only one input. The exact variant of Grover's algorithm therefore requires (see Theorem 7.2.1) $O(\sqrt{w(S)})$ queries to P_a to determine which chain (if any) contains a. A final $O(\log h(S))$ queries are used to find a within that chain, for an overall query complexity of $O(\sqrt{w(S)} \log h(S))$. □

If the binary search parts of this algorithm are replaced by the use of a quantum ordered search algorithm (e.g. [40]), the query complexity can be improved by a constant factor. Note that this algorithm actually also works in the abstract model of poset search, thus showing that, as one might expect, search in the abstract model is always at least as easy as in the concrete model (up to the $\log h(S)$ factor). Furthermore, note that an extension to search where a given integer may be stored at multiple positions in the poset is immediate: the Grover search steps are replaced by search

for an unknown number of marked items [28] to give an $O(\sqrt{w(S)} \log h(S))$-query bounded-error quantum algorithm.

We can now show that the classical and quantum query complexities of poset search in the concrete model are polynomially related.

Theorem 7.4.6. *Let S be an n-element poset. Then $D(S) = O(Q_2(S)^2 \log n) = O(Q_2(S)^3)$.*

Proof. Follows immediately from the quantum lower bounds of Lemma 7.4.3 and the classical upper bound of Lemma 7.4.5. □

7.4.2 Searching a partially sorted array

Consider the following problem. We are given a d-dimensional $m_1 \times m_2 \times \cdots \times m_d$ array of integers T that has been sorted in ascending order in each dimension (i.e. $(i_1 \leq j_1) \wedge (i_2 \leq j_2) \wedge \cdots \wedge (i_d \leq j_d) \Rightarrow T(i_1, \ldots, i_d) \leq T(j_1, \ldots, j_d)$), and must find a given integer in this array, or output "not found", using the minimum number of queries to the array. It is easy to see that this structure gives rise to a partially ordered set; see Figure 7.2 for the Hasse diagram of such a poset.

1	3	6
2	4	8
5	7	9

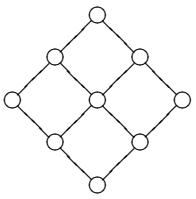


Figure 7.2: A 3×3 2-dimensional array sorted by rows and columns, and its corresponding Hasse diagram.

We are particularly interested in the special case where $m_i = m$ for all i. Call the poset corresponding to such a d-dimensional array $S_{d,m}$. Linial and Saks give [95] an $O(m^{d-1})$ classical algorithm for the problem of searching $S_{d,m}$, which is asymptotically optimal. When $d = 2$, it is easy to see that we have $w(S_{2,m}) = m$. For higher d, Linial and Saks show that $w(S_{d,m}) = \Theta(m^{d-1})$. This follows from consideration of the set of elements that are indexed by a position $(i_1, \ldots, i_d)$ such that $\sum_k i_k = m + 1$; this is clearly an antichain and can be shown to have size $\Theta(m^{d-1})$. It is thus immediate from Lemma 7.4.5 and Lemma 7.4.3 that there exists a quantum algorithm that searches this poset using $O(m^{(d-1)/2}(\log d + \log m))$ queries, which is optimal up to the $(\log d + \log m)$ factor.

However, we can immediately write down an improved algorithm achieving a complexity of $O(m^{(d-1)/2} \log m)$ queries. The algorithm for $d = 1$ is just binary search. For $d = 2$, we nest a binary search algorithm on the rows within Grover search on the columns for an overall query complexity of $O(\sqrt{m} \log m)$. For $d = 3$, the algo-

rithm simply performs Grover search on m copies of the $d = 2$ search algorithm, giving $O(m \log m)$ queries, and so on for $d > 3$.

It is worth noting that this poset structure is an example where searching in the abstract model is significantly easier than in the concrete model. Indeed, there exists a simple $O(d \log m)$ classical algorithm for search in the abstract model: simply perform binary search on each dimension of T.

In the following section, we will give an asymptotically optimal bounded-error quantum algorithm that searches a 2-dimensional $m \times m$ array of *distinct* integers in $O(\sqrt{m})$ queries. This then implies an asymptotically optimal $O(m^{(d-1)/2})$-query algorithm for searching a d-dimensional $m \times m \times \cdots \times m$ array of distinct integers. The optimal d-dimensional algorithm follows from treating the array as the union of m^{d-2} disjoint 2-dimensional $m \times m$ arrays. Each 2-dimensional array is searched by the optimal algorithm, which is treated as an oracle within an overall application of quantum search. Although the 2-dimensional search algorithm is bounded-error, a version of quantum search which can cope with bounded-error inputs (due to Høyer, Mosca and de Wolf [76]) can be used to achieve a constant probability of success in $O(m^{(d-1)/2})$ queries.

7.4.3 Optimal search of a 2-dimensional array sorted by rows and columns

In this section, we give an asymptotically optimal algorithm to search for a known integer a within an $r \times c$ 2-dimensional array of distinct integers sorted by rows and columns. We will start by describing a classical algorithm for the same problem, which is asymptotically (but not exactly [95]) optimal. The algorithm's operation will be described in terms of the original array, rather than the more abstract poset structure. Call the $\lceil \frac{r}{2} \rceil$'th row of the array the *central* row R, and similarly let the $\lceil \frac{c}{2} \rceil$'th column be the central column C.

If $r \leq c$, begin by performing binary search for a on the central column, using $O(\log r)$ queries. If $r > c$, do the same, but on the central row, using $O(\log c)$ queries. Assume $r \leq c$ and that a is not in the central column (otherwise, a will be found by the binary search, and can be returned immediately). Then by the end of the binary search we will have found an element x such that $x = \max_{x' \in C}(x' < a)$, and an element y such that $y = \min_{y' \in C}(y' > a)$ (so y is positioned directly below x in the array). This then implies that all elements in the array above and to the left of x are also less than a, and similarly all elements below and to the right of y are greater than a, so these elements can be discarded. As x and y are in the central column, we must have excluded at least half of the elements in the array from consideration.

