The Classical Simulation of Noisy Quantum Computers: A Polyhedral Approach

Nikhil Ratanje

12th April 2017

A thesis submitted to the Department of Physics
University of Strathclyde for the degree of
Doctor of Philosophy

Declaration

This thesis is the result of the author's original research. It has been composed by the author and has not been previously submitted for examination which has led to the award of a degree.' 'The copyright of this thesis belongs to the author under the terms of the United Kingdom Copyright Acts as qualified by University of Strathclyde Regulation 3.50. Due acknowledgement must always be made of the use of any material contained in, or derived from, this thesis.

Signed: 

Date: 12/04/2017

Acknowledgments

I would firstly like to give special thanks to Einar Pius, M. Hamed Mohammady, Jochen Bruckbauer, and Amit Ratanje for taking the time to proof-read this thesis - the feedback they gave me was helpful and greatly appreciated.

My time at Strathclyde has been a defining period in my life and I am deeply grateful to all the friends I made in Glasgow and Edinburgh for making Scotland a home away from home for me. I have left Glasgow with many cherished memories and treasured friendships.

I would like to thank the Semiconductor Spectroscopy & Devices Group at Strathclyde for all of their tea and cake I've consumed over the years - it was always appreciated!

Many thanks to Vedran Dunjko, Václav Potoček and Gergely Ferenczi for the deeply insightful conversations, to Ben Hourahine for his advice and support, and to Daniel Oi and John Jeffers for taking over supervision of me, seeing me through the final stretch.

I cannot express my gratitude enough to my supervisor Shashank Virmani for his mentorship and guidance throughout the PhD. It has been a long journey and I am truly thankful for all of the hours of support he gave me.

Finally, I would like to thank my parents for giving me more than anyone could ever hope for, and always being there for me.

Abstract

In this thesis we explored the consequences of considering generalised non-quantum notions of entanglement in the classical simulation of noisy quantum computers where the available measurements are restricted. Such noise rates serve as upper bounds to fault tolerance thresholds. These measurement restrictions come about either through imperfection, and/or by design to some limited set. By considering sets of operators that return positive measurement outcome probabilities for the restricted measurements, one can construct new single particle state spaces containing quantum and non-quantum operators. These state spaces can then be used with a modified version of Harrow and Nielsen's classical simulation algorithm to efficiently simulate noisy quantum computers that are incapable of generating generalised entanglement with respect to the new state spaces.

Through this approach we developed alternative methods of classical simulation, strongly connected to the study of non-local correlations, in that we constructed noisy quantum computers capable of performing non-Clifford operations and could generate some forms of multiparty quantum entanglement, but were classical in that they could be efficiently classically simulated and could not generate non-local statistics.

We focused on magic state quantum computers (that are limited to only Pauli measurements), with ideal local gates, but noisy control-Pauli Z gates, and calculated the noise needed to ensure the control-Z gates became incapable of generating generalised entanglment for a variety of noise models and state space choice, with the aim of finding an optimal single particle state space requiring the least noise to remove the generalised entanglement. The state spaces were required to always return valid measurement probabilities, this meant they also had had to have octahedral symmetry to ensure local gates did not take states outside the state space. While we able to find to the optimal choice for highly imperfect measurements, were we unable to find the optimal in all cases. Our best candidate state space required less joint depolarising noise at $\approx 56\%$ in comparison to noise levels of $\approx 67\%$ required if the algorithm used quantum notions of separability. This suggests that generalised entanglement may offer more insight than quantum entanglement when discussing the power of Clifford operation based quantum computers.

List of publications

Generalized state spaces and nonlocality in fault tolerant quantum-computing schemes

N. Ratanje, S. Virmani, Phys. Rev. A 83 (2011) 032309. doi:10.1103/PhysRevA.83.032309.

URL http://link.aps.org/doi/10.1103/PhysRevA.83.032309

Exploiting non-quantum entanglement to widen applicability of limited-entanglement classical simulations of quantum systems

N. Ratanje, S. Virmani, (2012), arXiv:1201.0613.

Contents

1 Introduction and preliminaries

1.1 Introduction

Quantum computers are information processing devices that appear to be capable of solving certain computational tasks more efficiently than any currently conceivable classical computer. Solving a task more efficiently means fewer time and space resources are required to solve the task [1]. It is important to consider how these resources scale as the input size of the task grows in size, i.e. how many more steps does it take to finish and how much more resources are used. For this thesis it is sufficient to consider a computation to be efficiently solvable by a device if the resources and time required to complete depend polynomially on the input size. Computational tasks that are hard, i.e. not efficient, require exponentially more resources when the input size of the task grows.

