MRes in Quantum Technologies

Understanding Quantum Speedup with the Stabilizer Rank Method

Author:
Padraic Henry Drummond Calpin

Supervisor:
Dr. Dan Browne

August 28, 2016

EPSRC Centre for Doctoral Training in Delivering Quantum Technologies
Department of Physics and Astronomy, University College London, Gower Street, London WC1E 6BT

Submitted to fulfill the requirements for the MRes in Quantum Technologies at University College London.

Abstract

The stabilizer rank algorithm is a new framework for classically simulating quantum circuits built out of Clifford+T gates. In particular, it represents an 8-fold quadratic speedup over Aaronson & Gottesman's `CHP` algorithm, with a complexity that scales as $\mathcal{O}(poly(n,t)2^{0.23t})$ for a circuit on n qubits with t T-gates. This method raises questions as to the value of the T-gate as a resource for universal quantum computing. Here, we review classical simulations of quantum computation, and examine complexity of different quantum resources in the stabilizer rank formalism. We argue that the stabilizer rank corresponds to the value of the state as a resource for quantum computation.

Contents

I, Padraic Calpin, confirm that the work presented in this report is my own. Where information has been derived from other sources, I confirm that this has been indicated in the report. Original code for the project can be found at `https://github.com/padraic-padraic/StabilizerRank`.

Signature

28/08/2016

Date

Chapter 1

Introduction

The pursuit of quantum computing is motivated by the promised 'quantum speedup': quadratic or exponential reductions in the computational complexity over the best classical techniques. However, the origins of this quantum advantage are not well understood. Quantum parallelism [1], unbounded multi-partite entanglement [2], and most recently contextuality [3] have all been identified as required for a speed-up over classical methods.
Developing techniques for classical simulation of quantum computation has emerged as an interesting method to try and constrain and study this speedup [2]. Any system that is classically simulable cannot, by definition, exhibit quantum speedup, as we can simply simulate the quantum computation to obtain the result without developing a quantum device. The converse conjecture, however, is not necessarily true, as it depends on the framework used to study the quantum circuits. Most modern proposals for a quantum processor are based on a quantum error correcting code, such as the Surface Code, using Clifford gates and additional 'magic state' ancillae to implement a universal gate set. Previous classical simulations of these 'Clifford+T' circuits had a computational complexity that scaled exponentially in the T-count [4], and this has been considered a strong signifier of available quantum speedup [2].

However, recent developments have called into question the value of these magic states as a resource. A new algorithm by Bravyi and Gosset [5], based on earlier work by Bravyi, Smolin & Smith [6], leads to a significant reduction in the exponential scaling as a function of the T-count. In particular, this amounts to an 8-fold quadratic reduction in computational complexity.
What makes this result especially interesting is that the authors postulate that this significant reduction in computational complexity is only achievable for magic states. This would suggest that they are in some sense the weakest potential resource for quantum computation over arbitrary states.

In this report, we explore the role of classical simulation in understanding and constraining quantum speedup. We present a review of the theory of quantum computing, including a discussion of classical simulation techniques, before introducing the techniques of Bravyi, Gosset, Smolin & Smith. We attempt to address their conjecture on the value of magic states as a resource, and use their techniques to examine alternative resources for qubit quantum computing.

Chapter 2

Literature Review

The origin of quantum computation is typically attributed to Richard Feynman, in a keynote address at the 1981 'First Conference on the Physics of Computation' at MIT [7]. Feynman's speech discussed the problems of simulating physical systems computationally. As one candidate solution, he proposed the technique of quantum simulation: using the controlled dynamics of one quantum system to probe the dynamics of another.

At the same conference, Peter Benioff presented a paper on using Hamiltonian evolution of a spin-lattice system to realise a Turing Machine [8], an abstract model used to study universal classical computation. The state of the system is represented by an 'tape', broken into cells which store binary values. The computation is carried out by a 'head' or processor, which can read and write bits and move along the tape according to a programme [1]. This simplified model is often employed in complexity theory, where the number of 'steps' taken by the head and number of cells used give the temporal and spatial resources consumed in the computation. In essence, Benioff proposed a quantum implementation of classical computation, encoding the binary state in the spin of the lattice, where the reading and writing would be controlled by spin-spin interactions [8]. Feynman's proposed quantum simulator followed a similar scheme, premised on applying annihilation, creation and 'number' operators to different components of the system. This is analogous to subtracting, adding or reading a value on a Turing machine tape. Feynman similarly noted that a two-level spin system would serve as the smallest implementation of such a device [7].

Both schemes are early proposal of the 'qubit', a two-level quantum system used as a building block in computation and communication protocols. The earliest example is attributed to Stephen Wiesner, who discussed using polarisation states of photons to implement a secure serial number in a 'Quantum Money' scheme [9]. Wiesner developed this scheme in the 70s, but was unable to publish the work until after Feynman and Benioff's talks [10]. The term qubit was later coined by Schumacher, in his paper on a quantum analogue of Shannon's noiseless coding theorem [11].

The 'quantum advantage' in these early proposals is presented somewhat abstractly. In Benioff's paper, he argues that implementing computation directly on a physical system should offer speed, size and energy advantages over existing classical implementations [8], but doesn't discuss these effects as originating from the quantum mechanical nature of the system. In Feynman's talk, however, the advantages of using a quantum simulator are summarised succinctly in the popular quote:
"Nature isn't classical, dammit, and if you want to make a simulation of nature you better make it quantum mechanical!"

2.1 Quantum Turing Machines

This question of what computational resources we need to simulate arbitrary physical systems was formalised by Deutsch in his 1985 paper 'Quantum theory, the Church-Turing principle, and the universal quantum computer'. This paper is widely considered as establishing the theory of quantum computation, as it formally defines the principle of 'universal quantum computing' [1].

In this paper, Deutsch relates the Church-Turing thesis, a statement about the nature of computationally solvable problems, to the problem of simulating physical systems highlighted by Feynman. This gives a novel definition of a universal computing machine: any device capable of perfectly simulating a finitely realisable physical system with finite resources [1].
The classical Turing Machine fails this definition, as binary numeral systems cannot be used to represent continuous real numbers, as would be required to simulate classical physical systems [1]. Jozsa describes this strengthened Church-Turing-Deutsch thesis as requiring that questions of 'What is computable?' be grounded in the laws of physics and not characterised by mathematics alone [12].

Instead, Deutsch introduces a model of a quantum Turing machine, which passes this Church-Turing-Deutsch criterion for simulating finite quantum mechanical systems. This is an extension of Feynman's ideas, and a quantum measurement automaton discussed by Albert [13]. Unlike Benioff's proposal, which restricted the system to classical computational states, the state of this device is explicitly quantum mechanical, allowing any state in the Hilbert space $\mathcal{H} = \mathbb{C}^{2^n} \otimes \mathbb{C}^{2^m}$ defined by the n qubits in the processor m qubits in the 'tape'. The evolution of the combined processor-tape system is the described with unitary dynamics [1].
Interestingly, Deutsch proves that a quantum Turing machine would still be restricted to solving the class of 'General Recursive' functions defined as computable under the Church-Turing thesis [1]. However, he goes on to demonstrate that even with this restriction, such a quantum Turing machine would be capable of simulating classical computation, and of simulating a much broader class of physical systems than a classical device [1].

In this paper, Deutsch also set out an algorithm which demonstrates the ability of a quantum computer to perform tasks faster than any classical device. In particular, he constructs a problem where we are given access to an 'oracle', which evaluates a function $f : \mathbb{F}_2^2 \to \{0, 1\}$ and returns the result, where $\mathbb{F}_2^n$ is the space of n-bit binary strings.
We are tasked with determining if the function is equal for all inputs ('constant'), or if it returns 0 for half of the inputs and 1 for the other ('balanced'), in as few evaluations or 'orcale queries' as possible. An extension of this problem to n-bit functions $f : \mathbb{Z}_2^n \to \mathbb{Z}_2$, called the Deutsch-Jozsa algorithm, was proposed in 1992, and more dramatically illustrates the computational advantage of the quantum Turing machine [14].

To answer this question deterministically, a classical computer would need to evaluate f for $\frac{2^n}{2} + 1$ inputs, half of the possible set. However, it can be shown that by preparing a quantum computer in a superposition of all n-bit binary strings $|\psi\rangle = \frac{1}{\sqrt{2}^n} \sum_{x \in \mathbb{F}_2^n} |x\rangle$, it suffices to evaluate f once. Deutsch dubbed this behaviour 'quantum parallelism', as it relies both on the ability of quantum states to exist in superposition, and for the existence of relative phase between states such that interference between then results in a deterministic answer.

2.2 The Nature of Quantum Speedup

The Deutsch-Jozsa algorithm demonstrates a dramatic reduction in query complexity, but this is relative to a classical algorithm sequentially testing inputs to be able to give a deterministic answer. In reality, a classical computer testing random inputs would be able to answer the Deutsch-Jozsa problem with a certain probability in much less than $\frac{2^n}{2}$ trials.
In classical computational complexity theory, problems of this type belong to a class called 'Bounded-error Probability Polynomial' (**BPP**); the class of problems where an efficient algorithm[1] including a randomised process can return the correct answer with an error that is bounded by a constant, independent of the problem size [15]. The quantum analogue of this class, Bounded-error Quantum Polynomial (**BQP**), is considered the class of problems efficiently solvable on a quantum computer [16].

The term 'polynomial' in the above definitions refers to the 'time' taken to complete the computation, as a function of the number of bits in the problem n. In the Turing machine picture, this is the number of steps in the computation, but it can equivalently be defined as the number of logic gates acting on n binary bits in a boolean circuit. In 1993, Yao demonstrated that an equivalent 'circuit' model exists for quantum computing, and that any quantum circuit is capable of simulating a quantum Turing machine [17]. This circuit model, where elementary unitary matrices are acted in succession on n-qubits to implement a computation U, is the framework used to develop and analyse quantum algorithms.
The use of query complexity might seem to be 'hiding' the true computational complexity of the problem: what if the oracle requires a long time to evaluate? However, in this and other problems the oracle is evaluating a classical boolean function f, which thus has an efficient implementation as as quantum circuit [16].

Following the work of Deutsch and Jozsa, the question of quantum speedup became: do there exist problems which are efficient for a quantum computer to solve, but which are 'harder' than **BPP** for a classical device?

2.2.1 Beating BPP

This question was answered in a series of three papers in 1997, which first established the existence of, and then developed explicit quantum algorithms for, problems in **BQP** which for a classical computer require a computational time that scales exponentially in the size of the problem [15].

The existence of this exponential quantum speedup over classical computation was first proved by Bernstein and Vazirani in their paper on quantum computational complexity theory [16]. As well as demonstrating the possibility of exponential speedup, this paper formalised Deutsch's quantum Turing machine construction, and used it to prove a number of properties about **BQP**, In particular, they demonstrated that $\mathbf{BPP} \subseteq \mathbf{BQP}$; that a quantum computer is capable of efficiently simulating efficient classical algorithms, including those assisted by random number generation.

The first explicit construction of such a problem was given by Simon in 1994, again using an 'oracle' formulation of the computation [18]. In Simon's problem, the oracle implements a binary function $f : \mathbb{F}_2^n \to \mathbb{F}_2^n$, such that for two binary strings $x, y \in \mathbb{F}_2^n$, $f(x) = f(y) \Leftrightarrow x = y \oplus s$, for some constant string s. We are tasked with finding the string s in as few queries as possible. For a quantum computer, this algorithm requires only $O(n)$ oracle queries. In contrast, a classical method would require $O\left(2^{\frac{n}{2}}\right)$ [18].

[1] 'Efficient' problems are generally defined as those with a running time polynomial in the problem size. These are the classes **P** (deterministic Polynomial), and **BPP** for classical computing, and **BQP** for quantum computing.

Simon's algorithm is another example of a 'toy' problem, but the techniques presented were then applied by Peter Shor to develop a general quantum algorithm capable of solving two mathematical problems: Prime Factorisation and the Discrete Logarithm [18, 19]. In complexity theoretic terms, they belong to the class **NP**, problems which require an exponential amount of time to solve, but for which a proposed solution can be checked efficiently [15].

It is believed that a polynomial-time classical algorithm for both problems does not exist. In face, the presumed computational hardness of prime factorisation is what guarantees the security of RSA and other cryptographic protocols currently relied upon for secure encryption [20]. The discovery of an efficient quantum algorithm spurred a great deal of interest in quantum computing, as it raised the possibility of devices capable of breaking most if not all of modern cryptography [19].

The existence of problems that reside in the class **BQP** but for which no efficient classical algorithm exists, further clarifies the idea of Feynman and Deutsch that a classical computer cannot efficiently simulate a quantum system. Any efficient simulation of arbitrary quantum systems would be also be able to implement quantum computations, and would in turn give us efficient classical algorithms for problems like prime factorisation [19].

2.2.2 NP-Complete Problems

While prime factorisation and the discrete logarithm are difficult to solve classically, they do not belong to the so called **NP**-complete problems; problems in the class **NP** for which *any other* problem in **NP** can be encoded as an instance of the problem [21]. As a result of this 'completeness' property, the **NP**-complete problems are considered some of the most import in computer science [15].

In 1996, Lov Grover developed a quantum algorithm capable of solving arbitrary **NP** problems, including the **NP**-complete problems, using the polynomial 'verifier' of a candidate solution implemented as an oracle [22]. This algorithm is commonly referred to as 'quantum search', as it was developed as a technique for searching for a single item in a set of $N = 2^n$ entries. A classical, brute-force search would, in the worst case, require testing all N items, an in general requires testing $O(N)$.

Grover's quantum algorithm, however, can succeed in $O(\sqrt{N})$ trials, a more modest quadratic reduction in the oracle complexity. The techniques employed are very similar to the Deutsch-Jozsa algorithm, applying the oracle to a uniform superposition of N inputs to 'mark' the solution. This is followed by the application of a 'diffusion' operator, which boosts the amplitude of the marked state. THis pair of operations is called the 'Grover step', and the amplitude of the state corresponding to the solution is maximal after $O(\sqrt{N})$ such steps [22].

However, such a brute-force search is likely not optimal for realistic database searching. For example, if the data can be sorted based on a certain criteria, then a 'bracketing and bisection' technique based on testing random entries in the sequence can succeed in $O(\log_2(N))$ trials. However, this kind of brute force problem does arise when considering **NP**-complete problems, such as optimisation problems. Classical algorithms for such problems often use heuristics to try and aid the search through the space of possible solutions.

In this case, Grover's algorithm allows a quantum computer to achieve a quadratic speedup for any **NP**-complete problem. Shortly afterwards, it was shown by Bennett et al. that Grover's quadratic speedup is likely optimal for any quantum algorithm tackling **NP**-complete problems [15].

These results have demonstrated the nature of quantum speedup. Quantum computing seems provably stronger than classical computation when it comes to simulation of physical systems, as it satisfies the Church-Turing-Deutsch thesis. A quantum computer can simulate classical algorithms,

as **BPP** $\subseteq$ **BQP**. It has also been proven that there exist problems believed to be in **NP**, which reside in **BQP** for a quantum computer. However, it is believed that quantum computers cannot offer a greater than quadratic speedup for the **NP**-complete problems.

2.3 What is the Origin of Quantum Speedup?

Given the existence of quantum speedup, it is interesting to ask what aspects of quantum mechanics underlie this effect. This is not only an interesting theoretical question, but also relevant to our attempts to realise quantum technologies. Control of quantum systems, including isolating them from the effects of decoherence, represents a significant challenge [19]. Understanding the origin of quantum speedup thus helps us to understand what criteria our experimental devices will need to satisfy.