We are then left with two smaller instances of the same problem to solve: the subarray below and to the left of y, and the subarray above and to the right of x. The

algorithm proceeds to search these subarrays recursively until a is found, performing binary search on central rows or central columns as appropriate.

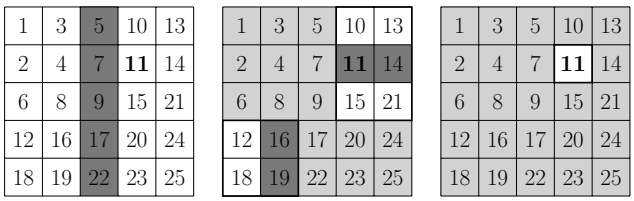


Figure 7.3: Example of the classical algorithm's operation when searching for the element 11: dark grey squares are those that are searched in each round, light grey squares have been excluded from consideration, white squares are still to be searched. Here, 11 is found with only 2 levels of recursion.

How many queries to the array does this algorithm require? Let $T(m)$ denote the number of queries used to search an $r \times c$ array, with $m = \max(r, c)$. Then it is easy to see that $T(m)$ will be maximised if each level of binary search always terminates as close to the centre of the central column/row as possible (thus maximising the number of queries required for binary search in the next level of recursion). We therefore have

$$T(m) \leq \lceil \log_2 m + 1 \rceil + 2T(m/2) \tag{7.5}$$

and unwinding the recursion gives $T(m) = O(m)$.

We would like to find an analogous quantum algorithm that achieves some reduction in queries by searching the subarrays in superposition, rather than sequentially. In fact, it turns out that we can make a general statement about when recursive classical search algorithms can be turned into improved quantum search algorithms, which is given as the following lemma. The proof is a fairly straightforward generalisation of a powerful result of Aaronson and Ambainis [1], and is given in the next section.

Lemma 7.4.7. *Let P_n be the problem of searching an abstract database, parametrised by an abstract size n, for a known element which may or may not be in the database. Let $T(n)$ be the time required for a bounded-error quantum algorithm to solve P_n, i.e. to find the element, or output "not found". Let P_n satisfy the following conditions:*

- *If $n \leq n_0$ for some constant n_0, then there exists an algorithm to find the element, if it is contained in the database, in time $T(n) \leq t_0$, for some constant t_0.*
- *If $n > n_0$, then the database can be divided into k sub-databases of size at most $\lceil n/k \rceil$, for some constant $k > 1$.*
- *If the element is contained in the original database, then it is contained in exactly one of these sub-databases.*
- *Each division into sub-databases uses time $f(n)$, where $f(n) = O(n^{1/2-\epsilon})$ for some $\epsilon > 0$.*

Then $T(n) = O(\sqrt{n})$.

We first show that the search problem in question fits the conditions of the lemma, and then turn to proving it. We consider the problem to be parametrised by a "size" $m = \max(r, c)$. Assuming that a is stored in the set and is not stored in the central row/column, one step of the classical procedure given above will divide any array of size m into two arrays of size at most $\lceil m/2 \rceil$, exactly one of which contains a, in time $O(\log m)$. This division can be performed recursively until the arrays are reduced to a constant size. In the case where the binary search of the central row/column actually finds a, the algorithm can easily be modified to not return a immediately, but to restrict the search area in the next recursion to two subarrays, exactly one of which includes a, and both of which are of size at most $\lceil m/2 \rceil$.

There thus exists a quantum algorithm, given explicitly below, that can find an arbitrary element a in the array in $O(\sqrt{m})$ time, and hence $O(\sqrt{m})$ queries.

7.4.4 Proof of Lemma 7.4.7

We now prove a somewhat generalised version of a powerful result that was shown by Aaronson and Ambainis [1] in the course of their work on quantum search of spatial regions. Informally, we would like to be able to find "cookbook" quantum algorithms for search problems for which there exists a recursive classical algorithm. We imagine that we are searching for a distinguished element in an abstract "database" that is parametrised by an abstract "size" n, which is some function of the number of elements in the database. We also imagine that we have the ability to search the database recursively: that is, in time given by some function $f(n)$, we can reduce the search problem to searching k instances of databases of size $\leq \lceil n/k \rceil$, for some constant $k > 1$.

It is straightforward to show that, classically, the marked element can be found deterministically in $O(n)$ time, by repeated use of this recursive search. An alternative probabilistic classical algorithm for this problem would be: split the input into a number of parts, pick one part uniformly at random, and call yourself recursively on that part. Our quantum algorithm will apply amplitude amplification to this probabilistic algorithm. It will turn out to be advantageous to only amplify a small number of times within the recursive algorithm, and then to amplify again at the end. Amplifying to high probabilities too soon is less efficient [1]; conversely, if amplitude amplification were only applied at the end of the algorithm, we would require $\Omega(\sqrt{n})$ iterations to amplify the probability to a constant. If the process of dividing the input required time $f(n) = \omega(1)$, this would hurt the overall complexity.

The fundamental amplitude amplification result of Brassard et al. [28] states that, given a quantum algorithm A with success probability ϵ, we can achieve a success probability of $\Omega(1)$ with only $O(1/\sqrt{\epsilon})$ uses of A. However, here we will need a tighter analysis due to Aaronson and Ambainis [1], as constants are important within the

recursive algorithm.

Lemma 7.4.8. *Given a quantum algorithm with success probability at least ϵ, then by executing it $t = 2m + 1$ times, where $m \leq \pi/(\arcsin\sqrt{\epsilon}) - 1/2$, we can achieve success probability at least $(1 - \frac{1}{3}t^2\epsilon)t^2\epsilon$.*

We are now ready to give a formal definition of a quantum algorithm for recursive search problems, and to upper-bound its time complexity. The algorithm and its analysis closely follow the results on spatial search of a d-dimensional cube of [1].