There remain many unresolved questions about the power of quantum computation; for example, what types of computational tasks can be efficiently solved on a quantum computer, but not on a classical computer? What features make a quantum computer more powerful? Intuitively, the difficulties related to simulating naturally occurring quantum mechanical systems on classical computers can help us answer these questions. Whether classical computers can efficiently simulate ideal quantum computers has yet to be proven, although it is not believed to be true.

It is widely believed that a key component to the power of quantum computers is the ability of quantum systems to generate entangled states. A common approach to building simulations of physical systems on classical computers is to use divide and conquer techniques by taking a computational task and breaking it down into two or more smaller sub-problems. The evolution of each of these sub-problems can be modelled individually, before combining their solutions to obtain a final solution to the original problem. Such techniques, in general, cannot be used on systems with entangled states, as such states cannot be split into subsystem states. In a sense entangled states contain more information as a whole than as a composite of their constituent sub-systems.

A key concept in discussing quantum computation is the idea of *universal quantum computers* (UQC), which will be described in greater detail later in the thesis. For now, any quantum computer capable of efficiently simulating any other quantum device to

an arbitrary accuracy can be thought of as being universal. While there exist universal quantum computers composed of different gate sets, of particular relevance to later work in the thesis are magic state architectures [2] which we consider as a particular example of gate model quantum computers.

Quantum computers are prone to errors made during computation. In particular, they are highly susceptible to the effects of environmental interaction, a phenomena called decoherence. Such environmental interactions disturb quantum states in the computation, collapsing the states and destroying information contained in the superpositions of the states. Such environmental interactions leak information from system leading to errors and the failure of the computation. It is therefore essential for quantum computers to have a method of controlling and correcting these errors. This came about with the development of *quantum error correcting codes* (QECC) [1, 3], which provided a way of spreading out the information contained within the states of the system by encoding them onto new states over more qubits, in a similar way to redundancy in classical error correction. Errors that may have occurred during any stage of computation could be identified and corrected before decoding the state. QECC therefore provides a way of protecting stored information within our quantum computer.

In addition to storage errors, the fallibility of the quantum logic gates manipulating the information in the qubits contributes additional errors to the computation. It is possible to encode the individual logic gates forming the quantum computer using the QECC, in such a way that for low enough physical error rates the action of the encoded gates on the encoded states produces the same decoded output as applying the errorless gates on the states. Using such techniques failure rates may be reduced in quantum computation. If this failure rate can be brought sufficiently low to efficiently simulate quantum computers then the circuit is called *fault tolerant* [1, 3, 4], and the quantum device will work effectively despite having non-ideal elementary components. The upper noise rate for which fault tolerant quantum computation can be achieved is called the *noise threshold*. In other words, a noisy quantum computer with error rates below the noise threshold can efficiently simulate an ideal universal quantum computer. The value of the noise threshold varies greatly depending on the gate architecture and noise models, as well as the threshold calculation method.

Instead of investigating whether noisy quantum devices can simulate ideal ones, we can consider whether the noisy quantum device can be chosen to be so noisy that its quantum features enabling powerful computation are lost, making it efficiently simulated on a classical computer. The rate of noise at which this happens can be used as an *upper bound* to the noise threshold, as the computation would offer no benefit over classical computation, whereas specific fault tolerance designs give us lower bounds to the threshold.

The noise threshold provides a way of identifying three regions of quantum/classical simulatability based on the bounds on the value of the threshold; the first, where the

noise rate is below the lower bound gives a region where noisy quantum computers can simulate an idealised universal quantum computer. The second describes an intermediary region where error correction techniques cannot be used to allow for efficient simulation of a UQC, nor can the noisy quantum system be simulated efficiently by a classical computer. The third and final region is where the noise rate is greater than the upper bound, in such a case a classical computer can efficiently simulate the noisy quantum computer.

In almost all cases the exact values of these lower and upper bounds are unknown and determining these bounds is a difficult task. This thesis looks to find new values for the upper threshold bound for a gate architecture with well known fault-tolerant error correction schemes, for a selection of simple but realistic noise models. Two relevant methods for classically simulating noisy quantum computers will be outlined. The first method [5] argues that sufficient noise in the quantum computer takes all of the operations and states in the noisy system to the restricted classically simulatable Clifford set [3]. The second method is based on an algorithm by Harrow and Nielsen [6] and is applicable to systems composed of operations incapable of generating entanglement. Their classical algorithm can efficiently sample from a probability distribution for the available measurements arbitrarily close to that from the noisy quantum computer. In other words the noisy quantum computer in such instances may be efficiently classically simulatable.

This thesis presents work from the following two papers [7, 8], which considered a modified version of the Harrow and Nielsen algorithm applied to "magic state architecture" devices based on gates from the Clifford set. Such a device has an innate restriction on the available measurements opening up the question of whether non-quantum entanglement can strengthen the Harrow and Nielsen algorithm. The work presented instead takes a cue from generalised probability theories and looks at new sub-divisions of system that include non-quantum states, based on the available measurements. These sub-divisions, or new single particle state spaces, allow for a definition of a new form of entanglement based on them. We also considered specifying the single particle state space based on restrictions from fault on input state preparation and/or the final state measurements.