In section 2.1, we introduced the notion of 'quantum parallelism', the ability of a quantum system to exist in superpositions of multiple states and an oracle function to be evaluated simultaneously across all inputs. Holevo's bound tells us, however, that while it is possible for the system to be in a superposition of 2^n states, we can only access classical information of 1 of these states at a time [23]. As such, we also require interference effects to be present in the computation, such that the system converges to a solution.
However, in a 1997 paper, Jozsa discusses the fact that superposition and interference are both effects demonstrated by classical waves [12]. Instead, Jozsa argues that the critical feature of quantum mechanics is the existence of entangled states.

2.3.1 Entanglement and Quantum Computing

A general quantum state on n-qubits, $|\psi\rangle$, exists in the Hilbert space $\mathcal{H}_{2^n}$. If this state can be decomposed into a tensor product of states on smaller Hilbert spaces, then it is called a 'separable' or 'product' state. If no such decomposition exists, the state is said to be entangled [20]. This definition is exact for pure states, though entanglement for mixed states is much less well characterised [24].

In his paper, Jozsa considers the example of a classical system with 2^n states, constructed out of n strings each in a superposition of their two lowest-energy vibrational modes. However, composite classical systems combine under the Cartesian product [12]

$$A \times B = \{(a, b) : a \in A, b \in B\} \implies |A \times B| = |A| + |B| \tag{2.1}$$

in contrast to quantum systems which use the tensor product [20]

$$A \otimes B = \{(ab) : a \in A, b \in B\} \implies |A \otimes B| = |A| \cdot |B| \,. \tag{2.2}$$

As a result, this physical system remains a separable product of n individual subsystems. The phenomenon of entanglement, where a physical system cannot be described as a combination of smaller subsystems, is not possible classically [12].
This argument was extended in a later paper in collaboration with Ekert [25]. In this paper, they also consider the possibility of constructing a classical simulator based on a superposition of 2^n vibrational modes, which would exploit an isomorphism between $\otimes^n \mathcal{H}_2$, the Hilbert space n qubits, and $\mathcal{H}_{2^n}$, the Hilbert space of one, 2^n-level quantum system. In this picture, the 'isomorphism' would allow us to label certain states of the classical simulator as entangled [25].
However, the authors demonstrate that such a classical simulator cannot be implemented without

an exponential overhead in some physical resource, and thus cannot be considered realistic. For example, a system of n-qubits, each with the same 'energy gap', would require an amount of energy to control that grows linearly in n. However, for the 2^n level classical simulator, the number of energy levels and thus the energy required to implement it grows exponentially in n [25].
A similar argument applies if we instead consider a different classical simulator where the 2^n levels accumulate below a finite upper bound [2]. In this case, the precision required to resolve these levels grows exponentially in n and the simulator is not physically feasible. Thus, the authors argue, the phenomenon of quantum speedup is due precisely due to the existence of entangled states.

Jozsa and Ekert also indicate the presence of entanglement in existing descriptions of quantum algorithms. For example, when an oracle gate, O_f, is applied to a superposition of input strings $\frac{1}{2^n}\sum_{x\in\mathbb{F}_2^n}|x\rangle$, the resulting state is an entangled state of each input with its evaluation $\sum_{x\in\mathbb{F}_2^n}|x\rangle|f(x)\rangle$ [12]. Ekert and Jozsa also demonstrate the existence of entanglement in the quantum Fourier transform, a subroutine employed in Simon's and Shor's algorithms [25].

Impact on Experimental Efforts

This identification of entanglement as a the key resource for quantum speedup had a significant impact on the quantum computing community. In particular, it called into question a series of early quantum computing experiments realised using NMR systems [26, 27] It was shown by Braunstein et al. that, for sufficiently high levels of noise, the 'pseudo-pure' states used in NMR computing would remain separable throughout the computation [26].
This analysis in fact demonstrated that all previous NMR experiments had systematic noise above this threshold. A subsequent paper by Linden and Popsecu explicitly demonstrated that executions of Shor's algorithm requires the system to become entangled, again ruling out previous NMR results [27]. A similar result was also shown for Grover's algorithm [28].
This demonstrates the ability of theoretical studies into the foundations of quantum theory to influence the course of quantum technologies.

2.3.2 Is Entanglement Necessary?

When trying to understand quantum speedup, it can also be instructive to consider the classical information required to fully describe a quantum system. Any quantum dynamics which can be efficiently simulated classically cannot, by definition, realise quantum speedup. The inverse statement, that a lack of efficient classical simulation implies quantum speedup, is not necessarily true, as it assumes no better classical simulation method exists. Nonetheless, an exponential classical complexity is considered indicative of quantum advantage [2].

If we consider the classical information required to fully characterise a quantum state, then a product state of n qubits $|\psi\rangle = \bigotimes_{i=1}^{j}|\phi\rangle_j$, can be completely described with $O(j)$ complex numbers [12]. In contrast, entangled states requires $O(2^n)$ numbers to describe the system completely [12].
This exponential classical description as a result of entanglement can also be applied to descriptions of quantum circuits. A given computation on n qubits involves a $2^n \times 2^n$ unitary matrix U. However, by decomposing this unitary into a sequence of elementary gates acting on one or two qubits, we can reduce this complexity to $O(\text{poly}(n))$ [25]. The overall complexity of the computation is then polynomial in n if the state can be described as a product state at all steps [25]

Jozsa & Linden used this observation to give a slightly more restricted definition of entanglement as a resource for quantum computation [2]. In particular, it is not sufficient for only a small, bounded subset of the qubits to become entangled. The authors define the notion of a

'p-blocked state', which can be split into subsystems of at most p qubits, and show that a quantum computation is efficiently simulable if at all steps there exists such a decomposition [2]. Instead, the entanglement in the system must be unbounded, growing to include all qubits present in the computation. They then explicitly demonstrate that, for Shor's algorithm, the entanglement grows in precisely this unbounded way [2].

Efficient Simulation of Entangled States

Interestingly, they authors also show that we can approximately simulate our quantum computation to within an error η, provided there exists at all steps a p-blocked state ϵ-close to the 'true' state of the computation where $\epsilon = c^T\eta$, T is the number of steps in the circuit, and c is a constant less than one [2].
This pair of results significantly constrain the quantum advantage offered by entanglement alone. A similar result was found by Vidal, who developed a novel method for simulating 1D-spin chains with 'restricted' entanglement. In particular, he showed that if the entanglement measure $E_\chi \equiv \log_2(\chi)$, where χ is the Schmidt-Rank of the state, grows only logarithmically in the system size, then it admits an efficient classical decomposition [29].

Jozsa & Ekert's conclusions about entanglement were challenged by Laflamme et al., in response to Braunstein's argument against NMR quantum computing on the grounds of separability [24]. Laflamme argued that, whereas the non-local behaviour of entangled systems gives a clear advantage in quantum communication protocols, the significance of entanglement to quantum computing is much less clear. In particular, they argued that the apparent significance of entanglement could be explained by the computational description employed by Jozsa and Ekert, in terms of the 2^n complex amplitudes of the n-qubit state [24].
As a counter example, Laflamme cites the 'Heisenberg picture' of quantum computation developed by Gottesman & Knill. The Heisenberg picture is a construction of quantum mechanics where time-evolution is carried by the operators rather than by the quantum states [20]. The Gottesman & Knill construction similarly allows the quantum circuits acting on a special subset of quantum states to be described in terms of their effect on groups of Pauli operators alone [30].

Originally developed in the context of quantum error correction codes, Gottesman's formalism is predicated around groups of operators $\mathcal{S} \subseteq \mathcal{P}_n$, subgroups of the n-qubit Pauli group. Gottesman showed that there exists subgroups on n qubits $\mathcal{S} : |\mathcal{S}| = 2^n$, which have a unique $+1$ eigenstate $|\phi\rangle : s\,|\phi\rangle = |\phi\rangle\ \forall s \in \mathcal{S}$. These eigenstates are called 'stabilizer' states, as they are stabilised by the group $\mathcal{S}$.
The action of these stabilizer groups can be entirely described by its generating set, a subset of independent elements which can be multiplied together to produce all other elements in the group. Because the stabilizer operators have a common eigenstate, the group is abelian, meaning it's 2^n elements can be described using just n generators. The action of a unitary U acting on a stabilizer state can then be described by evolving the stabilizer generators $s_i : i \in \{1, \cdots, n\}$ under the relation $s_i' = UsU^\dagger$.
This allows the stabilizer states to be fully characterised using n sets of n Pauli operators. Examples of stabilizer states include the computational basis states, but it also includes a number of entangled states. For example, the maximally entangled Bell state $|\Psi\rangle_+ = \frac{1}{\sqrt{2}}(|01\rangle + |10\rangle)$, is stabilised by the generators $X \otimes \mathbb{I}$ and $\mathbb{I} \otimes X$. Interestingly, this state demonstrates 'unbounded' two-qubit entanglement.

For a general computation U, these stabilizer generators are $2^n \times 2^n$ matrices, so this description remains classically complex. However, for a large number of elementary quantum gates, including

the 'entangling' CNOT gate, their action on the stabilizer states can be efficiently computed classically [30].
These operations are commonly referred to as the Clifford group, defined as the matrices [30]

$$U : UPU^{\dagger} \in \mathcal{P}_n \; \forall P \in \mathcal{P}_n. \tag{2.3}$$

As they only map Pauli operators to other Pauli operators, their action on the stabilizer states is simply given by appropriately updating the Pauli operators that generate each group.
While this is a restricted set of operations and states, the stabilizer formalism is nonetheless capable of describing entangled states. It can also be used to simulate intrinsically 'quantum' protocols, such as quantum teleportation, efficiently [4].

Laflamme thus argued that apparent requirement of entanglement for quantum speedup could be perhaps understood as a consequence of the amplitude formalism that Jozsa, Ekert, Linden and Popescu used to describe quantum states and their simulation [24]. Instead, given the efficient simulability of Clifford circuits, and that mixed-state entanglement is not well understood, Laflamme proposed that it is the full unitary dynamics that gives a quantum computer its advantage [24].

Jozsa & Linden acknowledge Laflamme's argument in the conclusion to their paper, agreeing that this apparent requirement for unbounded entanglement derives from their chosen classical description of the quantum states, and that different descriptions of quantum mechanics would have different sets of efficiently-described states.

The debate over the role of entanglement in computation continues today. For example, a recent result that could be seen as both supporting Laflamme's & Jozsa's arguments is the demonstration by Van den Nest of universal quantum computation based only on states with a small amount of entanglement under the 'Entanglement Entropy' measure [31]. Interestingly, even with access to this limited amount of entanglement, he demonstrated that these states can realise arbitrary unitary dynamics, and such be used to realise universal quantum computation. This could be interpreted as demonstrating the power of entanglement alone, or the role of unitary dynamics.

This debate illustrates the difficulty of demonstrating the origins of quantum advantage; the absence of an efficient classical description is not a clear indicator of quantum speedup, as this description depends on the formalism chosen. For each description $\mathcal{D}$, there exists a corresponding property $\text{prop}\,(\mathcal{D})$ that is required for quantum speedup [2]. In some sense, we can argue that a system demonstrating quantum speedup must satisfy $\text{prop}(\mathcal{D}) \; \forall \; \mathcal{D}$ [2, 24]. However, the existence of an efficient classical method is a clear signifier of the absence of quantum advantage, and so classical simulations of quantum computation are still an effective way to study the 'power' of a quantum computation.

2.4 Fault Tolerant Quantum Computing

Despite continued improvements in qubit control, quantum computation still has to contend with the effects of decoherence and environmental noise [20]. This has lead to the notion of fault tolerant schemes, designed to protect the quantum information throughout the computation [20].

In a fault tolerant quantum computer, the information is encoded in multiple physical qubits using a quantum error correction code. These codes can also be described using Gottesman's stabilizer formalism; the 2^m basis states of the codespace are the $+1$ eigenstates of a stabilizer group with $k = n - m$ generators, acting on n qubits. The computation is then performed using 'logical' gates that act on the encoded states [32].
If an error occurs on one of the physical bits during a logical gate, it can be detected by a -1

outcome when measuring the stabilizer generators [32]. By measuring the generators at each step in the computation, we can identify and correct any errors that occur [20].
However, this also requires that we construct our 'logical' gates in a fault tolerant way. If the logical gates induce multiple errors on each encoded qubit, then we are no longer able to deterministically detect and correct them [33]. Instead, we typically require that our logical gates be 'transversal': that they require no multi-qubit gates within a single code block. This prevents an error on one physical bit from spreading to other qubits, reducing the probability of our code failing [20, 33].

Unfortunately, it is known that no stabilizer error-correction code can have a set of transversal gates that is also universal for quantum computation [34]. In fact, most stabilizer codes admit only Clifford group operations as their transversal gate set. As the states of the code are entirely characterised by a stabilizer code, this means that not only are these codes not-universal, but their action can be efficiently simulated classically. It was in fact shown by Aaronson & Gottesman that these Clifford circuits are strictly *weaker* than a classical computer [4]. We thus need some way to boost our Clifford circuits to universality, and to do so in a fault tolerant way.

An important concept in the theory of universal computational gate sets is the 'Clifford Hierarchy', a recursive family of unitary operations defined by their actions on the Pauli operators [35]. We define level 1 of the hierarchy, $\mathcal{C}_1$, as the Pauli operators themselves. Level 2 then corresponds to the Clifford group which, as defined in Eq. 2.3, maps Pauli operators to other Pauli operators. This can alternatively be written

$$\mathcal{C}_2 \equiv \{U\} : UPU^\dagger \in \mathcal{C}_1 \; \forall P \in \mathcal{P}. \tag{2.4}$$

Thus, level 2 of the hierarchy has the property that it maps all Pauli operators to operators in level 1, the set of Pauli operators. Level 3 is then defined as the set of operators than map Pauli operators to Clifford group operations. We can thus extend this to define level n as

$$\mathcal{C}_n \equiv \{U\} : UPU^\dagger \in \mathcal{C}_{n-1} \; \forall P \in \mathcal{P} \tag{2.5}$$

It has been shown that a Clifford circuit can be boosted to universal quantum computation if we add just a single gate from $\mathcal{C}_{n\geq 3}$. Most commonly, fault tolerant techniques have focused on implementing gates from $\mathcal{C}_3$, such as the T gate

$$T \equiv \begin{pmatrix} 1 & 0 \\ 0 & \mathrm{e}^{1\frac{\pi}{4}} \end{pmatrix}$$

and the 'Toffoli' or 'Controlled-Controlled NOT' gate [19, 35].

2.4.1 'Magic' State Injection

A fault tolerant T gate can be realised using a '1-bit teleportation' or state-injection circuit, developed by Chuang et al [36]. This simplified teleportation circuit made up of an ancillae qubit $H|0\rangle = |+\rangle$, Clifford gates, and an additional measurement-controlled correction.
By using an appropriately prepared ancilla, this circuit can be used to realise the state $U|\psi\rangle$. To find the required state, we need to commute the operator U through the circuit, which can

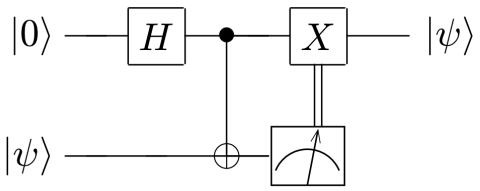

Figure 2.1: Circuit for '1-bit teleportation', taken from [36].

be done for any diagonal unitary $U \in \mathcal{C}_n \forall n \geq 3$ [36]. This has the affect of changing the required correction operation $X \to UXU^\dagger$, and changing the ancilla $|+\rangle \to U|+\rangle$ [36].
For the T gate, which satisfies the diagonal requirement, this requires preparing an ancilla $|A\rangle = |0\rangle + e^{i\frac{\pi}{4}}|1\rangle$. Because the state injection circuit is built out of Clifford gates and measurements, the problem of a fault tolerant implementation then reduces to preparing these ancilla states.