Our quantum algorithm will be parametrised by two constants α and δ, whose values we will take to be $\delta = \epsilon/2$, $\alpha = \frac{\epsilon(4-3\epsilon)}{8(2-\epsilon)}$, and will be based on the following probabilistic classical algorithm:

If $n \leq n_0$, then find the desired element directly or output "not found" (using at most t_0 steps). Otherwise, assume that there exists an integer l such that $n^\delta = k^l$ [2]. Recursively divide the problem into subproblems l times, leaving n^δ subproblems, each of size at most $n^{1-\delta}$. Pick one of the parts at random, and call yourself recursively on that part. Repeat until the desired element has been found.

We will perform a number of iterations of amplitude amplification on this algorithm such that it is executed n^α times. Then we have

$$\begin{aligned}
T(n) &\leq n^\alpha \left(\sum_{i=0}^{l-1} k^i f(n/k^i) + T(n^{1-\delta}) \right) \\
&\leq n^\alpha \left(l n^\delta f(n) + T(n^{1-\delta}) \right) \\
&= n^\alpha f'(n) + n^{\alpha(1+(1-\delta))} f'(n^{1-\delta}) + n^{\alpha(1+(1-\delta)+(1-\delta)^2)} f'(n^{(1-\delta)^2}) + \ldots + t_0 \\
&= O(n^{\alpha(1+(1-\delta)+(1-\delta)^2+\ldots)}) \\
&= O(n^{\alpha/\delta})
\end{aligned}$$

where we define $f'(n) = l n^\delta f(n) = O(n^{(1-\epsilon)/2} \log n)$. The fourth line follows because $(1-\epsilon)/2 < \alpha(1/\delta - 1)$, so for any $m \geq 0$ we have $f'(n^{(1-\delta)^m}) = O(n^{(1-\delta)^m(1-\epsilon)/2} \log n) = o(n^{(\alpha/\delta)(1-\delta)^{m+1}})$, so the $f'(n^{(1-\delta)^m})$ parts of the third line are negligible.

We now calculate a lower bound on the probability of success $P(n)$ of this algorithm. If there were no amplification, we would have $P(n) \geq n^{-\delta} P(n^{1-\delta})$ for $n > n_0$, and $P(n) = 1$ for $n \leq n_0$. So, by Lemma 7.4.8, we have

$$\begin{aligned}
P(n) &\geq (1 - n^{2\alpha-\delta}/3) n^{2\alpha-\delta} P(n^{1-\delta}) \\
&= [(1 - n^{2\alpha-\delta}/3)(1 - n^{(2\alpha-\delta)(1-\delta)}/3) \cdots]\, n^{(2\alpha-\delta)(1+(1-\delta)+(1-\delta)^2+\ldots)} \\
&= [(1 - n^{2\alpha-\delta}/3)(1 - n^{(2\alpha-\delta)(1-\delta)}/3) \cdots]\, \Omega(n^{2\alpha/\delta-1})
\end{aligned}$$

We claim that the remaining product of bracketed terms is lower bounded by a constant

[2] We assume here that l and n^α are integers. One can show that the need to round these quantities up or down has no effect on the overall asymptotic complexity.

that does not depend on n. First, note that the algorithm recurses R times, for some $R = O(\log \log n)$. Now

$$\prod_{k=0}^{R}(1-\frac{1}{3}n^{(2\alpha-\delta)(1-\delta)^k}) \geq 1-\frac{1}{3}\sum_{k=0}^{O(\log\log n)} n^{(2\alpha-\delta)(1-\delta)^k} \geq 1-O(n^{2\alpha-\delta}\log\log n) = 1-o(1)$$

giving the result $P(n) = \Omega(n^{2\alpha/\delta-1})$.

By wrapping this algorithm in another level of amplitude amplification, we can use $O(P(n)^{-1/2})$ iterations of it to achieve a constant probability of success of finding the marked element in time $O(T(n)P(n)^{-1/2}) = O(n^{\alpha/\delta}n^{1/2-\alpha/\delta}) = O(\sqrt{n})$.

7.4.5 Finding the intersection of two increasing lists

Classically, there is a correspondence between the problem of searching in an $r \times c$ array sorted by rows and columns and merging two sorted lists of length r and c: any decision tree for the one problem gives a decision tree for the other [95]. However, this does not appear to hold for quantum algorithms; indeed, it is straightforward to show, using Holevo's Theorem [74], an $\Omega(r + c)$ quantum query lower bound for the merge problem. Nevertheless, we can define a natural search problem that turns out to arise from the poset search problem.

Problem: Given two lists of integers in increasing order, output an integer that occurs in both lists, or report that no such integer exists.

This can be thought of as a special case of the element distinctness problem [2]. It was studied by Buhrman et al. [34], who also refer to it as the *monotone claw* problem (a claw is an input on which two functions take the same value). Let the lists be denoted L and M and be of length l and m respectively, with $l \geq m$. Then the ingenious algorithm of [34] finds an integer occuring in both lists using $O(\sqrt{l}c^{\log^* l})$ queries, where $\log^*$ is the iterated logarithm function and c is a constant. This algorithm can easily be translated into the setting of poset search, and allows an $m \times m$ array that is sorted by rows and columns, and may contain duplicates, to be searched using $O(\sqrt{m}c^{\log^* m})$ time for some constant c.

Here, we will go in the other direction, and show that the algorithm of Section 7.4.3 can be used to find the integer occurring in both sorted lists using $O(\sqrt{l})$ time. As noted in [34], there is an $\Omega(\sqrt{l})$ lower bound for this problem, so the algorithm given here is asymptotically optimal. However, as $c^{\log^* l}$ is already a near-constant function, the new algorithm may be only of theoretical interest, and we describe it briefly.

Consider a notional $l \times m$ array T where entry $T(x, y)$ contains the value $L_x - M_{m+1-y}$. Querying one entry of T uses one query to each list. As the entries in L and M are in increasing order, it is easy to see that T is increasing along rows and columns, and that finding a 0 entry in T corresponds to finding an element of L that also occurs

in M. Call such an element a *match*. If there is only one match, it is immediate that the algorithm of the previous section can be used to find the single 0 entry in T, or output that no such entry exists, in time $O(\sqrt{l})$.