The initial aim of the work was to show that Harrow and Nielsen algorithm, using this redefined single particle state space could (a) provide new regimes that could be efficiently classically simulated, and (b) obtain lower upper threshold bounds. This objective was achieved for a variety of single particle state spaces based on different preparation and measurement restrictions. This lead us to search for the single particle state space(s) giving the optimal bounds on the threshold value. Whilst we were unable to find the optimal for all kinds of preparation/measurement fault, we were able to specify a region containing the optimal value. We finally present the single particle state space corresponding to the best threshold value we were able to attain.

This thesis will have the following structure:

In the second half of this chapter we will introduce some key concepts in quantum mechanics and quantum information that will be used later in the thesis.

Chapters (2) and (3) will cover the main background topics; introducing the reader to quantum error correction, the Gottesman-Knill theorem, the threshold theorem, and magic state architectures in Chapter (2), and classical simulation of noisy quantum computers the Harrow and Nielsen algorithm in Chapter (3).

In Chapter (4) we will specify how restricted measurements relate to new single particle state spaces. The connections between non-locality and separability with respect to our redefined state space will be discussed for the specific case of the "Bloch cube" state space, before stating the main technical problem of the research.

Chapter (5) looks at the results from [7] where we determined the threshold values for state spaces corresponding to the Bloch sphere modified for preparation/measurement fault, and the Bloch cube state space.

Chapter (6) will present results from [8], relating to our best candidate - the "truncated cube" state space. We will give a give overview of linear optimisation and the simplex method, before showing how we used linear optimisation, for a specific case, to determine the threshold value and sufficiency of the result.

In Chapter (7) we will construct necessity bounds based on restrictions to the norm lengths of "Bloch vectors" of operators in single particle state spaces. We will show how these bounds lead us to obtain the optimal state space for measurement fault over a specific parameter value.

Chapter (8) will conclude the thesis.

1.2 Mathematical preliminaries

The remainder of this chapter will introduce some of the key concepts in quantum mechanics and quantum information relevant to the later discussions in the thesis. The reader will be assumed to have an understanding of the basic postulates of quantum mechanics and Dirac notation as found in the literature [1, 9, 10]. The following chapter will discuss the density matrix description of quantum states, the Bloch sphere representation of qubit states, POVMs, Kraus operation decompositions, the PPT criterion, and finally Choi-Jamiołkowski isomorphisms. Readers familiar with these concepts may wish to skip ahead to Chapter (3).

The following subsections will give a quick recap of linear operators before defining density operators as well as providing more details on their properties.

1.2.1 Linear operators on finite Hilbert spaces

Let a d-dimensional quantum system $\mathcal{S}$ be associated with a d-dimensional Hilbert space $\mathcal{H}_d$ with the pure states of $\mathcal{S}$ given by vectors $|\psi\rangle \in \mathcal{H}_d$. The inner product of $\mathcal{H}_d$ is written here as $\langle\phi|\psi\rangle \in \mathbb{C}$, and its norm as:

$$\|\psi\| =: \||\psi\rangle\| := \sqrt{\langle\psi|\psi\rangle} \tag{1.1}$$

A pure state $|\psi\rangle$ can be expressed as a linear combination of d orthonormal basis vectors $\{|i\rangle\}$ of $\mathcal{H}_d$:

$$|\psi\rangle = \sum_i^d \psi_i |i\rangle \qquad \text{where } \psi_i = \langle i|\psi\rangle \in \mathbb{C} \tag{1.2}$$

The dynamics of quantum states of $\mathcal{S}$ are given by operators A, defined as linear mappings $A : \mathcal{H}_d \to \mathcal{H}_d$, and written as $A|\psi\rangle =: |A\psi\rangle$. In analogy to vectors, an operator A has a corresponding dual operator $A^\dagger$ given by the *adjoint* or *Hermitian conjugate* of A, characterised by

$$\langle\phi|A\psi\rangle = \langle A^\dagger\phi|\psi\rangle \qquad \forall\ |\phi\rangle, |\psi\rangle \in \mathcal{H}_d \tag{1.3}$$

The linear operators acting on $\mathcal{H}_d$ form a complex vector space $\mathcal{L}(\mathcal{H}_d)$. An important function - the trace of an operator, $\mathrm{tr}[\cdot]$, is defined:

$$\mathrm{tr}[A] := \sum_i^d \langle i| A |i\rangle = \sum_i^d A_{ii} \tag{1.4}$$