A scheme for preparing these ancillae was proposed by Bravyi & Kitaev, allowing an arbitrarily pure state to be prepared from multiple, imperfectly prepared copies [37]. This method can be used to prepare $|H\rangle = \cos(\frac{\pi}{8})|0\rangle + \sin(\frac{\pi}{8})|1\rangle$, and a family of other states that are equivalent to $|A\rangle$ up to a Clifford group operation [37].
Because they can boost a stabilizer code to universality, these states are dubbed 'magic states' [37]. Bravyi & Kitaev also derived a threshold fidelity for the initial noisy magic states, for the distillation procedure to be able succeed, which corresponds to the edges of an octahedron in the Bloch Sphere defined by the single qubit stabilizer states [37]. $|A\rangle$ and other T-gate magic states are pure states that project out from the edges of the octahedron, which is shown in Fig. 2.2.
An alternative type of magic state, which we will dub $|F\rangle$ for 'face'-state, lies at the centre of the faces of this octahedron. This is also shown in Fig. 2.2. However, because there is currently no known deterministic procedure for using face-type states in state injection, fault tolerant computing schemes focus on using circuits built out of Clifford gates and T gates.

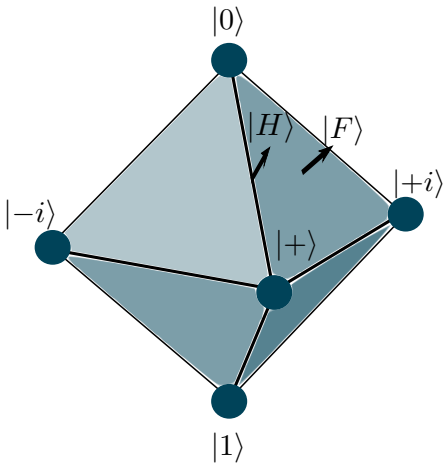

Figure 2.2: Representation of the set of classically simulable states in the Bloch Sphere, showing the edge and face-type magic states.

Simulating Clifford+T Circuits

The octahedron shown in Fig. 2.2 corresponds to convex mixtures of stabilizer states, and as such it also defines the set of efficiently simulable single-qubit states [3]. The single qubit Clifford group operations correspond to symmetric rotations of the octahedron, which illustrates the classical simulability of Clifford circuits on stabilizer states.

An efficient algorithm for classically simulating these Clifford circuits, called 'CHP'[2], was developed by Aaronson & Gottesman. Using the fact that the Pauli group can be generated by

[2]Named for the generators of the n-qubit Clifford group: CNOT, Hadamard and Phase

$\pm i, X$ and Z, and that stabilizer generators must must have real-valued phase as the group cannot contain the element $-\mathbb{I}$, they represent each stabilizer on n qubits as a $2n+1$ bit binary string. Pairs of the first $2n$ bits describe the Pauli operator on each of the n qubits: `00` corresponds to $\mathbb{I}$, `01` to Z, `10` to X and `11` to Y. The final bit r gives the phase of -1^r.
These n stabilizer bit strings are stored in a $2n \times (2n+1)$ tableau. The other n rows are the 'destabilizers', a dual set of operators that, together with the stabilizers, generate the full n-qubit Pauli group. The `CHP` method is then capable simulating arbitrary Clifford group operations and Pauli measurements on n qubits in a time $\mathcal{O}(n^3)$, by updating the tableau appropriately.

In the paper, Aaronson & Gottesman also discuss extensions of the CHP method to the case where a quantum computer is furnished with b non-stabilizer ancillae, d of which are subject to Pauli measurements. In this case, the `CHP` method acquires an exponential scaling $\mathcal{O}\left(2^{2b+2d}\right)$ [4]. As a Clifford+T computing scheme would consume all magic states, this implies $b = d \equiv t$, giving an effective scaling of $\mathcal{O}(2^{4t})$, where t is the total 't count' of the circuit [4, 6]. Interestingly, this method performs lightly better if the formalism is adapted to describe non-Clifford gates, giving a scaling $\mathcal{O}\left(4^{bd}\right)$ for b gates acting on at most d qubits. In the case of t T-gates acting on single qubits, we obtain a scaling $\mathcal{O}(4^t)$ [4].

This exponential dependence on the T-count is expected; adding appropriate non-Clifford gates (either directly or with ancillae) promotes our quantum computation to universality, and thus it should no-longer be classically simulable.

Contextuality

Most recently, a property of quantum mechanics called contextuality has emerged as an alternative requirement for quantum advantage. Contextuality in fact seems to correspond to a property, $\text{prop}\,(\mathcal{D})$, required for quantum speedup in the 'Clifford+T' picture of quantum computing.

Contextuality is understood as a generalisation of non-locality, which derives from a theorem by Bell, Kochen & Specker demonstrating that a measurement outcome in quantum mechanics cannot be understood as revealing a 'hidden variable' that describes the state of the system [3]. What makes this interesting as a candidate for quantum advantage is that it is a feature that cannot be replicated in classical models. For example, Spekken's toy model, an classical model with an 'epistemic' restriction preventing any more than half of the 'hidden variables' being known, is capable of describing other 'quantum' features like entanglement, but cannot demonstrate contextuality [38].

It has been demonstrated that the onset of contextuality for qudits, d-dimensional elements where $d > 2$, corresponds exactly with leaving the set of efficiently simulable states. For qudits, this set is slightly larger than the set of stabilizer states, but any state outside this set is sufficient to promote a Clifford circuit to universality [3].
However, for qubits, the case is slightly more complicated. The set of simulable states in this case coincides exactly with the set of stabilizer states, but it has in fact been shown that qubits demonstrate 'state-independent' contextuality: even stabilizer states will violate 'contextuality' inequalities [3]. However, it has nonetheless been shown that magic states give larger violations of contextuality inequalities than arbitrary quantum states, and that this is largest for the face-type states $|F\rangle$ [3].

Chapter 3

The Bravyi, Gosset, Smolin, Smith Technique

As discussed in Chapter 1, a significant caveat when using classical simulations to study quantum speedup is that the complexity of a simulation in a given framework does not exclude a more efficient description in an alternative picture. In general for a system with arbitrary states and arbitrary operations, no efficient classical implementation should exist under the Church-Turing-Deutsch thesis [1]; it is nonetheless instructive to consider the computational complexity of different quantum computing methods .

An interesting example of this is the development of a novel technique we will call `BGSS` or the 'stabilizer rank' method, for simulating Clifford+T quantum circuits, developed in a pair of papers by Bravyi, Gosset, Smolin & Smith [6, 5]. In particular, this algorithm is capable of sampling the output distribution of a computation made on n qubits with t T gates in a time that scales as $O(2^{0.23t}t^3)$ [5]. This represents a significant reduction in computational complexity over the `CHP` method of Aaronson & Gottesman, which scales as $O(2^{4t})$ [4].

The core of this `BGSS` algorithm lies in two results. The first is that any circuit on n qubits with t T-gates can be rewritten as a sequence of Pauli measurements acting on t edge-type magic states. This is a new model of quantum computation called a 'Pauli Based Computation'(PBC) [6].
The second result is that it is possible to find decompositions of n qubit states states into a sum of non-orthogonal n qubit stabilizer states. The number of states in this decomposition is called the stabilizer rank χ, and $\chi \leq 2^{\frac{n}{2}}$ for edge-type magic states [5, 6].

In this Chapter, we discuss the proof of these two techniques, and in particular focus on proving the asymptotic limit for the stabilizer rank of the edge type magic states. We then discuss the significance of this result for quantum computation. The details of the computational implementation are not discussed in this report, as our focus is on the interpretation of the stabilizer rank itself.

3.1 Pauli Based Computation

A PBC is defined by a sequence of m Pauli measurements on t qubits $P_i : i \in \{1, 2, \cdots, m\}$. As these are Pauli measurements, at each step we obtain an outcome $\sigma_i = \pm 1$. We allow the choice of Pauli operator to be 'adaptive', such that the measurement performed at step j can depend on all the previous outcomes $\{\sigma_1, \cdots, \sigma_{j-1}\}$. The final sequence of measurement outcomes is then processed classically to give the result of the computation.
Bravyi, Smolin and Smith showed that, if the t qubits used in the PBC are initialised as the magic

state $|A\rangle$, then this PBC is capable of simulating a Clifford+T circuit with t T-gates. The other components of the circuit determine the sequence of Pauli measurements [6].

We can begin to understand this correspondence by defining a looser model called a PBC*, where a subset of the qubits in the computation are initialised in the computational state $|0\rangle$, and the rest are initialised as magic states $|A\rangle$. We consider a computation U, made up of c Clifford and t T gates on n qubits, followed by a measurement in the computational basis. We can convert this to something resembling a PBC* by replacing each T gate with a magic state injection gadget, as shown in Fig. 2.1.
We thus define the new 'gadgetized' circuit V acting on $n+t$ qubits, which is made up of only Clifford gates, X basis measurements in the magic state gadgets and a final Z basis measurement on n qubits. As the Clifford group only permutes operators in the Pauli group, we can thus commute the entire circuit V to the end of the computation, after the final measurement, and discard these gates, updating the measurement operators accordingly [6].
Because of the measurement-controlled correction in the magic state gadgets, the resulting Pauli measurements on the computational states will be adaptive. This gives a sequence of t 1-qubit measurements, followed by a readout measurement on n qubits; we have successfully converted the computation to PBC* form.

We can assume without loss of generality that all these Pauli operators pairwise commute [6]. To understand why, consider step q, which anticommutes with a previous measurement P_p. Writing the state of the system after the $q-1$ measurements as $|\phi\rangle$, the state obtained after measurement q is $\frac{1}{\sqrt{2}}(\mathbb{I}+\sigma_q P_q)|\phi\rangle$.
We can rewrite this projector as $W_q = \frac{1}{\sqrt{2}}(\sigma_q P_q + \sigma_p P_p)$. For an anticommuting pair of operators P_q, P_p, the resulting operator $W \in \mathcal{C}_2$, and thus it suffices to pick σ_t at random, commute the resulting W_q to the end of the circuit, and discard it.
We can use this method to prove that the action of the Pauli measurements on the qubits initialised in the $|0\rangle$ state is trivial [6]. If we prepend the measurement sequence with computational basis measurements on these $|0\rangle$ qubits, then these measurements will all have deterministic outcomes $+1$. We can use the above argument to make these measurements commute with the sequence P_i, such that all these Pauli measurements act trivially, deterministically, on these n qubits. Thus, we can discard them, and obtain a PBC on t qubits [6].

Finding the PBC Projector

We can use this PBC formalism to obtain an explicit form for the probability $P_{out}(x)$ of obtaining a given output string x from the set of all w-bit binary strings $\mathbb{F}_2^w$, where $w \leq n$. The projector on to this output is given by $\Pi_x = |x\rangle\langle x| \otimes \mathbb{I}_{else}$. To simplify the analysis, we can postselect on the measurement outcome of the magic state gadgets, such that we don't have to introduce any correction operations. To compensate, we normalise the probabilities accordingly [6]. This gives the expression

$$P_{out}(x) = 2^t \left\langle 0^{\otimes n} A^{\otimes t} \middle| V^\dagger \left(\Pi_x \otimes \left|0^{\otimes t}\middle\rangle\middle\langle 0^{\otimes t}\right| \right) V \middle| 0^{\otimes n} A^{\otimes t} \right\rangle \tag{3.1}$$

where V is the circuit including magic state gadgets on the $n+t$ qubits. The projector $\Pi_x \otimes |0^{\otimes t}\rangle\langle 0^{\otimes t}| \equiv \Pi$, is a projector on to a stabilizer group $\mathcal{W} \subseteq \mathcal{P}_{n+t}$, generated by $-1^{x_i} Z_i$ for the ith bit of the output string x_i, Z_j for the jth magic state ancilla, and $\mathbb{I}$ otherwise. This group thus has $w+t$ generators.
The action of the Clifford circuit V maps us to a new stabilizer group $\mathcal{V}$. For any element of $\mathcal{V}$, if

the stabilizer doesn't act as I or Z on the first n qubits, then this term reduces to 0 as

$$\langle 0|\mathrm{X}|0\rangle = \langle 0|\mathrm{Y}|0\rangle = 0.$$

Thus, the matrix element is non-zero only on a subset $\mathcal{V}_0$ with v generators, and thus we can write

$$\langle 0^{\otimes n}|\Pi_{\mathcal{V}}|0^{\otimes n}\rangle = 2^{-w-t+v}\,\langle 0^{\otimes n}|\Pi_{\mathcal{V}_0}|0^{\otimes n}\rangle \tag{3.2}$$

with a normalisation factor $2^{-(w+t-v)}$ [5].
Evaluating the matrix element for the $|0^{\otimes n}\rangle$ states gives a reduced t qubit stabilizer group $\Pi_{\mathcal{G}}$ acting on the magic states. Taken all together, we thus have

$$\begin{aligned} P_{out}(x) &= 2^{v-w-t+t}\left\langle A^{\otimes t}\middle|\Pi_G\middle|A^{\otimes t}\right\rangle \\ &= 2^{v-w}\left\langle A^{\otimes t}\middle|\Pi_G\middle|A^{\otimes t}\right\rangle \end{aligned} \tag{3.3}$$

where $\Pi_G = \langle 0^{\otimes n}|V^\dagger \Pi V|0^{\otimes n}\rangle$ [5].

3.2 The Stabilizer Rank

Having proven this alternate form for a Clifford+T circuit, Bravyi, Gosset, Smolin & Smith then examined the problem of trying to efficiently simulate the PBC. In particular, they use the observation that a Pauli measurement on a stabilizer state can be simulated in a computational time that scales as $O(n^3)$ [4, 6].

We could then use this observation to try and simulate Pauli measurements on arbitrary states by decomposing them into a mixture of stabilizer states

$$|\psi\rangle = \sum_{i=1}^{\chi} z_i\,|\phi_i\rangle \quad : \exists\ \mathcal{S}_{\phi_i} \tag{3.4}$$

We can calculate the matrix element $\langle\phi_i|\Pi_G|\phi_i\rangle$ efficiently for each stabilizer state, and then combine them according to the amplitudes z_i. The computational complexity of this then scales as $O(\chi\ \mathrm{poly}(n))$, where χ is called the 'stabilizer rank' of the state, the number of terms in the decomposition. Stabilizer states also have the obvious property that $\chi = 1$, and χ for any n qubit state has a simple upper bound of 2^n, which corresponds to decomposing the state into the computational basis. But, certain states admit a significantly smaller stabilizer rank decomposition.

$\lvert H^{\otimes n}\rangle$	1	2	3	4	5	6
χ	2	2	3	4	6	7

Table 3.1: Table showing slow growth in the stabilizer rank for n copies of the magic state, as found in [6]. These values are estimates, as they were dervied by using Simulated Annealing to search for potential decompositions. In general, finding the stabilizer rank is computationally demanding as no known algorithm beyond brute-force searching exists.

In particular, Bravyi, Smolin & Smith noted that the stabilizer rank for $|H\rangle \otimes |H\rangle$[1] is 2; equal to the rank of an arbitrary single qubit state. This immediately constrains $\chi \leq 2^{\frac{n}{2}}$ for n edge-type magic states, by splitting the state into tensor products of pairs [6]. Numerical estimates on

[1]Recall that the state $|H\rangle$ is equivalent to the magic state $|A\rangle$ to within a Clifford rotation.

up to 6 magic states showed that the stabilizer rank grew slowly, approximately linearly, in the number of copies. A later analytical effort by Bravyi & Gosset demonstrated an asymptotic scaling $\chi(|H\rangle^{\otimes t}) = O(2^{0.23t})$ [5].