There are two possible reasons for there being more than one match. The first is that L and M may contain duplicate elements (i.e. may be increasing but not strictly increasing). If this is the case, and if one of the duplicate elements in L (say) is also in M, there will be a contiguous rectangle of 0 entries in the array T (call this a *zero block*), rather than a single 0. Assume that there is only one zero block. Then the algorithm of Section 7.4.3 must be modified to ensure that, after any splitting of the array into two subarrays, at most one of these arrays contains a 0 entry; i.e. to ensure that the zero block does not get split across subarrays. This is necessary to ensure that the conditions of Lemma 7.4.7 are satisfied. It is easy to see that, in each round of recursion, the zero block can only be split if it lies across a row or column that is used for binary search in that recursion. In order to ensure that only one of the two subarrays produced contains part of the zero block in this case, the binary search of a row (column) can simply be modified to split on the first or last zero entry in that row (column), with no change to the asymptotic complexity. Call this new algorithm the single-block algorithm.

The second case where there may be more than one match is when there is more than one element in L that also occurs in M (or vice versa). In this case, the idea (inspired by [1]) is to reduce the problem to searching for a single zero block by probabilistically removing elements from the lists. The extended algorithm first runs the single-block algorithm. Assuming that this algorithm outputs "not found", the next step is to produce a new pair of smaller lists $L^{(2)}$ and $M^{(2)}$, which will give rise to a notional array $T^{(2)}$, where $T^{(2)}(x, y) = L^{(2)}_x - M^{(2)}_{m+1-y}$.

The reduction in size is achieved by first splitting each list into chunks of size 2. One element (picked at random) within each chunk of L is included in $L^{(2)}$, and similarly for M and $M^{(2)}$. The single-block algorithm is then run on these smaller lists. Assuming that the result is again "not found", the chunk size is doubled to 4, and the process repeats, using a chunk size of 2^k in each round k. Assuming that the single-block algorithm does not find a match in any of the $O(\log l)$ rounds, the final output is "not found". The time required for this overall algorithm is then bounded by $O\left(\sum_k \sqrt{l/2^k}\right) = O(\sqrt{l})$.

We sketch a proof that this algorithm succeeds with constant probability. First, it is easy to see that there can be at most one zero block in each row and column of the array $T^{(k)}$ in any round k. Using this, one can show that, if there are z zero blocks in T, the probability that exactly one remains in $T^{(k)}$ is at least $z/2^{2k}(1 - z/2^{2k})$. If we take $k = \lceil \log z/2 \rceil + 1$, this is lower bounded by a constant, so for any z the single-block algorithm succeeds with constant probability in at least one round.

7.5 Random partially ordered sets

Finally, we briefly discuss the generic behaviour of quantum poset search – i.e. given a poset picked uniformly at random from the set of all n-element posets, how many quantum queries do we require to search it? It turns out that this question is easy to answer using a powerful (and perhaps surprising) theorem of Kleitman and Rothschild that says that almost all posets have only three layers.

Theorem 7.5.1. *(***Kleitman and Rothschild [88]***)*
Let Q_n be the set of n-element posets consisting of three layers L_1, L_2 and L_3, with $|L_1|, |L_3| = n/4 + o(n)$ and $|L_2| = n/2 + o(n)$, such that, for $j > i$, $x \in L_i$ and $y \in L_j$ implies $x \not\leq y$. Then the probability that an n-element partially ordered set is in Q_n is at least $1 - o(1)$.

It is thus immediate that quantum search of a random poset in either of the two models studied requires $\Theta(\sqrt{n})$ queries, as all of the maximal elements in the poset will need to be queried, contrasting with the classical $\Omega(n)$ queries required [35].

One could also consider a different model – the *random graph model* [35], which is parametrised by a size n and a probability p. In this model, a poset $P_{n,p}$ is produced by taking the transitive closure of a relation $R_{n,p}$ which includes each pair $(x, y) \in [n]^2$ with $x < y$ with probability p. For constant p, it is known that $w(P_{n,p}) = O(\sqrt{\log n})$, so a random poset in this model is "tall and thin" by comparison with the uniform model. We leave the question of the complexity of quantum search of random posets in this model open.

7.6 Conclusions

We have given general upper and lower bounds on quantum search of partially ordered sets, in two different models. Satisfyingly, in the two cases where results were already known on poset search (i.e. totally ordered sets and unstructured sets), our lower bounds reduce to known lower bounds, and our new quantum algorithms are (asymptotically) as efficient as the known most efficient algorithms. The bounds in the concrete model are perhaps particularly interesting, because they follow from decomposing a poset into "structured" and "unstructured" components, and show that, intuitively, almost all the speed-up that can be obtained from quantum search of a poset S is obtained from searching the unstructured parts of S.

Although we concentrated on the model of query complexity, our quantum algorithms in both models are efficient in the sense that, given a poset S to be searched, quantum circuits for the algorithms given here can be produced in time polynomial in the size of S. Also, the non-query transformations used by the algorithms given here are efficiently implementable.

However, there are still several open questions. Firstly: in the abstract model, is there a general lower bound of $Q(S) = \Omega(\log n)$? This would be an interesting generalisation of the known logarithmic quantum lower bound on searching an ordered list [6, 77]. Also, can the logarithmic factors in the quantum upper bounds in both models be improved, perhaps by being changed into additive terms?

There are several possible extensions involving search for multiple marked elements. In the abstract model, can a $O\left(\log n/\sqrt{\gamma^S}\right)$-query algorithm be produced for search for multiple marked elements in arbitrary posets? In the concrete model, could the algorithm of Section 7.4.3 be extended to arrays that may contain duplicate elements?

Bibliography

[1] S. Aaronson and A. Ambainis. Quantum search of spatial regions. *Theory of Computing*, 1:47–79, 2005. `quant-ph/0303041`.

[2] S. Aaronson and Y. Shi. Quantum lower bounds for the collision and the element distinctness problems. *J. ACM*, 51(4):595–605, 2004.