Here the trace function is independent of the orthonormal basis $\{|i\rangle\}$ chosen. Using the operator trace we define the Hilbert-Schmidt inner product, and Hilbert Schmidt norm on $\mathcal{L}(\mathcal{H}_d)$ as:

$$\langle A|B\rangle_{HS} := \mathrm{tr}\left[A^\dagger B\right] \tag{1.5}$$

$$\|A\|_{HS} = \sqrt{\mathrm{tr}[A^\dagger A]} \tag{1.6}$$

Just as vectors of the Hilbert space can be expanded in terms of an orthonormal basis of vectors, operators in $\mathcal{L}(\mathcal{H}_d)$ can be expanded in terms of an orthonormal basis of operators using the Hilbert-Schmidt inner product. Given $\mathcal{H}_d$ with the basis $\{|i\rangle\}$, an operator A can be expressed as

$$A = \sum_{i,j}^d \langle\mathcal{B}_{ij}|A\rangle_{HS} \mathcal{B}_{ij} \tag{1.7}$$

where $\{\mathcal{B}_{ij} = |i\rangle\langle j|\}$ is a d^2-dimensional orthonormal basis in $\mathcal{L}(\mathcal{H}_d)$ and $\langle\mathcal{B}_{ij}|A\rangle_{HS} = \langle i| A |j\rangle = A_{ij}$.

An important operator that will be used extensively is the identity operator $\mathbb{1}$ on $\mathcal{H}_d$

characterised by $\mathbb{1}|\psi\rangle = |\psi\rangle$, $\forall |\psi\rangle \in \mathcal{H}_d$. Following on from eq. (1.7) the identity operator has the decomposition

$$\mathbb{1} = \sum_i^d \mathcal{B}_{ii} = \sum_i^d |i\rangle\langle i| \tag{1.8}$$

Abstract quantum states and operators on a finite Hilbert space can be represented in matrix form for some particular choice of orthonormal basis, with the ψ_i's in eq. (1.2) forming elements of a d-dimensional row vector, and the A_{ij}'s corresponding to the d^2 complex matrix elements of a $d \times d$-dimensional matrix. The linear algebra of operators on $\mathcal{H}_d$ translates directly into matrix algebra. We will term density operators in their matrix representation *density matrices*.

1.2.2 Density operator representation of quantum states

Pure quantum states are often represented by state vectors $|\psi\rangle$ on a Hilbert space, this description alone cannot characterise quantum systems that are in statistical mixtures of quantum states. Density operators are an alternative representation of quantum states that can encode all accessible information about the quantum system. The density operator representation will be used throughout this thesis.

Let ρ denote a *density operator*. Suppose $\mathcal{S}$ is in a statistical mixture of N pure states $|\psi_n\rangle \in \mathcal{H}_d$, with the likelihood of finding $\mathcal{S}$ in the pure state $|\psi_n\rangle$ given by a probability p_n. The density operator ρ for S is then defined as:

$$\rho = \sum_n^N p_n |\psi_n\rangle\langle\psi_n| \qquad \text{where } \sum_n^N p_n = 1 \tag{1.9}$$

Rather than using density operators directly, we will often use their $d \times d$-dimensional matrix representation in some chosen basis. A *density matrix* corresponding to the density operator in eq. (1.9) has complex matrix elements given as:

$$\rho_{ij} = \sum_n^N p_n \langle i|\psi_n\rangle\langle\psi_n|j\rangle \tag{1.10}$$

In general any linear operator A on $\mathcal{H}_d$ with a corresponding matrix representation is a valid density operator if for all states $|\psi\rangle \in \mathcal{H}_d$, we have $\langle\psi|\rho|\psi\rangle \geq 0$, ρ is Hermitian, and if $\mathrm{tr}[\rho] = 1$.

An important property of ρ is the *purity* of ρ given by $\mathrm{tr}[\rho^2]$. Pure states have a $\mathrm{tr}[\rho^2] = 1$, whilst for mixed states $\mathrm{tr}[\rho^2] < 1$. If $\rho = \frac{\mathbb{1}}{d}$, ρ is called maximally mixed, and ρ cannot reveal any information about the quantum system if measured.

For brevity we will often refer to states as being in some quantum state rather than referring to the density matrix as representing the quantum state.

The single particle state space or set of all density operators $\mathcal{S}(\mathcal{H}_d)$, forms a closed convex set[1], whereby any state can be expressed as a convex combination of pure states - extrema of the convex set[2]. This means that any state $\rho' \in \mathcal{S}(\mathcal{H}_d)$ can be expressed as

$$\rho' = \sum_m \lambda_m \rho_m \tag{1.11}$$

where each $\lambda_m \in [0,\, 1]$, $\sum_m \lambda_m = 1$, and every $\rho_m = |\psi_m\rangle \langle\psi_m|$ is a pure state in $\mathcal{S}(\mathcal{H}_d)$

1.2.3 Expanding density matrices

It is possible to specify a complex $d \times d$ dimensional Hermitian matrix ($A = A^\dagger$) by specifying d^2 real numbers. The set $\mathcal{R}$ of $d \times d$ dimensional Hermitian matrices form a d^2-dimensional vector space over $\mathbb{R}$ under matrix addition and real number scalar multiplication. By fixing the trace to some specific value, the number of real parameters required to specify a Hermitian matrix is reduced by one, and so the set of traceless Hermitian matrices form a $(d^2 - 1)$ dimensional subspace of the d^2 dimensional vector space.