An important distinction in this method is that the stabilizer states in the decomposition do not have to be mutually orthogonal. This was a requirement in the extension of `CHP` to magic states developed by Aaronson & Gottesman [4, 39]. An alternative idea called 'stabilizer frame' decomposition was developed by Garcia et al., which built decompositions out of pairs of orthogonal stabilizer states [39, 40].
The authors successfully demonstrated that this method can achieve a 'frame rank' $|\mathcal{F}| < 2^k$. Using this method, they were able to simulate modular exponentiation circuits, which also form a subroutine in Shor's algorithm, achieving $|\mathcal{F}| = 64$ for a simulation on 5 logical qubits encoded in an error correction code [40]. However, they did not examine decompositions of magic states, and the pairwise frame construction means this was based on a simulation of $2 \times |\mathcal{F}|$ stabilizer states.

An alternative attempt to find a classical simulation of a Clifford+T simulation was based on the 'Discrete Wigner function' quasi-probability formalism, which allows efficient simulation for an state where the distribution is strictly positive valued [41, 3]. It was show that these can be extended to simulate general quantum circuits if they are combined with random sampling techniques, with a resulting running time exponential in the negativity of the Wigner function [6, 42]. Extending this to Clifford+T circuits allows a simulation of a 'restricted' PBC made up of only Pauli X and Z measurements, with a complexity $O(2^{0.543t}\text{poly}(n))$ [6, 43]. This is a similar scaling to the 'stabilizer rank' method, but importantly this restricted PBC is *not* capable of simulating universal quantum computation [6].

Bravyi, Smolin & Smith also make the following conjecture about the magnitude of the stabilizer rank.

Conjecture 1. For a given state $|\phi\rangle$

$$\chi_\phi \begin{cases} = 1 & \text{if } |\phi\rangle \text{ is a stabilizer states} \\ = \chi_n & \text{if } |\phi\rangle \text{ is a magic state} \\ > \chi_n & \text{otherwise} \end{cases}$$

where χ_n is the small stabilizer rank decomposition for an n-qubit magic state.

In particular, this would imply that a Clifford+'Magic State' computation seems to be the easiest model to simulate classically. This scaling is still exponential, but the growth of the computational description is smaller than for arbitrary quantum states [6].

3.2.1 Bounding χ for Edge States

In their 2016 paper building on the results of Bravyi, Smolin & Smith, Bravyi & Gosset sought an analytical form that would allow them to find a tight bound on χ for the edge-type states, giving a value of χ_n for the above conjecture. Importantly, they also allowed for the possibility of δ-approximate decompositions, such that $|\langle\psi|H\rangle|^2 \geq 1 - \delta$.

The edge-type state $|H\rangle$ is an eigenstate of the Hadamard operator, and is given by the projection onto the surface of the Bloch sphere of the midpoint along the edge of the set of simulable states connecting $|0\rangle$ and $|+\rangle$, as shown in Fig 2.2. As a result, the state has equal overlap with these two states

$$\langle 0|H\rangle = \langle +|H\rangle = \cos\left(\frac{\pi}{8}\right) \equiv \nu.$$

This means these two stabilizer states can form a 'basis' for the $|H\rangle$ state. Writing $|\tilde{0}\rangle = |0\rangle, |\tilde{1}\rangle = |+\rangle$, we can then write the state as a sum over binary strings in this basis

$$\left|H^{\otimes m}\right\rangle = \frac{1}{(2\nu)^t} \sum_{x \in \mathbb{F}_2^n} |\tilde{x}_1 \otimes \cdots \otimes \tilde{x}_t\rangle \tag{3.5}$$

where $\mathbb{F}_2^t$ is a finite field, equivalent to the set of t-bit binary strings [5]. Each of the terms in this decomposition is a tensor product of stabilizer states, and so is itself a stabilizer state. This decomposition then has a maximal stabilizer rank of 2^t for t-qubits.
The authors thus consider finding k-dimensional linear subspaces of t-bit strings, $\mathcal{L} \subseteq \mathbb{F}_2^t$. This can be understood is a subset of 2^k strings, such that the combination of any pair of strings in the set under addition (modulo 2) is also in the set. For each of these subspaces, we define an associated normalised state [5]

$$|\mathcal{L}\rangle = \frac{1}{\sqrt{2^k Z(\mathcal{L})}} \sum_{x \in \mathcal{L}} |\tilde{x}_1 \otimes \cdots \otimes \tilde{x}_t\rangle \tag{3.6}$$

where $Z(\mathcal{L})$ is a 'partition function' based on the Hamming weight of each string $|x|$ that ensures proper normalisation, given by [5]

$$Z(\mathcal{L}) \equiv \sum_{x \in \mathcal{L}} f(x) : \; f(x) = 2^{-\frac{|x|}{2}}. \tag{3.7}$$

Each of these normalised states has a stabilizer rank 2^k, and so the problem now reduces to understanding how small k can be to find a subspace capable of δ approximating $\left|H^{\otimes t}\right\rangle$.

From the overlap with each basis state, we can show that the overlap of the magic states with a subspace $|\mathcal{L}\rangle$ is given by

$$\left|\left\langle H^{\otimes t} \middle| \mathcal{L} \right\rangle\right|^2 = \frac{2^k \nu^{2t}}{Z(\mathcal{L})} \tag{3.8}$$

which, as $Z(\mathcal{L}) \geq 1$ gives a bound $2^k \geq \nu^{-2t}(1-\delta)$ for a δ approximate subspace. This can be reined to give the previously stated result

$$2^k = O(\nu^{-2t}\delta^{-1}) = O(2^{\gamma t}\delta^{-1}), \tag{3.9}$$

where the value $\gamma = 0.23$ comes from rearranging the two expressions to give $\gamma = -2\log_2(|\langle \tilde{x}|H\rangle|)$.

To find this bound, we need to bound the value of the partition function $Z(\mathcal{L})$ for an arbitrary subspace.

Lemma 3.2.1. *Let $\mathcal{L} \subseteq \mathbb{F}_2^t$ be a k dimensional subspace chosen uniformly at random. Then the expectation value of the partition function $\mathbb{E}(Z(\mathcal{L})) \leq 1 + 2^k \nu^{2t}$.*

Proof. [2]

$$Z = \sum_{x \in \mathcal{L}} 2^{-\frac{|x|}{2}}, \implies f(00\cdots0) = 1.$$

Thus, we can write

$$\mathbb{E}(Z(\mathcal{L})) = 1 + \sum_{x \in \mathbb{F}_2^t \setminus 0^t} f(x) \cdot \mathbb{E}\left(\chi_{\mathcal{L}(x)}\right)$$

[2] The proof described here is as presented in [5]. The argument will be given in more detail in section 4.1.

where $\chi_{\mathcal{L}}(x)$ is the 'indicator function', which returns 1 if $x \in \mathcal{L}$ and 0 otherwise.
We can thus show that

$$\begin{aligned}\mathbb{E}\left(Z(\mathcal{L})\right) &= 1 + \frac{2^k - 1}{2^t - 1} \sum_{x \in \mathbb{F}_2^t \setminus 0} 2^{-\frac{|x|}{2}} \\ &= 1 + \frac{2^k - 1}{2^t - 1} \left(2^t \nu^{2t} - 1\right) \\ &\leq 1 + 2^k \nu^{2t}\end{aligned}$$

□

Corollary. *There exists at least one subspace $\mathcal{L}$ such that it's partition function $Z(\mathcal{L}) \leq 1 + 2^k \nu^{2t}$.*

Using Markov's inequality, and fixing $k : 4 \geq 2^k \nu^{2t} \delta \geq 2$, we can show that [5]

$$\begin{aligned}\Pr[\frac{Z(\mathcal{L})}{(1 + 2^k \nu^{2t})(1 + \frac{\delta}{2})} \geq 1] &\leq \frac{\mathbb{E}(Z(\mathcal{L}))}{(1 + 2^k \nu^{2t})(1 + \frac{\delta}{2})} \\ &\leq 1 - \frac{\delta}{2 + \delta}.\end{aligned} \tag{3.10}$$

This means that, by picking $O(\frac{1}{\delta})$ subspaces at random, we can find $\mathcal{L}^\star$ such that

$$Z(\mathcal{L}^\star) \leq (1 + 2^k \nu^{2t})(1 + \frac{\delta}{2}) \tag{3.11}$$

which, by plugging in to Eq. 3.8, gives us the result

$$\left|\left\langle H^{\otimes t} \middle| \mathcal{L}^\star \right\rangle\right|^2 \geq 1 - \delta. \tag{3.12}$$

Thus, we can find a subspace capable of δ-approximating $\left|H^{\otimes t}\right\rangle$ with 2^k elements that satisfies the above requirement on k. This allows us to write $\chi = 2^k \leq 4\nu^{-2t}\delta^{-1}$, and obtain the asymptotic bound stated before.

It is important to note that this is an approximate decomposition, and so it's worth asking how this would compare to an approximate decomposition in to computational basis states. Using an argument from information theory, it is clear that the amplitude of t copies of a state $\alpha \left|0\right\rangle + \beta \left|1\right\rangle$ is concentrated on the 'typical' strings with Hamming weight [5, 44]

$$|x| = (1 - |\alpha|^2)t \pm O(\sqrt{t}). \tag{3.13}$$

The proportion of these typical strings is given by the binary Shannon entropy [44],

$$H_2(p) = p \log_2 \left(\frac{1}{p}\right) + (1 - p) \log_2 \left(\frac{1}{1 - p}\right)$$

and so this approximate computational decomposition gives a stabilizer rank [6]

$$\chi \sim 2^{t H_2(\nu^2)} \approx 2^{0.6t} \tag{3.14}$$

The authors conclude their argument with a second conjecture.

Conjecture 2. Any approximate decomposition of t edge type magic states that achieves a constant approximation error must have a stabilizer rank of at least $\Omega\left(\nu^{-2t} \approx 2^{0.23t}\right)$ [5].

This follows from the following lemma:

Lemma 3.2.2. *Consider a state* $|\psi\rangle = \sum_{a=1}^{\chi} z_a |\phi_a\rangle : \exists \mathcal{S}_{\phi_a} \forall |\phi\rangle_a$. *Suppose* $|\psi\rangle = 1$ *and* $\left|\langle\psi|H^{\otimes t}\rangle\right| \geq f$. *Then* $\chi \geq \nu^{-2t} f^2 \|z\|^{-2}$ *where* $z = (z_1, \cdots, z_\chi) \in \mathbb{C}^\chi$.

Proof. We define a quantity F_t such that

$$F_t \equiv \max_{\phi:\exists \mathcal{S}_\phi} \left|\left\langle \phi \middle| H^{\otimes t} \right\rangle\right|.$$

This quantity is clearly lower-bounded by ν^t, as $\left|\langle 0^{\otimes t}|H^{\otimes t}\rangle\right| = \nu^t$. We can also show that $F_t \leq \nu F_{t-1}$, by considering the case of performing a Pauli measurement on the first qubit of a t-qubit state $|\phi\rangle$. As this is a stabilizer measurement, the probability of a measurement outcome $P_a \in \{0, 1, \frac{1}{2}\}$. We have three cases:
Case 1: $P_0 = 1 \implies |\phi\rangle = |0\rangle \otimes |\psi\rangle : \exists \mathcal{S}_\psi$, and $F_t = \nu \left|\langle\psi|H^{\otimes t-1}\rangle\right| \leq \nu F_{t-1}$.
Case 2: $P_0 = 0 \implies |\phi\rangle = |1\rangle \otimes |psi\rangle$, and thus $F_t = \sqrt{1-\nu^2}\left|\langle\psi|H^{\otimes t-1}\rangle\right| \leq \nu F_{t-1}$
Case 3: $P_0 = \frac{1}{2}$, which implies that

$$|\phi\rangle = \frac{1}{\sqrt{2}}\left(|0\rangle \otimes |\psi_0\rangle + |1\rangle \otimes |\psi_1\rangle\right)$$

for stabilizer states $|\psi_{0,1}\rangle$. From the triangle inequality, this gives

$$F_t \leq \frac{1}{\sqrt{2}}\left(\nu + \sqrt{1-\nu^2}\right) F_{t-1} = \nu F_{t-1}$$

Thus, using $F_1 = \nu$, from the overlap of $|0\rangle$ with $|H\rangle$, we can show that the fidelity of the stabilizer state $|\psi\rangle$ with the magic states obeys

$$f \leq \left|\left\langle \psi \middle| H^{\otimes t} \right\rangle\right| \leq \nu^t \sum_{a=1}^{\chi} |z_a| \leq \nu^t \chi^{\frac{1}{2}} \|z\|$$

which can be rearranged to give the expression in the Lemma. □

This states that, when adding another magic state, the stabilizer rank must grow by at least ν^{-2}, and thus χ is strictly lower bounded by $\nu^{-2t} = 2^{\gamma t}$ where $\gamma = -2\log_2(\nu)$.

3.3 Significance for Quantum Computing

What Bravyi, Gosset, Smolin & Smith have achieved is a significant reduction in the computational complexity needed to simulate a Clifford+T circuit. An example of the relative magnitude of this exponential scaling between the `CHP` and `BGSS` is shown in Fig 3.2.1.
As can be seen, even for small T-counts ~ 25, the 8-fold quadratic reduction in the exponential scaling is an extremely significant improvement. The scaling remains exponential — we have not developed a method for efficiently simulating a quantum computation — but the circuits are significantly easier to simulate than might be expected. Indeed, a circuit with up to 50 T-gates has been simulated using a laptop [6], rather than High Performance Computing resources.

If we conceptualise quantum speedup as the regime where classical simulation becomes hard, this relative ease of simulation seems to suggest that the T gate has a limited computational power. This would imply we need to consume a large of magic states to realise a quantum algorithm which shows exponential speedup.

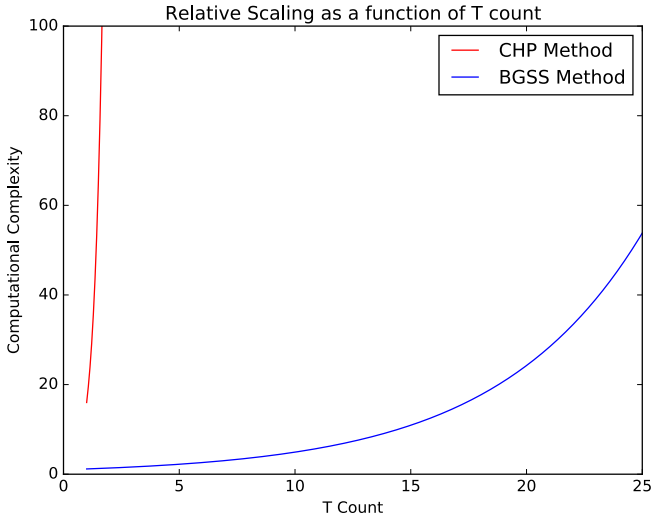

Figure 3.1: Plot showing the relative behaviour of the dominant exponential scaling factors in the computational complexity for the `CHP` and `BGSS` algorithms, simulating a Clifford+T circuit.

This indeed seems to be the case; using an implementation of a unitary synthesis algorithm for Clifford+T circuits, I obtained an estimate of 2000 T-gates required to realise Shor's algorithm on 10 qubits [45]. Even with the significant reduction in complexity, this would be impractical to simulate using the `BSSG` method.

This suggests, then, that the stabilizer Rank of a given state might instead serve as a useful measure of its value as a quantum computational resource.