[3] A. Acín. Statistical distinguishability between unitary operations. *Phys. Rev. Lett.*, 87(17):177901, 2001. `quant-ph/0102064`.

[4] D. Aharonov, A. Ambainis, J. Kempe, and U. Vazirani. Quantum walks on graphs. In *Proc. 33rd Annual ACM Symp. Theory of Computing*, pages 50–59, 2001. `quant-ph/0012090`.

[5] N. Alon and J. Spencer. *The probabilistic method*. Wiley-Interscience Series in Discrete Mathematics and Optimization. Wiley, New York, 2000.

[6] A. Ambainis. A better lower bound for quantum algorithms searching an ordered list. In *Proc. 40th Annual Symp. Foundations of Computer Science*, pages 352–357. IEEE, 1999. `quant-ph/9902053`.

[7] A. Ambainis. Quantum lower bounds by quantum arguments. *J. Comput. Syst. Sci.*, 64:750–767, 2002. `quant-ph/0002066`.

[8] A. Ambainis. Quantum walk algorithm for element distinctness. In *Proc. 45th Annual Symp. Foundations of Computer Science*, pages 22–31, 2004. `quant-ph/0311001`.

[9] A. Ambainis. Polynomial degree vs. quantum query complexity. *J. Comput. Syst. Sci.*, 72(2):220–238, 2006. `quant-ph/0305028`.

[10] A. Ambainis, K. Iwama, A. Kawachi, H. Masuda, R. Putra, and S. Yamashita. Quantum identification of Boolean oracles. In *Proc. STACS 2004*, pages 93–104. Springer, 2004. `quant-ph/0403056`.

[11] A. Ambainis, K. Iwama, A. Kawachi, R. Raymond, and S. Yamashita. Improved algorithms for quantum identification of boolean oracles. *Theor. Comput. Sci.*, 378:41–53, 2007. `quant-ph/0411204`.

[12] A. Ambainis, L. J. Schulman, A. Ta-Shma, U. Vazirani, and A. Wigderson. The quantum communication complexity of sampling. *SIAM J. Comput.*, 32(6):1570–1585, 2003.

[13] A. Atici and R. Servedio. Improved bounds on quantum learning algorithms. *Quantum Information Processing*, 4(5):355–386, 2005. `quant-ph/0411140`.

[14] D. Avis, J. Hasegawa, Y. Kikuchi, and Y. Sasaki. A quantum protocol to win the graph colouring game on all Hadamard graphs. *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences*, E89-A(5):1378–1381, 2006. `quant-ph/0509047`.

[15] Z. D. Bai. Convergence rate of expected spectral distributions of large random matrices. Part II. Sample covariance matrices. *Ann. Prob.*, 21(2):649–672, 1993.

[16] Z. D. Bai. Methodologies in spectral analysis of large dimensional random matrices, a review. *Statist. Sinica*, 9(3):611–677, 1999.

[17] H. Barnum and E. Knill. Reversing quantum dynamics with near-optimal quantum and classical fidelity. *J. Math. Phys.*, 43(5):2097–2106, 2002. `quant-ph/0004088`.

[18] S. Basu, R. Pollack, and M. F. Roy. *Algorithms in Real Algebraic Geometry*, volume 10 of *Algorithms and Computation in Mathematics*. Springer, 2006.

[19] R. Beals, H. Buhrman, R. Cleve, M. Mosca, and R. de Wolf. Quantum lower bounds by polynomials. *J. ACM*, 48(4):778–797, 2001. `quant-ph/9802049`.

[20] Y. Ben-Asher, E. Farchi, and I. Newman. Optimal search in trees. *SIAM J. Comput.*, 28(6):2090–2102, 1999. ECCC TR96-044.

[21] C. Bennett, E. Bernstein, G. Brassard, and U. Vazirani. Strengths and weaknesses of quantum computing. *SIAM J. Comput.*, 26(5):1510–1523, 1997. `quant-ph/9701001`.

[22] C. H. Bennett, P. Hayden, D. Leung, P. Shor, and A. Winter. Remote preparation of quantum states. *IEEE Trans. Inform. Theory*, 51(1):56–74, 2005. `quant-ph/0307100`.

[23] E. Bernstein and U. Vazirani. Quantum complexity theory. *SIAM J. Comput.*, 26(5):1411–1473, 1997.

[24] P. Bocchieri and A. Loinger. Quantum recurrence theorem. *Phys. Rev.*, 107(2):337–338, 1957.

[25] K. Bogart. *Introductory combinatorics (3rd edition)*. Brooks Cole, 2000.

[26] G. Brassard, A. Broadbent, and A. Tapp. Quantum pseudo-telepathy. *Found. Physics*, 35(11):1877–1907, 2005. quant-ph/0407221.

[27] G. Brassard, R. Cleve, and A. Tapp. Cost of exactly simulating quantum entanglement with classical communication. *Phys. Rev. Lett.*, 83(9):1874–1877, 1999. quant-ph/9901035.

[28] G. Brassard, P. Høyer, M. Mosca, and A. Tapp. Quantum amplitude amplification and estimation. *Quantum Computation and Quantum Information: A Millennium Volume*, pages 53–74, 2002. quant-ph/0005055.

[29] H. Buhrman, R. Cleve, R. de Wolf, and C. Zalka. Bounds for small-error and zero-error quantum algorithms. In *Proc. 40th Annual Symp. Foundations of Computer Science*, pages 358–368. IEEE, 1999. cs/9904019.

[30] H. Buhrman, R. Cleve, J. Watrous, and R. de Wolf. Quantum fingerprinting. *Phys. Rev. Lett.*, 87(16):167902, 2001. quant-ph/0102001.

[31] H. Buhrman, R. Cleve, and A. Wigderson. Quantum vs. classical communication and computation. In *Proc. 30th Annual ACM Symp. Theory of Computing*. ACM Press, 1998. quant-ph/9802040.