As with eq. (1.7) a Hermitian matrix in $\mathcal{R}$ can be decomposed as a linear sum of d^2 basis vectors that are orthonormal to each other with respect to the Hilbert-Schmidt inner product. A real multiple of the $d-$dimensional identity taken together with the $d^2 - 1$ traceless and Hermitian generators of the group $SU(d)$ form an orthonormal basis for $\mathcal{R}$. A Hermitian matrix A can therefore be split into trace and traceless parts (see [11]) as:

$$A = \frac{\mathrm{tr}\,[A]}{d}\mathbb{1}_d + \frac{1}{2}\sum_{i=1}^{d^2-1} \mathrm{tr}\,[\sigma_i A]\,\sigma_i \tag{1.12}$$

where $\mathbb{1}_d$ is the d-dimensional identity matrix, and the matrices σ_i satisfy the relations:

1. $\mathrm{tr}\,[\sigma_i \sigma_j] = 2\delta_{ij}$
2. $\sigma_i = \sigma_i^\dagger$
3. $\mathrm{tr}\,[\sigma_i] = 0$

The density matrix ρ, as a Hermitian matrix can therefore be written as:

$$\rho = \frac{1}{d}\left(\mathbb{1}_d + \sum_{i=1}^{d^2-1} r_i \sigma_i\right) \tag{1.13}$$

where $\underline{r} = (r_1,\, r_2,\, \ldots\, ,\, r_{d^2-1})^{\mathrm{T}}$. The vector $\underline{r} \in \mathbb{R}^{d^2-1}$ will be referred to as a *Bloch vector.* The structure of the set of valid density matrices comes as restrictions on

[1] A set A is said to be *convex* if for all x_1,$x_2 \in A$, and $\lambda \in [0, 1]$, the sum $\lambda x_1 + (1 - \lambda)x_2 \in A$. The convex hull of a set of points B is the smallest convex set containing B.

[2] Another way of saying this is that the set $\mathcal{S}(\mathcal{H}_d)$ is the convex hull of all pure states $|\psi\rangle \langle\psi|$, $|\psi\rangle \in \mathcal{H}_d$.

the l_2 norm lengths and "directions" on the Bloch vectors, with quantum pure states corresponding to the density matrices with the longest Bloch vectors equal to 1, and < 1 for mixed states.

1.2.4 Multiparticle systems and entanglement and PPT criterion

This subsection will briefly introduce multipartite quantum systems and entanglement before finally describing the PPT criterion as a means to decide whether a pair of quantum subsystems are entangled or not.

When combining two or more quantum systems, the Hilbert space of the compound system is given by the tensor product of the constituent parts. The tensor product Hilbert space $\mathcal{H}^{AB}$ of two sub-system Hilbert spaces $\mathcal{H}^A$ and $\mathcal{H}^B$, whose dimensions need not be the same, can be expressed as:

$$\mathcal{H}^{AB} = \mathcal{H}^A \otimes \mathcal{H}^B \tag{1.14}$$

where the tensor product satisfies scalar multiplication and associativity on both sides. A state in $\mathcal{H}_{AB}$ can be expressed in terms of states in the individual sub-systems . For a pair of pure states $|\psi\rangle^A \in \mathcal{H}^A$ and$|\phi\rangle^B \in \mathcal{H}^B$ there exists a product state in $\mathcal{H}^{AB}$ denoted in the following equivalent ways:

$$|\psi\rangle^A \otimes |\phi\rangle^B \equiv |\psi\rangle^A |\phi\rangle^B \equiv |\psi,\, \phi\rangle^{AB} \tag{1.15}$$

Where the sets $\left\{|i\rangle^A\right\}$ and $\left\{|j\rangle^B\right\}$ form an orthonormal basis for $\mathcal{H}^A$ and $\mathcal{H}^B$, the set of products $\left\{|i,\, j\rangle^{AB}\right\}$ form an orthonormal basis for $\mathcal{H}^{AB}$.