Chapter 4

Stabilizer Rank for Alternative Resource States

This interpretation, that the stabilizer rank of a resource state allows us to quantify in some sense its computational power, naturally leads to the question of how the stabilizer rank scales for alternative resources.
Here, we consider single qubit resource states $|R\rangle$, which we use in conjunction with Clifford group operations. The state $|R\rangle$ is an eigenstate of an operator $R \in \mathcal{C}_n$, and can be generated by applying a unitary U to a stabilizer state. We can then show that

$$\begin{aligned} R|R\rangle &= RU|\phi\rangle : \exists \mathcal{S}_\phi \\ U^\dagger|R\rangle &= U^\dagger U|\phi\rangle = |\phi\rangle \\ \implies UsU^\dagger &= R \quad \forall s \in \mathcal{S}_\phi \\ &\implies U \in \mathcal{C}_{n+1} \end{aligned} \tag{4.1}$$

As an example, the edge-type magic state $|H\rangle$ is so named as it is an eigenstate of the Hadamard operator $H \in \mathcal{C}_2$, and thus the state is generated by a transform $U \in \mathcal{C}_3$, and allows us to realise this operation U.

In the first section, we will examine Conjecture 1 made by Bravyi, Smolin & Smith, by extending their method to find the asymptotic behaviour of $\chi(|F^{\otimes t}\rangle)$.
We will then use a pair of computational methods, a brute-force search and a proposed resource measure 'robustness', to try and find explicit values of χ. We use this to analyse arbitrary states, magic states, and finally resource states that would allow us to realise gates from higher levels in the Clifford hierarchy.

4.1 Stabilizer Rank of the Face-Type States

The face states are a family of 8 Clifford-equivalent magic states with a Bloch vector that projects out from the faces on the 'simulable' octahedron shown in Fig. 2.2. We will focus on the state $|F\rangle$[1], which has a description in the computational basis [37]

$$|F\rangle = \cos(\beta)|0\rangle + e^{i\frac{\pi}{4}}\sin(\beta)|1\rangle \ : \ 2\beta = \cos^{-1}(\frac{1}{\sqrt{3}}). \tag{4.2}$$

[1]This state is also called $|T\rangle$ in the original paper on magic state distillation [37], and $|R\rangle$ by Bravyi, Smoilin & Smith [6].

This state is an eigenstate of an operator we shall refer to as Δ, as it is a rotation of the triangular face in the octahedron around the axis $|F\rangle$. In particular, it rotates between the states that form the corners of the face.

$$\begin{aligned}\Delta\,|0\rangle &= |+i\rangle\\ \Delta\,|+i\rangle &= |+\rangle\\ \Delta|+\rangle &= |0\rangle\end{aligned} \tag{4.3}$$

Because this operator maps stabilizer states to stabilizer states, we can see that it is a Clifford group operation, and thus $|F\rangle$ allows us to realise an operation $U_F \in \mathcal{C}_3$.

To apply the results of Bravyi & Gosset discussion in Section 3.2.1, we need to find a similar expression in terms of t-bit strings. As the state is a symmetric point between three stabilizer states, we can define a ternary basis

$$\begin{aligned}|\tilde{0}\rangle &= |0\rangle\\ |\tilde{1}\rangle &= |+i\rangle\\ |\tilde{2}\rangle &= |+\rangle\end{aligned} \tag{4.4}$$

which has the desired property that each state has the same overlap with $|F\rangle$

$$|\langle\tilde{x}|F\rangle| = \cos(\beta) \equiv \mu. \tag{4.5}$$

Including the relevant phase corrections, we can thus write t copies of the F state as a sum over t-bit ternary strings

$$\left|F^{\otimes t}\right\rangle = \frac{1}{(3\mu)^t}\sum_{x\in\mathbb{F}_3^t} \mathrm{e}^{i(|x|_2-|x|_1)\phi}\,|\tilde{x}_1\otimes\cdots\otimes\tilde{x}_t\rangle \tag{4.6}$$

where the angle ϕ is equal to $\frac{\pi}{12}$, and $|x|_{1,2}$ is the 1 and 2-weight of the string x. This decomposition, however, contains 3^t stabilizer states, and so is actually over-complete compared to the computational basis representation.

We thus need a way to define a normalised state associated to linear subspaces $\mathcal{L} \subseteq \mathbb{F}_3^t$, such that we can find decompositions with $\chi = 3^k$. This requires redefining the partition function $Z_F(\mathcal{L}) = \sum_{x\in\mathcal{L}} f'(x)$. The additional complication here is the presence of a complex phase in the state. Again, the amplitude of the overlap between the three states is equal

$$|\,\langle\tilde{x}|\tilde{y}\rangle\,| = 2^{-\frac{(1-\delta_{xy})}{2}},$$

where δ_{xy} is the Kronecker delta, but we also have that

$$\langle\tilde{1}|\tilde{2}\rangle = \frac{1}{\sqrt{2}}\mathrm{e}^{-i\frac{\pi}{4}} = (\langle\tilde{2}|\tilde{1}\rangle)^*$$

Thus, the absolute value of the overlap between a pair of strings will have a magnitude given by the difference in their Hamming weights, and an additional correction given by the phases, giving

$$f'(x) \equiv 2^{-\frac{|x|}{2}}\cos((|x|_2 - |x|_1)\phi). \tag{4.7}$$

This was verified numerically by calculating the overlap between different linear subspaces. Thus, we can define the appropriate normalised state

$$|\mathcal{L}\rangle = \frac{1}{\sqrt{3^k Z_F(\mathcal{L})}}\sum_{x\in\mathcal{L}} \mathrm{e}^{i(|x|_2-|x|_1)\phi}\,|\tilde{x}_1\otimes\cdots\otimes\tilde{x}_t\rangle \tag{4.8}$$

Once we have this explicit form for the stabilizer states, we can thus verify a similar form for the overlap between the magic states and a linear subspace

$$\left|\left\langle F^{\otimes t}\middle|\mathcal{L}\right\rangle\right|^2 = \frac{3^k\mu^{2t}}{Z_F(\mathcal{L})}. \tag{4.9}$$

which arises from the fact that each bit in each string has overlap μ with the state $|F\rangle$, and that there are 3^k such strings, giving

$$\left|\left\langle F^{\otimes t}\middle|\mathcal{L}\right\rangle\right|^2 = \left(\frac{3^k\mu^t}{\sqrt{3^k Z_F(\mathcal{L})}}\right)^2 = \frac{3^k\mu^{2t}}{Z_F(\mathcal{L})}.$$

We can use this result to prove an equivalent of Lemma 3.2.1. In particular, we need to evaluate the expression

$$\sum_{x\in\mathbb{F}_3^t\setminus 0^t} f'(x)\mathbb{E}(\chi_{\mathcal{L}}(x))$$

The partition function for the string $\left|\tilde{0}^{\otimes t}\right\rangle$ is 1. We can use the fact that $\left|\langle F^{\otimes t}|\mathbb{F}_3^t\rangle\right|^2 = 1$, and rearrange Eq. 4.9 to express it in terms of $Z_F(\mathcal{L})$, to give

$$\begin{aligned}\sum_{x\in\mathbb{F}_3^t\setminus 0^t} f(x) &= \frac{3^t\mu^{2t}}{\left|\langle F^{\otimes t}|\mathbb{F}_3^t\rangle\right|^2} - 1 \\ &= 3^t\mu^{2t} - 1\end{aligned} \tag{4.10}$$

We also note that, as the indicator function $\chi_{\mathcal{L}}(x)$ is 1 if and only if $x \in \mathcal{L}$, and as $\chi_{\mathcal{L}}(0^t) = 1\forall\mathcal{L}$ as the zero-string is the 'identity' element for these subspaces, it's expectation value is simply given by

$$\sum_{x\in\mathbb{F}_2^t\setminus 0^t} \mathbb{E}\left(\chi_{\mathcal{L}}(x)\right) = \frac{3^t-1}{3^k-1} \tag{4.11}$$

the relative number of elements in the subspace.

Thus, we can prove that Lemma 3.2.1 translates to these linear subspaces on $\mathbb{F}_3^t$, namely

$$\mathbb{E}\left(Z_F(\mathcal{L})\right) \leq 1 + 3^k\mu^{2t}. \tag{4.12}$$

The subsequent argument given in Eq. 3.10, using Markov's inequality to show that we can find $\mathcal{L}^\star : Z\left(\mathcal{L}^\star\right) \leq \left(1+2^k\nu^{2t}\right)\left(1+\frac{\delta}{2}\right)$, seems to apply directly for Z_F and the linear subspaces on $\mathbb{F}_3^t$, provided we fix

$$9 \geq 3^k\mu^{2t}\delta \geq 3 \tag{4.13}$$

$$\implies \exists\, \mathcal{L}^\star : Z_F\left(\mathcal{L}^\star\right) \leq \left(1+3^k\mu^{2t}\right)\left(1+\frac{\delta}{2}\right). \tag{4.14}$$

Thus, we can use this inequality on $Z_F(\mathcal{L})$ in Eq. 4.9, to show that$|\mathcal{L}^\star\rangle$ δ approximates $\left|F^{\otimes t}\right\rangle$.

$$\begin{aligned}\left|\left\langle F^{\otimes t}\middle|\mathcal{L}^\star\right\rangle\right|^2 &\geq \frac{3^k\mu^{2t}}{\left(1+3^k\mu^{2t}\right)\left(1+\frac{\delta}{2}\right)} \\ &\geq \frac{1}{\left(1+3^{-k}\mu^{-2t}\right)\left(1+\frac{\delta}{2}\right)} \\ &\geq \frac{1}{\left(1+\frac{\delta}{2}\right)^2} \qquad \text{From Eq. 4.13} \\ &\geq 1-\delta\end{aligned} \tag{4.15}$$

This resulting state has a stabilizer rank

$$\chi = 3^k \leq 9\mu^{-2t}\delta^{-1} = O\left(\mu^{-2t}\delta^{-1}\right) = O\left(2^{\gamma_F t}\delta^{-1}\right) \tag{4.16}$$

where the scaling factor γ_F is obtained by using the properties of the logarithm to rewrite $\mu^{-2t} = 2^{-2t\log_2(\mu)}$, giving $\gamma_F \approx 0.345$.
Lemma 3.2.2 then follows immediately for the face-type magic states, as it only depends on the maximal value of the overlap between a stabilizer state and the magic state. In this case, it is given by $\mu = \cos(\beta)$, and gives the strict lower bound $\chi = \Omega\left(\mu^{-2t} = 2^{\gamma_F t}\right)$.

Finally, we can consider the problem of approximate computational basis decompositions. The argument given in Equations 3.13 and 3.14 also applies for our $|F\rangle$ states, but instead is based on the binary entropy $H_2(\mu^2)$. This gives the rank of the approximate computational basis decomposition as

$$\chi \sim 2^{H_2(\mu^2)} \approx 2^{0.95t}. \tag{4.17}$$

We can check this argument by finding the Hamming weight of the computational state with the largest amplitude for the states $|F\rangle^{\otimes n}$, and comparing it to the approximate bound on the maximal Hamming weight given in Eq. 3.13. This is shown for up to $n = 15$ in Fig. 4.1.

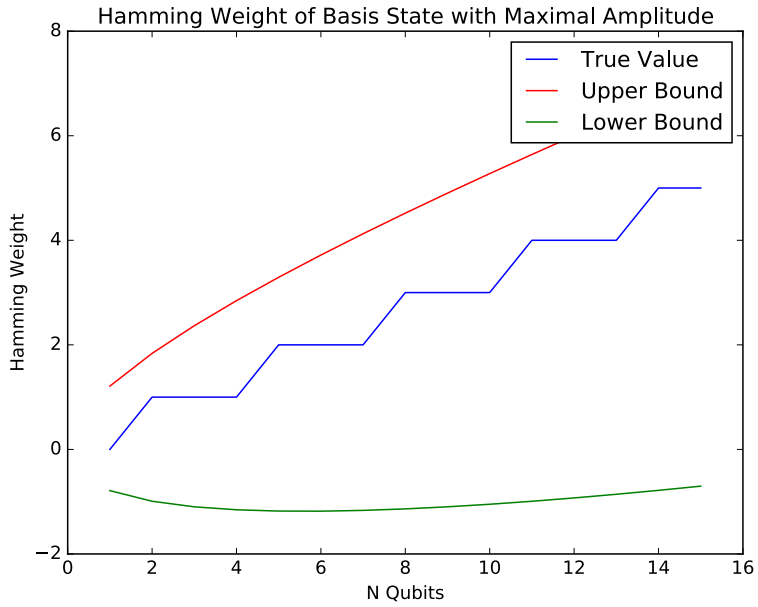

Figure 4.1: Plot showing the Hamming weight of the computational basis state with the highest amplitude for n copies of the magic states $|F\rangle$, as well as the approximate bounds $|x| \approx (1-\mu^2)t \pm O(t)$.

Thus, by defining these approximate decompositions in to ternary strings, we have been able to show that the stabilizer rank of a face-type state has a larger asymptotic bound that for edge-type states. Additionally, if Conjecture 2 holds, which seems evident from the proof of Lemma 3.2.2, then these states have a stabilizer rank which is strictly larger than for the edge-type states.

4.2 Explicit Stabilizer Rank Decomposition

The technique used above to bound the stabilizer rank of a given state does not easily generalise, because it requires that the state be decomposable into a symmetric superposition of stabilizer states. This is satisfied by magic states, due to their definition as symmetric points between the single-qubit stabilizer states, but doesn't apply for more general resource states.
In particular, we would like to analyse the stabilizer rank for resource states that can be used to realise gates from higher levels in the Clifford hierarchy. An easy way to generate such states is to make use of an observation about the Pauli Z operator. By definition as a Pauli operator, Z is in level $\mathcal{C}_1$. It's square-root, the Phase gate, is a Clifford operation, and thus $\in \mathcal{C}_2$. The root of the phase gate, T, is another level higher, $\in \mathcal{C}_3$. This seems to suggest that taking repeated roots of the gate Z generators operators from successively higher levels of the Clifford hierarchy. This is indeed the case.

These operators are diagonal and live in $\mathcal{C}_{n\geq 3}$, and thus they meet the criteria to be realised using the state injection method discussed in Section 2.4.1. This means the corresponding resource state can be generating by acting the operator $Z^{\frac{1}{n}}$ on the state $|+\rangle$.
However, as these states are given by a rotation of the state $|+\rangle$ by an angle $\phi < \frac{\pi}{2}$ along the equator of the Bloch sphere, they are not symmetric points between pairs of stabilizer states. This means we will need to determine their stabilizer rank explicitly.

As discussed in Section 3.2, Brayi, Smolin & Smith found approximate values for the stabilizer rank of $|H\rangle$ for up to 6 qubits. They did this by implementing a random-walk within the space of n-qubit stabilizer states, seeking to find a decomposition set $\tilde{\phi} : \left\| \Pi_{\tilde{\phi}} |H^{\otimes n}\rangle \right\| = 1$ [6].

In this algorithm, β is a positive-valued parameter analogous to an inverse temperature. This method then resembles a form of simulated annealing, where we 'walk' through an energy landscape defined by the objective function F, here given by $1 - \|\Pi_\phi |H^{\otimes n}\rangle\|$, trying to find a global minimum, and move 'up' the energy landscape with a probability given by the Boltzmann distribution for β. Increasing β amounts to decreasing the temperature, effectively 'freezing' our random walk into a local minimum that presumed to be close-to-optimal. This method is described in Algorithm 1.

However, we would like to find minimal decompositions deterministically for a small number of qubits to be able to state categorically the relative stabilizer rank of different resource states. We attempt this using two methods: brute-force search, and a resource measure called robustness.