[32] H. Buhrman and R. de Wolf. Communication complexity lower bounds by polynomials. In *Proc. 16th Annual IEEE Conf. Computational Complexity*, pages 120–130, 2001. cs.CC/9910010.

[33] H. Buhrman and R. de Wolf. Complexity measures and decision tree complexity: a survey. *Theoretical Computer Science*, 288:21–43, 2002.

[34] H. Buhrman, C. Dürr, M. Heiligman, P. Høyer, F. Magniez, M. Santha, and R. de Wolf. Quantum algorithms for element distinctness. *SIAM J. Comput.*, 34(6):1324–1330, 2005. quant-ph/0007016.

[35] R. Carmo, J. Donadelli, Y. Kohayakawa, and E. Laber. Searching in random partially ordered sets. *Theoretical Computer Science*, 321(1):41–57, 2004.

[36] A. Chefles. Quantum state discrimination. *Contemporary Physics*, 41(6):401–424, 2001. quant-ph/0010114.

[37] W. K. Chen. *Graph theory and its engineering applications*. Singapore: World Scientific, 1996.

[38] A. Childs, R. Cleve, E. Deotto, E. Farhi, S. Gutmann, and D. Spielman. Exponential algorithmic speedup by a quantum walk. In *Proc. 35th Annual ACM Symp. Theory of Computing*, pages 59–68, 2003. quant-ph/0209131.

[39] A. Childs, E. Farhi, and S. Gutmann. An example of the difference between quantum and classical random walks. *Quantum Information Processing*, 1:35–43, 2002. quant-ph/0103020.

[40] A. Childs, A. Landahl, and P. Parrilo. Quantum algorithms for the ordered search problem via semidefinite programming. *Phys. Rev. A.*, 75(3):032335, 2007. quant-ph/0608161.

[41] R. Cleve and H. Buhrman. Substituting quantum entanglement for communication. *Phys. Rev. A.*, 56(2):1201–1204, 1997. quant-ph/9704026.

[42] R. Cleve, W. van Dam, M. Nielsen, and A. Tapp. Quantum entanglement and the communication complexity of the inner product function. In *Selected papers from the First NASA International Conference on Quantum Computing and Quantum Communications*, pages 61–74, 1998. quant-ph/9708019.

[43] R. Cleve, P. Høyer, B. Toner, and J. Watrous. Consequences and limits of nonlocal strategies. In *Proc. 19th Annual IEEE Conf. Computational Complexity*, pages 236–249, 2004. quant-ph/0404076.

[44] C. Colbourn and J. Dinitz. *The CRC Handbook of Combinatorial Designs*. CRC Press, 1996.

[45] T. Cormen, C. Leiserson, R. Rivest, and C. Stein. *Introduction to Algorithms (second edition)*. MIT Press, 2007.

[46] W. van Dam and P. Hayden. Renyi-entropic bounds on quantum communication, 2002. quant-ph/0204093.

[47] W. van Dam, M. Mosca, and U. Vazirani. How powerful is adiabatic quantum computation? In *Proc. 42nd Annual Symp. Foundations of Computer Science*, pages 279–287. IEEE, 2001. quant-ph/0206003.

[48] E. B. Davies. Information and quantum measurement. *IEEE Trans. Inform. Theory*, 24(5):596–599, 1978.

[49] R. P. Dilworth. A decomposition theorem for partially ordered sets. *The Annals of Mathematics*, 51(1):161–166, 1950.

[50] H.-D. Ebbinghaus, H. Hermes, F. Hirzebruch, M. Koecher, K. Mainzer, J. Neukirch, A. Prestel, and R. Remmert. *Numbers*. Graduate Texts in Mathematics. Springer Verlag, 1996.

[51] Y. C. Eldar and G. D. Forney, Jr. On quantum detection and the square-root measurement. *IEEE Trans. Inform. Theory*, 47(3):858–872, 2001. quant-ph/0005132.

[52] Y. C. Eldar, A. Megretski, and G. Verghese. Designing optimal quantum detectors via semidefinite programming. *IEEE Trans. Inform. Theory*, 49(4):1007–1012, 2003. quant-ph/0205178.

[53] M. Fannes. A continuity property of the entropy density for spin lattice systems. *Comm. Math. Phys.*, 31(4):291–294, 1973.

[54] E. Farhi and S. Gutmann. Quantum computation and decision trees. *Phys. Rev. A.*, 58(2):915–928, 1998. quant-ph/9706062.

[55] Edward Farhi, Jeffrey Goldstone, Sam Gutmann, and Michael Sipser. Quantum computation by adiabatic evolution. Technical Report MIT-CTP-2936, MIT, 2000. quant-ph/0001106.

[56] P. Frankl and V. Rödl. Forbidden intersections. *Trans. Amer. Math. Soc.*, 300(1):259–286, 1987.

[57] V. Galliard, A. Tapp, and S. Wolf. The impossibility of pseudotelepathy without quantum entanglement. In *Proc. ISIT 2003*, page 457. IEEE, 2003.

[58] V. Galliard and S. Wolf. Pseudo-telepathy, entanglement, and graph colorings. In *Proc. ISIT 2002*, page 101. IEEE, 2002.

[59] M. R. Garey. Optimal binary identification procedures. *SIAM J. Appl. Math.*, 23(2):173–186, 1972.

[60] D. Gavinsky. On the role of shared entanglement. *Quantum Inf. Comput.*, 8:82–95, 2008. quant-ph/0604052.

[61] D. Gavinsky, J. Kempe, and R. de Wolf. Strengths and weaknesses of quantum fingerprinting. In *Proc. 21^{st} Annual IEEE Conf. Computational Complexity*, pages 288–298, 2006. quant-ph/0603173.

[62] D. Gavinsky, J. Kempe, I. Kerenidis, R. Raz, and R. de Wolf. Exponential separations for one-way quantum communication complexity, with applications to cryptography. In *Proc. 39^{th} Annual ACM Symp. Theory of Computing*, pages 516–525, 2007. quant-ph/0611209.