Linear operators acting on the tensor product Hilbert space act naturally as follows; for linear operators $A \otimes B$ acting on $\mathcal{H}^{AB}$, the pure product state $|\psi\rangle = \sum_i \alpha_i |u_i\rangle \otimes |v_i\rangle \in \mathcal{H}^{AB}$, $\alpha_i \in \mathbb{C}$ transforms as follows:

$$(A \otimes B)(\sum_i \alpha_i |u_i\rangle \otimes |v_i\rangle) \equiv \sum \alpha_i A |u_i\rangle \otimes B |v_i\rangle \tag{1.16}$$

This expression also ensures the linearity of $A \otimes B$. The density matrix for a given compound system state contains not only information about the collective system but also provides us with a way of describing the sub-systems composing the whole quantum system. The description of the sub-system is provided by the *reduced density operator.* Suppose for systems A and B whose states is described by the density matrix ρ^{AB} the reduced density operator for system A is denoted by:

$$\rho^A \equiv tr_B[\rho^{AB}] \tag{1.17}$$

The operation $tr_B[\cdot]$ is called the *partial trace* over system B. For the states $|u_1\rangle$, $|u_2\rangle \in$

$\mathcal{H}^A$ and $|v_1\rangle, |v_2\rangle \in \mathcal{H}^B$ the partial trace operation over B is defined by:

$$tr_B[|u_1\rangle\langle u_2| \otimes |v_1\rangle\langle v_2|] \equiv |u_1\rangle\langle u_2| \, tr[|v_1\rangle\langle v_2|] \tag{1.18}$$

The trace operation $tr[|v_1\rangle\langle v_2|]$ in eq. (1.18) has its usual meaning. The partial trace operation is defined to be linear in its input. Suppose a composite quantum system is in the pure state $\rho^{AB} = \varrho^A \otimes \sigma^B$, where $\varrho^A \in \mathcal{H}^A$, $\sigma^B \in \mathcal{H}^B$. Taking the partial trace of ρ^{AB} over system B can be calculated as:

$$tr_B[\rho^{AB}] = tr_B[\varrho^A \otimes \sigma^B] = \varrho^A tr[\sigma^B] = \varrho^A \tag{1.19}$$

Similarly the partial trace over the system A can be calculated as follows:

$$tr_A[\rho^{AB}] = tr_A[\varrho^A \otimes \sigma^B] = tr[\varrho^A]\sigma^B = \sigma^B \tag{1.20}$$

The tensor product of the remaining sub-systems ϱ^A and σ^B intuitively recreates the product state ρ^{AB}. Hence the tensor product of the sub-systems of the state ρ^{AB} contain the full details of the measurement statistics of the composite state ρ^{AB}. The reduced density matrix ϱ^A in this example provides the correct measurement statistics for the system A, and we can identify the density matrix ϱ^A corresponding to the state of the system A and σ^B for B. The sum of the information about the individual sub-systems can provide a full description of the whole composite system, but this is not always the case as we will see in the next example.

Let us consider the pure quantum state - the Bell state $|\psi_+\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$ with the following density matrix

$$\rho = \frac{1}{2}(|00\rangle\langle 00| + |11\rangle\langle 00| + |00\rangle\langle 11| + |11\rangle\langle 11|) \tag{1.21}$$

If we trace out the second system, we will arrive at the reduced density matrix $\rho_1 = \frac{\mathbb{1}}{2}$ - the maximally mixed state. This Bell state is an example of a special type of multi-particle quantum state that is called an *entangled state*, as shown in the example these states in a sense contain more information as a whole than the information gained from the sub-states together. We call all non-entangled states *separable states*. Entangled pure states like this Bell state are defined to be those that cannot be written as products of individual subsystems. Bipartite separable mixed states are defined as those that can be written as:

$$\rho^{AB} = \sum_i p_i \rho_i^A \otimes \rho_i^B \tag{1.22}$$

where $\sum_i p_i = 1$, and the states $\left\{\rho_i^A\right\}$ and $\left\{\rho_i^B\right\}$ are density matrices of the respective sub-systems.

Where $\left\{\sigma_i^A \otimes \sigma_j^B\right\}$ is some orthonormal tensor product basis we can in general express a bipartite density matrix as:

$$\rho = \sum_{ij} p_{ij} \sigma_i^A \otimes \sigma_j^B \tag{1.23}$$

A tool used to determine when a given bipartite product state is entangled or separable is called the PPT (positive partial transpose) criterion [12]. We will start by defining the *partial transposition* over system B as the following:

$$\rho^{T_B} = \sum_{ij} p_{ij} \sigma_i^A \otimes \left(\sigma_j^B\right)^T \tag{1.24}$$

The PPT criterion states that if a quantum state ρ is separable then it is necessary that its partial trace ρ^{T_B} is a positive operator. In the case where ρ corresponds to a quantum state in a 2×2 or 2×3 dimensional Hilbert space, then the positivity of ρ^{T_B} is also sufficient for the separability of ρ. In cases where ρ is a state in a higher dimensional Hilbert space the PPT criterion no longer becomes sufficient for separability, that is in higher dimensions there exist density matrices corresponding to entangled states that are positive and PPT.