4.2.1 Computationally Generating Stabilizer States

Both techniques require us to computationally generate the stabilizer states on n qubits. The number of such states grows rapidly in the number of qubits, and is equal to [4]

$$N_\phi = 2^n \prod_{k=0}^{n-1} \left(2^{n-k} + 1\right). \tag{4.18}$$

To find the states, we must first build all n-qubit stabilizer groups, which we can do by finding their sets of n generators. For each group $\mathcal{S} = \langle s_1, \cdots, s_n \rangle$ on n qubits, we can then build a projector on to the stabilizer space [32]

$$\Pi_{\mathcal{S}} = \prod_{i=1}^{n} \frac{1}{2} \left(\mathbb{I} + s_i\right) \tag{4.19}$$

and the corresponding state can be found by finding it's +1 eigenstate.

To generate the stabilizer groups, we employed the binary string representation used by Aaronson & Gottesman, which was described in Section 2.4.1. Namely, a Pauli operator is defined by

Algorithm 1 Random Walk on Stabilizer States

Require: $\beta_{init}, \beta_{max}, M$, target integer χ

$\tilde{\phi} \leftarrow (\phi_1, \cdots \phi_\chi)$ where each ϕ_a is chosen at random.
$\beta \leftarrow \beta_{init}$
while $\beta < \beta_{max}$ **do**
  for $i = 0$ to 100 **do**
    Evaluate $F(\tilde{\phi}) = 1 - \|\Pi_\phi |H^{\otimes n}\rangle\|$
    if $F(\tilde{\phi}) = 0$ **then**
      return $\tilde{\phi}$
    end if
    Pick random integer $a \in [1, n]$ and random Pauli $P \in \mathcal{P}_n$
    $\phi'_a \leftarrow c\,(\mathbb{I} + P)\,\phi_a$
    $\tilde{\phi}' \leftarrow (\phi_1, \cdots, \phi'_a, \cdots \phi_\chi)$ ▷ Replace ϕ_a with ϕ'_a
    if $F(\tilde{\phi}') < F(\tilde{\phi})$ **then**
      $\tilde{\phi} \leftarrow \tilde{\phi}'$
    else
      $p_{accept} \leftarrow \exp\Big[-\beta\left(F(\tilde{\phi}') - F(\tilde{\phi})\right)\Big]$
      Pick random $r \in [0, 1]$
      if $r > p_{accept}$ **then**
        $\tilde{\phi} \leftarrow \tilde{\phi}'$
      end if
    end if
  end for
  $\beta \leftarrow \beta + \left(\frac{\beta_{max} - \beta_{init}}{M}\right)$
end while
return No decomposition found.

$2n + 1$ bits: $x_i : i \in \{1, \cdots, n\}$, $z_i : i \in \{1, \cdots, n\}$ and r. The global phase is given by -1^r, and the ith operator in the tensor product by the bits x_i, z_i.
This allows us to efficiently generate all n-qubit Pauli operators with real-valued phase by generating the binary strings B for the numbers 0 to $2^{2n+1} - 1$ [4].

We can try to generate a stabilizer group by picking a set of n strings from this full set, and generating the corresponding linear subspace by combining the strings under binary addition. To be a valid stabilizer state, we must first check that all n strings mutually commute. We note that two n-qubit Pauli operators commute if an even number of qubits are acted on by a pair of Pauli operators. Otherwise, the operators anticommute. For example,

$$[X \otimes \mathbb{I} \otimes X, Z \otimes \mathbb{I} \otimes Z] = 0$$
$$\{X \otimes X \otimes X, Z \otimes Z \otimes Z\} = 0.$$

In this binary representation, the commutativity of two Pauli operators represented by strings B_h, B_i can be efficiently checked by evaluating their symplectic inner product [4]

$$B_h \cdot B_i \equiv x_{h,1} z_{i,1} \oplus \cdots \oplus x_{h,n} z_{i,n} \oplus z_{h,1} x_{i,1} \oplus \cdots \oplus z_{h,n} x_{i,n} \tag{4.20}$$

where the addition is modulo 2. The corresponding Pauli operators commute if their binary representations have a symplectic inner product of 0, and anticommute otherwise.
To describe a stabilizer space, the subspace then generated by these commuting operators must contain 2^n elements, which can be used as a way to check candidate spaces. The resulting space also cannot contain the string corresponding to $0 \cdots 01$, as this element is equivalent to $-1\mathbb{I}$. Any Pauli subgroup containing the element $-1\mathbb{I}$ cannot have be a stabilizer group, as $-\mathbb{I}\,|\psi\rangle = -\,|\psi\rangle$.
If the resulting group hadn't previously been found, it was then stored. Otherwise, the candidate group was discarded.

In practice, this method found more 'unique' stabilizer groups than expected from Eq. 4.18. This was because the technique was identifying as distinct two groups equivalent up to global phase. Instead, we noted that there are 2^n ways of distributing the phase between the n generators. For example, on two qubits

$$\begin{aligned}\mathcal{S} &= \langle s_1, s_2\rangle \\ &\neq \langle -s_1, s_2\rangle \\ &\neq \langle s_1, -s_2\rangle \\ &\neq \langle -s_1, -s_2\rangle.\end{aligned}$$

Thus, we can generate the full set of stabilizer groups but first generating the $\frac{N_\phi}{2^n}$ groups with all-positive Pauli operators, and then adding phase to the resulting generators 'by hand'. This full method is outlined in Algorithm 2, and was implemented in Python using `Qutip` [46] and the `bitarray` library. Due to the large number of such groups, we were only able to generate the stabilizer states for $n \leq 4$ qubits, due to constrains on available memory.

4.2.2 Brute Force Search

This is the simplest method of finding a decomposition in to magic states. For a given n, we pick a set of i states from the set of all n-quit stabilizer states. We then have to build the projector on to the space generated by these states, and test the overlap with the target states $|R^{\otimes n}\rangle$. If it is equal to one, we are done. Else, we continue testing all i states. If this is exhausted, we increment i and resume. This will continue up to $i = 2^n$, where the method will find the computational basis decomposition and exit. Pseudo-code for the method is given in Algorithm 4.

The projector onto the space spanned by the Stabilizer states is found using Gram-Schmidt orthogonalization [20]. This well established technique for building an orthogonal basis for the space spanned by a collection of linearly independent vectors $\{v_i\}$. As the stabilizer states are known to form a mutually unbiased basis [47], this independence condition is always met for combinations of stabilizer states. Pseudo-code for the method is shown in Algorithm 3.
The Gram-Schmidt orthogonalisation procedure was implemented using the `Numpy` library for fast matrix operations [48].

4.2.3 The Robustness Measure

Robustness as a resource measure was first defined by Vidal & Tarrach in 1999 [49]. It was proposed as an alternative measure of entanglement that, combined with other metrics like the entanglement entropy, could be used to fully characterise an entangled state. The motivation behind the measure is to quantify how 'robust' an entangled state would be to local operations

Algorithm 2 Generating n qubit stabilizer groups

function StringsToPauli(G)
  $\mathcal{S} \leftarrow \emptyset$
  for $g \in G$ **do**
    $\mathcal{S} \leftarrow \mathcal{S} \cup \bigotimes_{i=1}^{n} i^{x_i \wedge z_i} X^{x_i} Z^{z_i}$ ▷ where $g = (x_1 z_1 \cdots x_n z_n) \in \mathbb{F}_2^{2n}$
  end for
  return $\mathcal{S}$
end function

Require: n ▷ Number of qubits
Require: GetEigenstate(Projector)
S $\leftarrow \mathbb{F}_2^n$, Generators $\leftarrow \emptyset$, States $\leftarrow \emptyset$
TotalGenerators $\leftarrow \frac{N_\phi}{2^n}$ ▷ where N_ϕ is defined in Eq. 4.18.
while |Generators| < TotalGenerators **do**
  $G = \{g_1, \cdots, g_n\} \subseteq S$
  if $[g_i, g_j] = 0 \; \forall \; g_i, g_j \in G$ **then**
    $\mathcal{L} \leftarrow \langle g_1, \cdots, g_n \rangle$
    if $-\mathbb{I} \in \mathcal{L}$ **or** $|\mathcal{L}| \neq 2^n$ **then**
      Reject candidate
    end if
    if $\mathcal{L} \notin$ Generators **then**
      Generators$\leftarrow$ Generators $\cup\{\mathcal{L}\}$
    end if
  end if
end while
for $G \in$ Generators **do**
  $\mathcal{S}$ = StringsToPauli(G)
  Projector $\leftarrow \prod_{i=1}^{n} \frac{1}{2} (\mathbb{I} + s_i) \quad \forall s_i : \mathcal{S} = \langle s_1, \cdots, s_n \rangle$
  States $\leftarrow$ states $\cup$ GetEigenstate(Projector)
  for $\tilde{x} \in \mathbb{F}_2^n \setminus 0^n$ **do**
    Projector $\leftarrow \prod_{i=1}^{n} \frac{1}{2} (\mathbb{I} + -1^{\tilde{x}_i} s_i) \quad \forall s_i : \mathcal{S} = \langle s_1, \cdots, s_n \rangle$
    States $\leftarrow$ states $\cup$ GetEigenstate(Projector)
  end for
end for
return States

Algorithm 3 Gram Schmidt Orthogonalisation

function PROJECTION($|u_i\rangle, |v_j\rangle$)
  return $\frac{\langle v_j|u_i\rangle}{\langle u_i|u_i\rangle}|u_i\rangle$
end function

Require: $\{|v_1\rangle, \cdots, |v_k\rangle\}$ ▷ Set of k stabilizer states

for $i = 1$ **to** k **do**
  $|u_i\rangle \leftarrow |v_i\rangle$
  $j \leftarrow i - 1$
  while $j > 0$ **do**
    $|u_k\rangle \leftarrow |u_k\rangle -$ PROJECTION($|u_j\rangle$, $|v_i\rangle$)
    $j \leftarrow j - 1$
  end while
  $|u_i\rangle \leftarrow \frac{1}{\langle u_i|u_i\rangle}|u_i\rangle$ ▷ Ensure normalisation
end for
$A \leftarrow [A_{i,j} = \langle u_i|u_j\rangle]$
return A

Algorithm 4 Brute Force Search for stabilizer rank

Require: Φ ▷ the set of n qubit magic states
Require: GRAMSCHMDIT($\tilde{\phi}$) ▷ Implementation of Orthogonalisation

function FINDCHI($|\psi\rangle$)
  for $i = 1$ to $2^n - 1$ **do**
    $\tilde{\phi} = (\phi_1, \cdots, \phi_n) : \phi_i \in \Phi \forall\ i$
    $\Pi_{\tilde{\phi}} \leftarrow$ GRAMSCHMIDT($\tilde{\phi}$)
    if $\left\|\Pi_{\tilde{\phi}}|\psi\rangle\right\|$ **equals** 1 **then**
      return $n, \tilde{\phi}$
    end if
    $i \leftarrow i + 1$
  end for
  return 2^n, $\tilde{x}$ ▷ the set of computational basis states on $\mathbb{C}^{2^n}$
end function

that increase the mixedness of one subsystem [49].
In a resource theory, states or operations are split in to two categories: 'free' resources, $\mathcal{P}_{free}$ and 'expensive' resources. For example, in entanglement theory separable states are the free resource, and entangled states the expensive resource [49]. This gives the robustness a clear geometric picture in terms of the set of free resources, as illustrated in Fig. 4.2.

From the figure, we can see that we define a pair of states at extreme ends of the free resource set. Given this pair of states, the Robustness is then defined as [49]

$$\mathcal{R}(\rho) \equiv \min_{\rho_+,\rho_- \in \mathcal{P}_{free}} 2p+1 : \rho = (p+1)\,\rho_+ - p\rho_-. \tag{4.21}$$

This resource has the property that it is 1 for any state that forms a vertex of $\mathcal{P}_{free}$, and that it is subadditive under the combination of systems [49, 50].

$$\mathcal{R}(\rho_1 \otimes \rho_2) \leq \mathcal{R}(\rho_1)\mathcal{R}(\rho_2).$$

The geometric interpretation here is similar to the techniques used to quantify the power of 'non-local' correlations in general probabilistic models, where the 'free' resource is the polytope defined by strictly local or 'classical' correlation [51].

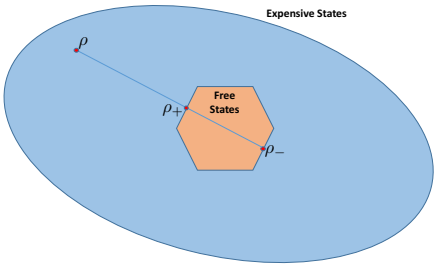

Figure 4.2: Simplified representation of the robustness measure, showing the definition of the vertices $\rho_\pm$ relative to the resource state we wish to quantity. Depiction inspired by [49, 50].

This measure can be adapted to different resource states; in particular, we could consider using it to quantify quantum speedup by defining the free resource as the set of efficiently simulable states, with vertices defined by the stabilizer states. The expensive resource is then the resource states $|R\rangle$ we will use to promote the Clifford circuits to universality [50].
The use of robustness to quantify 'magicness' was proposed by Campbell & Howard [50], motivated by studying contextuality in quantum computing, using magic states as the resource state. In particular, they sought a measure that would allow them to quantify contextuality as a resource in both qubit and qudit computing.

It was noted by Campbell & Howard that the problem of evaluating the robustness can be converted into a linear programming problem called ℓ_1 minimisation. By defining a matrix A where column j is given by the jth vertex of $\mathcal{P}_{free}$, and defining a corresponding column vector b for our resource state, they show that [50]

$$\mathcal{R}(\rho) = \min \|x\|_1 \ : Ax = b \tag{4.22}$$

where $\|x\|_1 \equiv \sum_i |x_i|$.

Algorithm 5 The `SL0` algorithm for ℓ_0 estimation.

Require: $\{\Phi_i\}$ ▷ The set of stabilizer states on a given number of qubits
Require: SigmaDecreaseFactor ▷ Authors recommend a value of 0.5 [52].
Require: μ_0 ▷ Gradient used in gradient ascent. Authors recommend 2 [52].
Require: PseudoInverse(Mat) ▷ Provided by most linear algebra libraries.
  function SmoothedL0(n, $|R\rangle$, σ_{min}) ▷ Number of qubits, target state, minimum value of σ
    $A \leftarrow [A_{i,j} = \langle i|\phi_j\rangle] \quad \forall\, i \in [0, 2^n - 1], \phi_j \in \Phi_n$
    $b \leftarrow \{b_i = \langle i|R\rangle\} \quad \forall\, i \in [0, 2^n - 1]$
    $A^+ \leftarrow$ PseudoInverse(A)
    $x \leftarrow A^+ b$
    $\sigma \leftarrow 2 \times \max_{\forall x_i \in x}\{|x_i|\}$
    while $\sigma > \sigma_{min}$ **do**
      for $i = 0$ to $i = 2$ **do**
        $\delta \leftarrow \{f_\sigma(x_i)\ \forall x_i \in x\}$
        $x \leftarrow x - \mu_0 \delta$ ▷ Do the gradient ascent step
        $x \leftarrow x - A^+ (Ax - b)$ ▷ Project x back onto the set of solutions to $Ax = b$ [52].
      end for
      $\sigma \leftarrow \sigma \times$SigmaDecreaseFactor
    end while
    return CountNonZero(x)
  end function

In their talk, Campbell & Howard noted that for single magic states $|H\rangle$, evaluating a similar problem called ℓ_0 minimisation gave a related measure

$$\mathcal{R}'(\rho) = \min \|x\|_0 \ : Ax = b \tag{4.23}$$

that corresponded to the value of the stabilizer rank decomposition $\chi(|H\rangle) = 2$, where $\|x\|_0 \equiv \{\#x_i : x_i \neq 0\}$, the 'sparseness' of the vector x [50].
This can be understood by considering a the explicit construction of the matrix A and vector b in the case where $\mathcal{P}_{free}$ is defined by the set of stabilizer states. In this case, we can define

$$A_{ij} \equiv \langle i|\phi_j\rangle \tag{4.24}$$

where $|i\rangle$ is the ith computational basis state given by the binary representation of the integer $i \in [0, 2^n - 1]$, and $|\phi_j\rangle$ is the jth stabilizer state [50]. The corresponding definition of b is then

$$b_i \equiv \langle i|R\rangle \tag{4.25}$$

where $|R\rangle$ is the resource state we want to analyse.
The vector x thus gives the representation of our state in the stabilizer basis, and the sparsest vector should correspond to the optimal stabilizer rank decomposition.