[63] I. S. Gradshteyn and I. M. Ryzhik. *Table of integrals, series and products*. Academic Press, New York, 1980.

[64] L. Grover. Quantum mechanics helps in searching for a needle in a haystack. *Phys. Rev. Lett.*, 79(2):325–328, 1997. quant-ph/9706033.

[65] J. Gu, P. Purdom, J. Franco, and B. Wah. Algorithms for the Satisfiability (SAT) problem: a survey. In *DIMACS Series in Discrete Mathematics and Theoretical Computer Science*. American Mathematical Society, 1997.

[66] M. Hadamard. Résolution d'une question relative aux déterminants. *Bull. des Sciences Mathematiques*, 17:240–246, 1893.

[67] P. Hausladen, R. Jozsa, B. Schumacher, M. Westmoreland, and W. Wootters. Classical information capacity of a quantum channel. *Phys. Rev. A.*, 54(3):1869–1876, 1996.

[68] P. Hausladen and W. Wootters. A "pretty good" measurement for distinguishing quantum states. *J. Mod. Opt.*, 41(12):2385–2390, 1994.

[69] P. Hayden, D. Leung, and A. Winter. Aspects of generic entanglement. *Comm. Math. Phys.*, 265(1):95–117, 2006. `quant-ph/0407049`.

[70] C. W. Helstrom. *Quantum detection and estimation theory.* Academic Press, New York, 1976.

[71] E. Hirsch. New worst-case upper bounds for SAT. *Journal of Automated Reasoning*, 24(4):397–420, 2000.

[72] T. Hogg. Highly structured searches with quantum computers. *Phys. Rev. Lett.*, 80:2473–2476, 1998.

[73] T. Hogg and D. Portnov. Quantum optimization. *Information Sciences*, 128:181–197, 2000. `quant-ph/0006090`.

[74] A. S. Holevo. Bounds for the quantity of information transmitted by a quantum communication channel. *Problemy Peredachi Informatsii*, 9(3):3–11, 1973. English translation *Problems of Information Transmission*, vol. 9, pp. 177-183, 1973.

[75] R. A. Horn and C. R. Johnson. *Matrix Analysis.* Cambridge University Press, 1985.

[76] P. Høyer, M. Mosca, and R. de Wolf. Quantum search on bounded-error inputs. In *Proc. ICALP'03*, pages 291–299, 2003. `quant-ph/0304052`.

[77] P. Høyer, J. Neerbek, and Y. Shi. Quantum complexities of ordered searching, sorting, and element distinctness. *Algorithmica*, 34(4):429–448, 2002. `quant-ph/0102078`.

[78] P. Høyer and R. Špalek. Lower bounds on quantum query complexity. *Bulletin of the European Association for Theoretical Computer Science*, 87:78–103, 2005. `quant-ph/0509153`.

[79] M. Hunziker and D. Meyer. Quantum algorithms for highly structured search problems. *Quantum Information Processing*, 1(3):145–154, 2002.

[80] M. Hunziker, D. Meyer, J. Park, J. Pommersheim, and M. Rothstein. The geometry of quantum learning. quant-ph/0309059.

[81] N. Johnson, S. Kotz, and N. Balakrishnan. *Continuous univariate distributions*. Wiley, 1994.

[82] R. Jozsa. Fidelity for mixed quantum states. *Journal of Modern Optics*, 41(12):2315–2323, 1994.

[83] R. Jozsa and J. Schlienz. Distinguishability of states and von Neumann entropy. *Phys. Rev. A.*, 62(1):012301, 2000. quant-ph/9911009.

[84] E. Kashefi, A. Kent, V. Vedral, and K. Banaszek. A comparison of quantum oracles. *Phys. Rev. A.*, 65(5):050304, 2002. quant-ph/0109104.

[85] F. Kelly. *Reversibility and stochastic networks*. Wiley, 1979.

[86] J. Kempe. Quantum random walks: an introductory overview. *Contemporary Physics*, 44(4):307–327, 2003. quant-ph/0303081.

[87] H. Klauck. Lower bounds for quantum communication complexity. In *Proc. 42nd Annual Symp. Foundations of Computer Science*, pages 288–297. IEEE, 2001. quant-ph/0106160.

[88] D. Kleitman and B. Rothschild. Asymptotic enumeration of partial orders on a finite set. *Trans. Amer. Math. Soc.*, 205:205–220, 1975.

[89] I. Kremer. Quantum communication. Master's thesis, Hebrew University, 1995.

[90] H. Krovi and T. Brun. Hitting time for quantum walks on the hypercube. *Phys. Rev. A.*, 73:032341, 2006. quant-ph/0510136.

[91] E. Kushilevitz and N. Nisan. *Communication Complexity*. Cambridge University Press, 1997.

[92] M. Ledoux. *The concentration of measure phenomenon*. AMS Mathematical Surveys and Monographs 89. American Mathematical Society, 2001.

[93] G. Lindblad. A general no-cloning theorem. *Lett. Math. Phys.*, 47(2):189–196, 1999.

[94] N. Linial and M. Saks. Every poset has a central element. *J. Comb. Th. Ser. A*, 40:195–210, 1985.

[95] N. Linial and M. Saks. Searching ordered structures. *Journal of Algorithms*, 6(1):86–103, 1985.

[96] N. Linial and A. Shraibman. Learning complexity vs. communication complexity, 2006. http://www.cs.huji.ac.il/~nati/PAPERS/lcc.pdf.

[97] N. Linial and A. Shraibman. Lower bounds in communication complexity based on factorization norms. In *Proc. 39th Annual ACM Symp. Theory of Computing*, pages 699–708, 2006. `http://www.cs.huji.ac.il/~nati/PAPERS/quant_cc.pdf`.

[98] G. Long. Grover algorithm with zero theoretical failure rate. *Phys. Rev. A.*, 64(2):022307, 2001. `quant-ph/0106071`.

[99] O. Lopez Acevedo and T. Gobron. Quantum walks on Cayley graphs. *Journal of Physics A: Mathematical and General*, 39(3):585–599, 2006. `quant-ph/0503078`.