For composite quantum systems composed of more than two sub-systems describing entanglement becomes a considerably difficult and complex task. We will later discuss many-particle systems that are entangled across pairs of sub-systems but separable across others. For example, say we had a four qubit system ρ^{ABCD}, with sub-systems denoted A, B, C, D, and the sub-systems A and B, C, and D in two-way entanglement, the state ρ^{ABCD} is said to be *separable with respect to the* $AB : CD$ *cut* if ρ^{ABCD} can be expressed as:

$$\rho^{ABCD} = \sum_j p_j \rho_j^{AB} \otimes \rho_j^{CD} \tag{1.25}$$

Where ρ^{AB}, and ρ^{CD} correspond to sub-systems AB, and CD respectively. Note that by expressing ρ^{ABCD} as in eq. (1.25) we are also saying that there is no entanglement between the sub-systems (AB) and (CD).

1.2.5 Measurements of quantum states

In describing a quantum measurement we require rules describing (a) the measurement statistics obtained from the measurement, (b) a description of the potential post-measurement state of the system. In situations where the post-measurement state is of little interest we may use POVM (Positive Operator-Valued Measure) formalism to analyse the measurement. Suppose a measurement described by the measurement operator M_m is performed on a quantum system in the state $|\psi\rangle$. The probability of getting an outcome m associated with M_m is then given by the Born rule as:

$$p(m) = \langle\psi| M_m^\dagger M_m |\psi\rangle \tag{1.26}$$

We describe the operator E_m associated to the outcome m to be an *element of the POVM* or *POVM element* in short. It is defined as:

$$E_m \equiv M_m^\dagger M_m \tag{1.27}$$

The POVM element E_m is a positive operator with outcome probabilities given as $p(m) = \langle\psi| E_m |\psi\rangle$. The full set of POVM elements are required to satisfy $\sum_m E_m = \mathbb{1}$ and are sufficient in determining the probabilities for the different measurement outcomes associated with the measurement. The entire set of operators $\{E_m\}$ is called the POVM.

In the instance where the measurement performed is described by projection operators P_m satisfying $P_m P_n = \delta_{mn} P_m$, and $\sum_m P_m = \mathbb{1}$, the POVM elements associated with this measurement satisfy:

$$E_m = P_m^\dagger P_m = P_m P_m = P_m \tag{1.28}$$

This equality between the measurement operator and POVM element is only true in this instance.

In the density operator formalism, for a measurement associated with Hermitian operator E_m, with eigenvalue/measurement outcome m, the Born rule becomes:

$$p(m) = \mathrm{tr}\left[\rho E_m\right] \tag{1.29}$$

1.2.6 Quantum dynamics

In closed quantum systems, states evolve linearly to new states in $\mathcal{H}_d$ through unitary operations U defined as

$$U^\dagger U = U U^\dagger = \mathbb{1} \tag{1.30}$$

Crucially unitary operators act to preserve the lengths of vectors in $\mathcal{H}_d$, acting to "rotate" states to new states. In the density operator representation, unitary operators act in conjugation as:

$$\rho \to \rho' = U \rho U^\dagger \qquad \text{where } \rho,\, \rho' \in \mathcal{L}(\mathcal{H}_d) \tag{1.31}$$

Many realistic quantum systems are not closed, and the evolution of quantum systems cannot be described purely in terms of unitary operators. The general evolution of a quantum system transforming the input state ρ to the final state ρ' can be described by the quantum operation $\mathcal{E}$, which is:

1. A linear mapping.
2. Satisfies $tr[\mathcal{E}(\rho)] \leq 1$.

3. A completely positive mapping. Where complete positivity is defined by

$$\mathcal{E} \otimes \mathbb{1}_n \geq 0 \quad \forall n \in Z_+ \tag{1.32}$$

Here the index n is a positive integer denoting the dimension of the additional particle. The final transformed state can be normalised and expressed as $\rho' = \frac{\mathcal{E}(\rho)}{tr[\mathcal{E}(\rho)]}$. It can be shown [9] that an operation $\mathcal{E}$ is a valid quantum operation if and only if it can be decomposed as:

$$\mathcal{E}(\rho) = \sum_i K_i \rho K_i^\dagger \tag{1.33}$$

with linear operators K_i, that satisfy the condition:

$$\sum_i K_i^\dagger K_i \leq \mathbb{1} \tag{1.34}$$

We term the operators K_i *Kraus operators* and describe the decomposition of the operation as a *Kraus decomposition.* A key point of this decomposition is that the Kraus operators for a given quantum operation are not unique - the operation can often be represented by many different sets of Kraus operations. Any two sets of Kraus operators for a quantum operation are related via a unitary operation [9, 10].