It is known that the problem of finding the global minimum of the ℓ_0 norm is **NP**-hard [53]. However, as this problem occurs in a signal processing context, fast heuristic algorithms exist. In particular, we implemented a method called 'Smoothed ℓ_0' (`SL0`) [52].
The `SL0` algorithm works by using a 'gradient ascent' method to approximately maximise a function $F_\sigma(x) = \sum_i f_\sigma(x_i)$ for each element in the vector x. This function has the property that

$$\lim_{\sigma \to 0} f_\sigma(x_i) = \begin{cases} 1 & \text{if } x_i = 0 \\ 0 & \text{if } x_i \neq 0 \end{cases} \tag{4.26}$$

and so it's maximization for increasingly small values of σ should cause us to converge to the sparsest solution of the vector x [52] In particular, they define a Gaussian function [52]

$$f_\sigma(x_i) \equiv x_i \exp\left(-\frac{|x_i|^2}{2\sigma^2}\right). \tag{4.27}$$

Pseudo-code for the `SL0` method is given in Algorithm 5. The method was evaluated by building the A matrix for n qubits, and then running `SL0` using this matrix and the corresponding vector b. Correspondence with stabilizer rank for the single magic state $|H\rangle$ was found using a minimum $\sigma_{min} = 1 \times 10^{-12}$; further reductions in σ_{min} didn't impact the sparseness returned.

4.2.4 Results

We began by examining the stabilizer rank for arbitrary quantum states, to verify that these n-qubit states have a rank $\chi \approx 2^n$. The results for both the `SL0` and brute force searches, are shown in Table 4.1. We use $|rand_n\rangle$ to denote a random state on n qubits, which is distinct from $|rand\rangle^{\otimes n}$, n copies of a given random state.

Table 4.1: Stabilizer Rank for arbitrary quantum states

State	$\|rand\rangle$	$\|rand\rangle^{\otimes 2}$	$\|rand_2\rangle$	$\|rand\rangle^{\otimes 3}$
$\|x\|_0$	4	14	15	168
Brute Force	2	3	4	> 4

We can see that, for the explicit decompositions of n qubit random states, the stabilizer rank is in fact maximal. For multiple copies of a single random qubit, we can see that there is a small reduction in χ; this would be expected from the complexity of the classical description, where a product state requires fewer classical numbers to characterise fully.
It is also immediately clear from the results that the `SL0` method has not successfully converged to an optimally sparse solution, but instead returns an estimate of χ larger than the simple upper bound given by the computational basis. However, the values returned do seem to be indicative of the magnitude of χ: for example, $\texttt{SL0}\left(|rand\rangle^{\otimes 2}\right) < \texttt{SL0}(|rand_2\rangle)$.

Table 4.2: Stabilizer Rank for Magic States $|H\rangle$ and $|F\rangle$.

n qubits	1	2	3	4
$\|x_H\|_0$	2	6	116	3676
$\|x_F\|_0$	3	9	108	3753
Brute Force H	2	2	3	4
Brute Force F	2	2	3	> 3

We can then extend this to examine explicit decompositions for the magic states $|H\rangle$ and $|F\rangle$, which are shown in Table 4.2. Again, the estimates returned by the `SL0` method are over-complete, significantly larger than the computational basis decomposition. However, it is interesting to note that the estimates are smaller than the estimates for arbitrary states, and that they are generally larger for the face-type magic states.
We can see that the deterministic values of χ for $|H^{\otimes n}\rangle$ coincide with the approximate values found by Bravyi, Smolin & Smith [6], and given in Table 3.1. Interestingly, the explicit values found for the $|F\rangle$ states coincide with the results for $|H\rangle$, at least as far as 3 qubits; unfortunately due to

the computational time consumed by the brute-force search, the decomposition for the state $\left|F^{\otimes 4}\right\rangle$ did not complete in time, so the stabilizer rank given is merely a bound.

Table 4.3: Stabilizer Rank for $\left|\sqrt{T}\right\rangle$, $\left|\sqrt[4]{T}\right\rangle$.

n qubits	1	2	3	4
$\left\|x_{\sqrt{T}}\right\|_0$	2	14	150	4487
$\left\|x_{\sqrt[4]{T}}\right\|_0$	2	12	173	5023
Brute Force $\sqrt{T}$	2	3	> 3	> 3
Brute Force $\sqrt[4]{T}$	2	3	> 3	> 3

Finally, we can consider the results for resource states generating gates from higher levels in the Clifford hierarchy. In particular, we consider the 3-th and 4-th square-roots of the Z gate, which we denote as $\sqrt{T}$, $\sqrt[4]{T}$ respectively. These gates are equivalent to phase gates R_ϕ with $\phi = \frac{\pi}{8}$ and $\frac{\pi}{16}$, and realise operations from $\mathcal{C}_4$ and $\mathcal{C}_5$. The corresponding decompositions are shown in Table 4.3.

What is interesting is that, both the `SLO` estimates and the explicit decompositions for these $\mathcal{C}_{n>3}$ resource states are larger than for the magic states. The `SLO` estimates are actually comparable to those for arbitrary quantum states, and the explicit decompositions are already bigger than for magic states for $n = 2$.

We note here that the some of the results in the tables above are only bounds on the value of χ. This was due to the inefficiency of the brute-force search, which is subject to combinatiorial explosion in the number of combinations required to test, both as a function of χ and as a function of the number of qubits.

Chapter 5

Discussion

As discussed in Section 3.3, we are interested in understanding the stabilizer rank of a state as a potential quantifier of its value in quantum computation. This correspondence is predicated on the idea of using classical simulation to probe the quantum advantage, motivated by the Church-Turing-Deutsch thesis.
However, the classical simulation complexity is strongly representation dependent [2]. Assuming the validity of the CTD thesis, the presence of an efficient classical simulation can be used to categorically rule out quantum speedup, but an inefficient classical description does not guarantee quantum advantage. Nonetheless, we argue that the stabilizer rank has the potential to constrain quantum speedup.
Because it focuses on simulating universal quantum computations, we do not expect it to reduce to an efficient classical simulation. However, it does allow a relatively more efficient classical description of certain resource states, and thus it is interesting to probe the connection between χ and the value of different resources.

A simple confirmation of this concept is that large T-counts are required to implement quantum algorithms which demonstrate exponential speedup, described in Section 3.3. Alternatively, we have shown in Table 4.1 that these stabilizer decompositions offer little to no computational advantage for arbitrary quantum states; their stabilizer rank is maximal $\chi \approx 2^n$. This can be understood as representing the ideal of 'perfect' control of a quantum system. Access to arbitrary states suggests the ability to perform arbitrary single and multi-qubit operations with high-fidelity, as opposed to having to build them out of the elements of a restricted universal set of gates and ancillae.
A reduction in stabilizer rank for resource states, and subsequently in the simulation complexity of the corresponding Clifford+'R' fault tolerant scheme, then has a natural correspondence to the increase in the number of elementary operations and physical qubits required to realise the same circuit relative to 'perfect' control.

5.1 Face-Type Magic States

Under this interpretation, then, the proof presented in Section 4.1 would suggest that the face-type magic states serve as a stronger resource for quantum computation than the edge-type states, as they admit a smaller reduction in classical complexity. This is in fact congruent with the observation in [3] that the face-type states lead to a larger violation of contextuality inequalities than edge type states. This seems to further support the interpretation of contextuality as a resource for quantum computing.

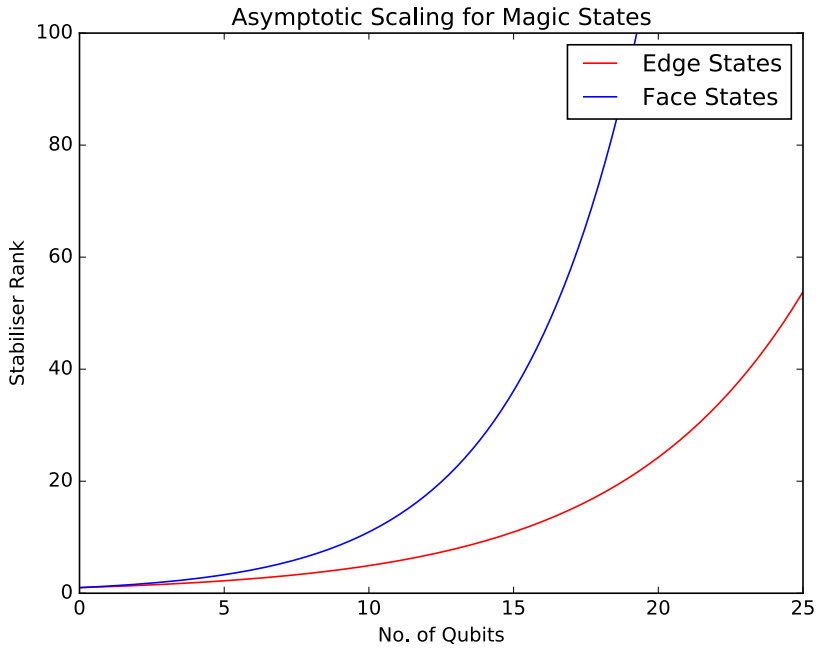

Figure 5.1: Plot demonstrating the growth in the exponential scaling factors $2^{\gamma t}$ for both edge and face-type magic states.

The relative size of the scaling factors for the two families of magic states is given by

$$\frac{\gamma_F}{\gamma_H} = \frac{\log_2\left(\cos(\beta)\right)}{\log_2\left(\cos(\frac{\pi}{8})\right)} = \frac{3}{2} \tag{5.1}$$

which has a simple geometric interpretation in terms of the definition of each state: edge-type magic states are symmetric points between a pair of stabilizer states, whereas face-type states are the symmetric point between 3 stabilizer states. This can be seen in Fig. 2.2. Interestingly, this ratio of $\frac{3}{2}$ is also the ratio of the `SLO` estimates of χ for single-copies of the $|F\rangle$ and $|H\rangle$ states.

However, despite the demonstration of a larger asymptotic bound, the explicit decompositions found for the two families of states were equal up to 3 qubits. This is most likely a consequence of the relatively small effect of the exponential scaling for a small number of qubits. The exponential scaling factors $2^{\gamma t}$ for each family of magic states are plotted in Fig. 5.1. Here, you can see that the initial difference in the scaling is small, and that the two are approximately equal up to 5 qubits. Thus, the lack of any apparent difference is a consequence of the computational constraints when finding these decompositions.

Unfortunately, a significant reason as to why the quantum computing community focus on the edge-type magic states is that there is currently no known circuit capable of using a face-type state to realise a gate from $\mathcal{C}_3$ *deterministically*. The rotation $U : |F\rangle = U\,|+\rangle$ is equal to a Y-rotation followed by a T gate. While demonstrably in $\mathcal{C}_3$ from Eq. 4.1, it is not a diagonal operator and thus the face-type states cannot be used in the state-injection circuit. In their original paper on magic state distillation, Bravyi & Kitaev demonstrated a circuit that could, probabilistically, convert the state $|F\rangle \otimes |F\rangle$ to a resource state $\left|A_{\frac{\pi}{6}}\right\rangle$, that can be used to realise the gate $R_{\frac{\pi}{6}}$ [37], but no other circuit seems to exist in the literature.

5.2 $\mathcal{C}_{n>3}$ Resource States

The resource states generated by taking successive roots of the T gate, however, do satisfy the diagonality criteria, and thus can readily be consumed in a state injection circuit. This makes the increase in their stabilizer rank observed already at 2 qubits especially interesting, as it would suggest that these states are immediately a better candidate resource than either family of magic state.
However, this is extrapolating these results from 2 qubits, and so further work is needed to examine the stabilizer rank of these states. The `SLO` results seem to suggest a continued growth such that $\chi > \chi_{\text{magic}}$, but the power of this estimator is not well known. Given the seeming accuracy of Algorithm 1 up to $|H^{\otimes 4}\rangle$, this may serve as a better technique for studying the $\mathcal{C}_{n>3}$ states at higher n.

5.3 Outlook for Quantum Computing

As is clear form the case of arbitrary states, and from the lack of a known circuit for employing face-type magic states, a larger stabilizer rank does not immediately translate to a new, stronger resource for fault tolerant quantum computing.
There are several related questions that have to be considered when proposing a candidate resource for a fault tolerant scheme. Firstly, there is the question of consuming the resource; how readily can we employ it in our computation?

Secondly, we then have to consider the problem of preparing the resource fault tolerantly. For magic states, distillation schemes have been known and developed since 2005 [37]. In particular, magic state distillation is based on performing stabilizer measurements on multiple, faulty copies, effectively performing quantum error correction to clean up errors in the preparation [37]. That this technique converges for magic states could perhaps be a direct consequence of their definition as symmetric points on the Bloch sphere between stabilizers.
However, distillation schemes for alternative resource states are known. The most well developed alternative is Toffoli state distillation and injection, which was first proposed by Shor in the earliest papers on fault tolerant computing [54].
In general, these Toffoli distillation schemes are slightly more resource intensive than magic state distillation [55]. Most modern fault-tolerant proposals instead build a Toffoli gate in a circuit consuming $4-5$ magic states [50]. There seems to be a connection, then, between the computational power of the Toffoli gate (equivalent to several T gates) and the resources required to distil the corresponding ancilla. This suggests that distillation schemes for alternative resource states likely exist, but that they may not be as efficient as magic state distillation.

The final question we have to consider is the problem of circuit synthesis: what is the number of 'expensive' gate operations required to implement a given computation in our chosen universal gate set? Current circuit synthesis techniques are best developed for the Clifford+T basis. There is known universal result, the Solovay-Kitaev theorem, for generating any arbitrary unitary in any gate set, but the resulting circuit complexity is known to be well in excess information theoretic bounds. Thus, the question of what resources different Clifford+R schemes require to implement a given computation remains open.
Some indications come from current fault-tolerant schemes, which require a certain number of T gates to realise other unitary operations, such as the Toffoli example given above, but it is not yet known if these realisations are optimal.

Once continued development in scalable qubit manufacturing and control enables the construc-

tion large quantum devices, these difference in resource costs will become less and less important; we will have sufficiently large physical resources that any universal construction will be realisable. However, in the near term, the resource costs represent a significant hurdle.

The PBC model of computing was derived by Bravyi, Smolin & Smith in the context of trying to compile a quantum computation for implementation on a small quantum device with a fixed number of qubits, $O(10^2)$. Initially, it might seem that the reduction of the circuit to a measurement on t magic states represents a significant resource saving. But the rapid growth in the T-count of a circuit already discussed reveals the challenge of choosing a weak universal gate set.