[100] F. Magniez, A. Nayak, J. Roland, and M. Santha. Search via quantum walk. In *Proc. 39th Annual ACM Symp. Theory of Computing*, pages 575–584, 2007. `quant-ph/0608026`.

[101] V. A. Marčenko and L. A. Pastur. Distributions of eigenvalues of some sets of random matrices. *Math. USSR-Sb.*, 1:457–486, 1967.

[102] G. Midrijānis. Exact quantum query complexity for total Boolean functions, 2004. `quant-ph/0403168`.

[103] A. Montanaro. A lower bound on the probability of error in quantum state discrimination, 2007. `arXiv:0711.2012`.

[104] C. Moore and A. Russell. Quantum walks on the hypercube. In *Proc. RANDOM '02, LNCS 2483*, pages 164–178, 2002. `quant-ph/0104137`.

[105] A. Nayak and J. Salzman. On communication over an entanglement-assisted quantum channel. In *Proc. 34th Annual ACM Symp. Theory of Computing*, pages 698–704, 2002. `quant-ph/0206122`.

[106] M. W. Newman. *Independent Sets and Eigenspaces*. PhD thesis, Waterloo, 2005.

[107] M. A. Nielsen. *Quantum information theory*. PhD thesis, University of New Mexico, Albuquerque, 1998. `quant-ph/0011036`.

[108] M. A. Nielsen and I. L. Chuang. *Quantum computation and quantum information*. Cambridge University Press, 2000.

[109] K. Onak and P. Parys. Generalization of binary search: searching in trees and forest-like partial orders. In *Proc. 47th Annual Symp. Foundations of Computer Science*, pages 379–388. IEEE, 2006.

[110] C. Papadimitriou. *Computational Complexity*. Addison-Wesley, 1994.

[111] A. Peres. *Quantum theory: concepts and methods*. Kluwer, 1995.

[112] S. Popescu, A. J. Short, and A. Winter. Entanglement and the foundations of statistical mechanics. *Nature Physics*, 2(11):754–758, 2006. quant-ph/0511225.

[113] R. Raz. Exponential separation of quantum and classical communication complexity. In *Proc. 31st Annual ACM Symp. Theory of Computing*, pages 358–367. ACM Press, 1999.

[114] A. A. Razborov. Quantum communication complexity of symmetric predicates. *Izvestiya of the Russian Academy of Science*, 67:145–159, 2003. quant-ph/0204025.

[115] O. Reingold. Undirected ST-connectivity in log-space. In *Proc. 37th Annual ACM Symp. Theory of Computing*, pages 376–385, 2005.

[116] A. Rényi. *Probability theory*. North-Holland, Amsterdam, 1970.

[117] U. Schöning. A probabilistic algorithm for k-SAT and constraint satisfaction problems. In *Proc. 40th Annual Symp. Foundations of Computer Science*, pages 410–414, 1999.

[118] R. Servedio and S. Gortler. Quantum versus classical learnability. In *Proc. 16th Annual IEEE Conf. Computational Complexity*, pages 138–148, 2001. quant-ph/0007036.

[119] S. Severini. On the digraph of a unitary matrix. *SIAM Journal on Matrix Analysis and Applications*, 25(1):295–300, 2003. math.CO/0205187.

[120] S. Severini. On the structure of the adjacency matrix of the line digraph of a regular digraph. *Discrete Appl. Math.*, 154(12):1763–1765, 2006. quant-ph/0210055.

[121] S. Severini. Graphs of unitary matrices. *Ars Combinatoria*, 89, 2008. math.CO/0303084.

[122] N. Shenvi, J. Kempe, and K. Whaley. Quantum random-walk search algorithm. *Phys. Rev. A.*, 67(5):052307, 2003. quant-ph/0210064.

[123] P. W. Shor. Algorithms for quantum computation: discrete logarithms and factoring. In *Proc. 35th Annual Symp. Foundations of Computer Science*, pages 124–134, 1994.

[124] M. Szegedy. Quantum speed-up of Markov chain based algorithms. In *Proc. 45th Annual Symp. Foundations of Computer Science*, pages 32–41, 2004. quant-ph/0401053.

[125] B. Tregenna, W. Flanagan, R. Maile, and V. Kendon. Controlling discrete quantum walks: coins and initial states. *New J. Phys.*, 5:83, 2003. quant-ph/0304204.

[126] A. Uhlmann. The "transition probability" in the state space of a *-algebra. *Rep. Math. Phys.*, 9(2):273–279, 1976.

[127] J. Watrous. Quantum simulations of classical random walks and undirected graph connectivity. *J. Comput. Syst. Sci.*, 62(2):376–391, 2001. `quant-ph/9812012`.

[128] R. de Wolf. *Quantum Computing and Communication Complexity.* PhD thesis, Amsterdam, 2001.

[129] R. de Wolf. Quantum communication and complexity. *Theoretical Computer Science*, 287(1):337–353, 2002.

[130] W. Wootters. Random quantum states. *Foundations of Physics*, 20(11):1365–1378, 1990.

[131] A. Yao. Some complexity questions related to distributive computing. In *Proc. 11^{th} Annual ACM Symp. Theory of Computing*, pages 209–213. ACM Press, 1979.

[132] A. Yao. Quantum circuit complexity. In *Proc. 34^{th} Annual Symp. Foundations of Computer Science*, pages 352–361. IEEE, 1993.

[133] A. Yao. Graph entropy and quantum sorting problems. In *Proc. 36^{th} Annual ACM Symp. Theory of Computing*, pages 112–117. ACM Press, 2004.

[134] C. Zalka. Grover's quantum searching algorithm is optimal. *Phys. Rev. A.*, 60(4):2746–2751, 1999. `quant-ph/9711070`.

[135] K. Zyczkowski and H. Sommers. Average fidelity between random quantum states. *Phys. Rev. A.*, 71:032313, 2005. `quant-ph/0311117`.