There is strong evidence from the work of Bravyi et al. and these results that the stabilizer rank of the edge-type magic states is minimal. Taken together with the existence of circuits requiring multiple T gates to implement other useful $\mathcal{C}_3$ gates, does suggest that the Clifford+T basis is perhaps the most resource intensive choice for a universal fault tolerant design. The Toffoli, in particular, is a useful gate because it is universal for classical computation, making it easy to translate a classical circuit to a quantum device using this gate. The high T-count of the Toffoli is precisely the reason for the blow-up in T-count in Shor's algorithm [56].

Thus, the problem of developing alternative fault tolerant schemes is related the problem of extracting as much computational power as possible from our early quantum devices.

Chapter 6

Conclusion and Future Work

In this report, we have discussed the role of classical descriptions and simulations of quantum mechanics as a tool to try and understand the origins of quantum speedup. In particular, we focused on a novel simulation technique developed by Bravyi, Gosset, Smolin & Smith.
This technique does not admit an efficient classical simulation of quantum computation, which is believed to be impossible. Instead, it is able to significantly reduce the classical complexity of simulating quantum circuits built using the Clifford+T gate set. This particular gate set lies at the heart of modern schemes for fault tolerant quantum computation.

This reduction in complexity derives from the decomposition of the edge-type magic states used to realise T gates into a sum of χ stabilizer states, where χ is called the stabilizer rank. The stabilizer rank seems to be minimal for these edge-type states; combined with existing observations about the T-count of other quantum gates, and the stabilizer rank of arbitrary quantum states, we argue that computational savings achieved by the stabilizer rank method are related to the value of the state as a resource for fault tolerant computing.
Given this interpretation, we examined the behaviour of the stabilizer rank for alternative resource states: the class of face-type magic states, and resource states for gates in higher levels of the Clifford hierarchy.

The results seem to confirm that the edge-type magic states have the smallest stabilizer rank, and imply that there could be an advantage to building a fault tolerant scheme around alternative resource states, including gates from $\mathcal{C}_{n>3}$. Further work on adapting circuit synthesis techniques for different universal gate sets would be required to extend these results to give clear proposals for building quantum devices.

Future Work

These results, especially concerning $\mathcal{C}_{n>3}$ resource states, are preliminary, mostly due to computational constraints in finding the stabilizer rank decompositions with the brute-force search methods. The numerical techniques employed thus need further development.

The programme used to find the stabilizer states could be significantly improved in several ways. The simplest point is an implementation detail: rewriting the programme in `C` would allow finer memory management, reducing the footprint of the programme.
A more significant improvement would be to use an alternative representation of the stabilizer states, that does not require building the Pauli operators in each group and explicitly solving for the corresponding +1 eigenstate. This representation extends the use of binary subspaces on $\mathbb{F}_2^n$ to include binary quadratic functions [57]. It can then be shown that a stabilizer state is characterised

by a linear subspace $\mathcal{L} \subseteq \mathbb{F}_2^n$, a quadratic function $q : \mathbb{F}_2^n \to \mathbb{F}_2$ and a 'shift' vector $t \in \mathbb{F}_2^n$ [57, 58]. This is in fact the computational representation used by Bravyi & Gosset in their implementation of the stabilizer rank method. Combined with a better method of generating these linear subspaces than brute-force search, this would allow stabilizer states to be generated much more efficiently. Alternatively, given the good correspondence between the simulated annealing method and brute-force up to 4 qubits, switching to a simulated annealing search would allow the numerical analysis to be extended. The significant advantage here is that we only need to be able to generate χ unique, random n-qubit stabilizer states, rather than the full set. The other points in the phase space are then generated by applying Pauli operators to random states in the decomposition set $\tilde{\phi}$, as outlined in Algorithm 1.

Extending the technique used by Bravyi & Gosset to bound the stabilizer rank beyond magic states would also allow us to find asymptotic bounds for the rank of the $\mathcal{C}_{n>3}$ states, to compliment further numerical work.

It would also be useful to understand the computational complexity of determining the stabilizer rank of a given state. This problem is evidently in **NP**, as any candidate decomposition can be checked in polynomial time by building the projector and evaluating the overlap.
There is also an indication that the problem is **NP**-hard. For example, finding the sparsest vector $x : Ax = b$ is known to be **NP**-hard [53]. Setting b as the computational basis representation and A as the transformation between the computational and stabilizer state bases, as constructed in Section 4.2.3, makes finding the sparsest solution x equivalent to finding χ. An alternative proof strategy could be based on the known **NP**-hardness of finding sparse decompositions of doubly-stochastic matrices [59].

Finally, it would also be interesting to improve on the use of the `SLO` method to estimate the stabilizer rank. While the ℓ_0 minimization problem is known to be **NP**-hard, the related ℓ_1 problem is in fact solvable exactly [50]. It is also known that under certain conditions, the global minima of the ℓ_1 and ℓ_0 problems coincide [60]. Thus, it would be interesting to see if this heuristic can be used to solve for stabilizer rank by calculating the 0-norm of solutions found using ℓ_1 minimization techniques.

Acknowledgements

I would like to thank my supervisor Dr. Dan Browne for his discussions and input throughout the project, and Dr. Mark Howard for sharing his slides on the robustness measure for quantifying 'magicness'. I would also like to thank the staff and students of the EPSRC CDT in Delivering Quantum Technologies for their training and support.

References

[1] D. Deutsch. Quantum Theory, the Church-Turing Principle and the Universal Quantum Computer. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, **400**, 97 (1985). arXiv:quant-ph/0407008.

[2] R. Jozsa and N. Linden. On the role of entanglement in quantum-computational speed-up. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, **459**, 2011 (2003).

[3] M. Howard, J. Wallman, V. Veitch *et al.* Contextuality supplies the 'magic' for quantum computation. *Nature*, **510**, 351 (2014). 1401.4174.

[4] S. Aaronson and D. Gottesman. Improved simulation of stabilizer circuits. *Physical Review A - Atomic, Molecular, and Optical Physics*, **70**, 1 (2004). quant-ph/0406196.

[5] S. Bravyi and D. Gosset. Improved Classical Simulation of Quantum Circuits Dominated by Clifford Gates. *Physical Review Letters*, **116**, 250501 (2016). 1601.07601.

[6] S. Bravyi, G. Smith, and J. Smolin. Trading classical and quantum computational resources. page 14 (2015). 1506.01396.

[7] R. P. Feynman. Simulating physics with computers. *International Journal of Theoretical Physics*, **21**, 467 (1982). quant-ph/9508027.

[8] P. Benioff. Quantum mechanical Hamiltonian models of computers. *Annals NY Academic of Science*, **480**, 475 (1986).

[9] S. Wiesner. Conjugate coding. *ACM SIGACT News*, **15**, 78 (1983). arXiv:1011.1669v3.

[10] G. Brassard. Brief history of quantum cryptography: a personal perspective. In *IEEE Information Theory Workshop on Theory and Practice in Information-Theoretic Security, 2005.*, volume 2005, pages 19–23. IEEE (2006). quant-ph/0604072.

[11] B. Schumacher. Quantum coding. *Physical Review A*, **51**, 2738 (1995).

[12] R. Jozsa. Entanglement and Quantum Computation. *Foundations of Science*, **84**, 11 (1997). quant-ph/9707034.

[13] D. Z. Albert. On quantum-mechanical automata. *Physics Letters A*, **98**, 249 (1983).

[14] D. Deutsch and R. Jozsa. Rapid Solution of Problems by Quantum Computation. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, **439**, 553 (1992).

[15] C. H. Bennett, E. Bernstein, G. Brassard *et al.* Strengths and Weaknesses of Quantum Computing. *SIAM Journal on Computing*, **26**, 1510 (1997). quant-ph/9701001.

[16] E. Bernstein and U. Vazirani. Quantum Complexity Theory. *SIAM Journal on Computing*, **26**, 1411 (1997).

[17] A. C.-C. Yao. Quantum circuit complexity. *Foundations of Computer Science, 1993. Proceedings., 34th Annual Symposium on*, pages 352–361 (1993).

[18] D. Simon. On the power of quantum computation. In *Proceedings 35th Annual Symposium on Foundations of Computer Science*, pages 116–123. IEEE Comput. Soc. Press (1998).

[19] P. W. Shor. Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer. *SIAM Journal on Computing*, **26**, 1484 (1997). quant-ph/9508027.

[20] M. A. Nielsen and I. L. Chuang. *Quantum computation and quantum information* (2000).

[21] S. Aaronson. Complexity Zoo. https://complexityzoo.uwaterloo.ca/Complexity_Zoo:N#npc.

[22] L. K. Grover. A fast quantum mechanical algorithm for database search. In *Proceedings of the twenty-eighth annual ACM symposium on Theory of computing - STOC '96*, pages 212–219. ACM Press, New York, New York, USA (1996). quant-ph/9605043.

[23] A. S. Holevo. Some estimates for information quantity transmitted by quantum communication channels,". *Problems of Information Transmission*, **9**, 177 (1973).

[24] R. Laflamme, D. G. Cory, C. Negrevergne *et al.* NMR Quantum Information Processing and Entanglement. *Quantum*, **1**, 11 (2001). quant-ph/0110029.

[25] A. Ekert and R. Jozsa. Quantum algorithms: entanglement-enhanced information processing. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, **356**, 1769 (1998). quant-ph/9803072.

[26] S. Braunstein, C. Caves, R. Jozsa *et al.* Separability of Very Noisy Mixed States and Implications for NMR Quantum Computing. *Physical Review Letters*, **83**, 1054 (1999). quant-ph/9811018.

[27] N. Linden and S. Popescu. Good Dynamics versus Bad Kinematics: Is Entanglement Needed for Quantum Computation? *Physical Review Letters*, **87**, 047901 (2001). quant-ph/9906008.

[28] S. L. Braunstein and A. K. Pati. Speed-up and entanglement in quantum searching. **1**, 1 (2000). quant-ph/0008018.

[29] G. Vidal. Efficient classical simulation of slightly entangled quantum computations. *Physical review letters*, **91**, 147902 (2003).

[30] D. Gottesman. The Heisenberg Representation of Quantum Computers. *Proceedings of the XXII International Colloquium on Group Theoretical Methods in Physics*, **1**, 32 (1999). quant-ph/9807006.

[31] M. Van Den Nest. Universal quantum computation with little entanglement. *Physical Review Letters*, **110**, 1 (2013). 1204.3107.

[32] D. Gottesman. Stabilizer codes and quantum error correction. *arXiv preprint quant-ph/9705052*, **2008**, 114 (1997). quant-ph/9705052.

[33] D. Gottesman. An Introduction to Quantum Error Correction and Fault-Tolerant Quantum Computation. *Proceedings of Symposia in Applied Mathematics*, **0000**, 1 (2009). 0904.2557v1.

[34] B. Eastin and E. Knill. Restrictions on Transversal Encoded Quantum Gate Sets. *Physical Review Letters*, **102**, 110502 (2009). 0811.4262.

[35] I. L. Chuang and D. Gottesman. Quantum Teleportation is a Universal Computational Primitive. *Nature*, **402**, 390 (1999). quant-ph/9908010.

[36] X. Zhou, D. W. Leung, and I. L. Chuang. Methodology for quantum logic gate construction. *Physical Review A - Atomic, Molecular, and Optical Physics*, **62**, 052316 (2000). arXiv:quant-ph/0002039v2.

[37] S. Bravyi and A. Kitaev. Universal quantum computation with ideal Clifford gates and noisy ancillas. *Physical Review A - Atomic, Molecular, and Optical Physics*, **71** (2005). quant-ph/0403025.

[38] R. W. Spekkens. Evidence for the epistemic view of quantum states: A toy theory. *Physical Review A*, **75**, 032110 (2007).

[39] H. J. Garcia, I. L. Markov, and A. W. Cross. Efficient Inner-product Algorithm for Stabilizer States. *arXiv preprint arXiv:1210.6646*, pages 1–14 (2012). 1210.6646.

[40] H. J. Garcia and I. L. Markov. Simulation of Quantum Circuits via Stabilizer Frames. *IEEE Transactions on Computers*, **64**, 2323 (2015).

[41] V. Veitch, C. Ferrie, D. Gross *et al.* Negative quasi-probability as a resource for quantum computation. *New Journal of Physics*, **14** (2012). 1201.1256.

[42] H. Pashayan, J. J. Wallman, and S. D. Bartlett. Estimating Outcome Probabilities of Quantum Circuits Using Quasiprobabilities. *Physical Review Letters*, **115**, 1 (2015). 1503.07525.

[43] N. Delfosse, P. Allard Guerin, J. Bian *et al.* Wigner Function Negativity and Contextuality in Quantum Computation on Rebits. *Physical Review X*, **5**, 1 (2015). 1409.5170.

[44] J. Preskill. Quantum Shannon Theory (2016). 1604.07450.

[45] P. Selinger. Efficient Clifford+T approximation of single-qubit operators (2012). 1212.6253.

[46] J. R. Johansson, P. D. Nation, and F. Nori. QuTiP: An open-source Python framework for the dynamics of open quantum systems. *Computer Physics Communications*, **183**, 1760 (2012). 1211.6518.

[47] M. Howard, E. Brennan, and J. Vala. Quantum contextuality with stabilizer states. *Entropy*, **15**, 2340 (2013). arXiv:1501.04342v1.

[48] S. van der Walt, S. C. Colbert, and G. Varoquaux. The NumPy Array: A Structure for Efficient Numerical Computation. *Computing in Science & Engineering*, **13**, 22 (2011).

[49] G. Vidal and R. Tarrach. Robustness of entanglement. *Physical Review A*, **59**, 141 (1999). arXiv:quant-ph/9806094v1.

[50] M. Howard. Magic State Quantum Computing and Contextuality. In *Contextuality as a Resource for Quantum Computation*, June. University College London (2016).

[51] A. Acín, N. Gisin, and L. Masanes. From Bell's Theorem to Secure Quantum Key Distribution. *Physical Review Letters*, **97**, 120405 (2006).

[52] H. Mohimani, M. Babaie-Zadeh, and C. Jutten. A Fast Approach for Overcomplete Sparse Decomposition Based on Smoothed L0 Norm. *IEEE Transactions on Signal Processing*, **57**, 289 (2009). 0809.2508.

[53] D. Ge, X. Jiang, and Y. Ye. A note on the complexity of ℓ_0 minimization. *Mathematical programming*, **129**, 285 (2011).

[54] P. Shor. Fault-tolerant quantum computation. In *Proceedings of 37th Conference on Foundations of Computer Science*, pages 56–65. IEEE Comput. Soc. Press (1996). quant-ph/9605011.

[55] B. Eastin. Distilling one-qubit magic states into Toffoli states. *Physical Review A*, **87**, 032321 (2013). quant-ph/1212.4872.

[56] A. Fowler, C. Wellard, and L. Hollenberg. Error rate of the Kane quantum computer controlled-NOT gate in the presence of dephasing. *Physical Review A*, **67**, 012301 (2003). arXiv:quant-ph/0207103v2.

[57] J. Dehaene and B. De Moor. The Clifford group, stabilizer states, and linear and quadratic operations over GF(2). *ArXiv*, page 9 (2003). quant-ph/0304125.

[58] D. Gross and M. V. D. Nest. The LU-LC conjecture, diagonal local operations and quadratic forms over GF(2). *Quantum Inf. Comput.*, **8**, 263 (2007). 0707.4000.

[59] F. Dufossé and B. Uçar. Notes on Birkhoff-von Neumann decomposition of doubly stochastic matrices. *Linear Algebra and its Applications*, **497**, 108 (2016).

[60] H. Sawada, S. Makino, and S. Winter. On Real and Complex Valued 1 -Norm Minimization for Overcomplete Blind Source Separation. *Signal Processing*, pages 86–89 (2005).