If the transformation on a quantum system is due to a measurement, we can relate the measurement operators to the Kraus operators describing the transformation. The POVM elements of the measurement are related as:

$$E_m = K_m^\dagger K_m \tag{1.35}$$

If the equality holds in eq. (1.34) then the collection of the Kraus operators $\left\{E_m = K_m^\dagger K_m\right\}$ form a valid POVM.

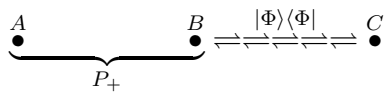
1.2.7 The Choi-Jamiołkowski Isomorphism

The Choi-Jamiołkowski isomorphism [13, 14] is an isomorphism from the set of trace preserving quantum operations $\mathcal{E}$ mapping from $\mathcal{H}_A$ to $\mathcal{H}_{A'}$ to bipartite density operators ρ in the tensor product Hilbert space $\mathcal{H}_A \otimes \mathcal{H}_{A'}$. The Hilbert spaces $\mathcal{H}_A$ and $\mathcal{H}_{A'}$ need not be of the same dimension, however will be taken to be so in this thesis. We will take $\mathcal{H}_A$, $\mathcal{H}_{A'}$ to be both of dimension d. We express the *Choi-Jamiołkowski state (CJ state)* $\rho(\mathcal{E})$ as:

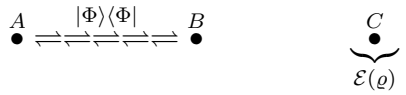
$$\rho(\mathcal{E}) := (\mathbb{1}_A \otimes \mathcal{E}_A)\left(|\Phi\rangle\langle\Phi|_{AA}\right) \qquad |\Phi\rangle = \frac{1}{\sqrt{d}} \sum_{i=1}^{d} |ii\rangle_{AA} \tag{1.36}$$

The state $|\Phi\rangle$ is the canonical maximally entangled state for the state space $\mathcal{H}_A \otimes \mathcal{H}_A$. For $\rho(\mathcal{E})$ to be a valid CJ state the reduced state $\mathrm{tr}_A[\rho(\mathcal{E})]$ must be a maximally mixed state.

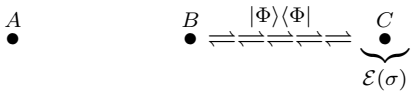
A useful tool to understand the CJ isomorphism is by comparing it to quantum teleportation; suppose we have three quantum systems A, B, C, where the systems B and C share the maximally entangled state $|\Phi\rangle\langle\Phi|$ and system A is in the state ϱ. Let us follow the transformation due to the action of the Bell state measurement POVM element $P_+ = |\Phi\rangle\langle\Phi|$ on systems A and B:



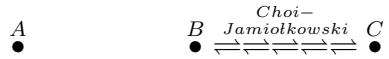
The Bell POVM element teleports the system A state ϱ to system C. Now we do a local trace preserving quantum operation $\mathcal{E}(\cdot)$ on C's particle taking the overall system state to $|\Phi\rangle\langle\Phi|_{AB} \otimes \mathcal{E}(\varrho_C)$:



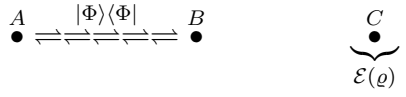
We can generate CJ state with a method similar to a slightly re-ordered teleportation procedure. Just as with "normal" teleportation we start with three systems A, B, C where B and C are in the maximally entangled $|\Phi\rangle\langle\Phi|_{BC}$ state, only this time we start by doing a $\mathcal{E}(\cdot)$ on the C state σ:



This takes our $B - C$ state to a CJ state:



The action of the POVM element P_+ on A and B now takes us to the final teleportation state under action of the operation $\mathcal{E}(\cdot)$:



A simple rearrangement of the teleportation protocol described above generates a CJ state, and provides us insights on how to use the CJ state to arrive at say how the operation $\mathcal{E}$ transforms an input state. We can use this to determine the action of the CP map for a given CJ state and input state.

Where $\mathcal{E}$ is a two particle operation acting on particles A and B mapping from the Hilbert spaces $\mathcal{H}_A \otimes \mathcal{H}_B$ to $\mathcal{H}_{A'} \otimes \mathcal{H}_{B'}$ the corresponding CJ state ρ is defined as:

$$\rho(\mathcal{E}_{AB}) := (\mathbb{1}_A \otimes \mathbb{1}_B \otimes \mathcal{E}_{AB})\,(|\Phi\rangle\langle\Phi|_{AA} \otimes |\Phi\rangle\langle\Phi|_{BB}) \tag{1.37}$$

2 Quantum computation and error correction

2.1 The circuit model of quantum computation

Quantum computation is a physically realisable model of computation based on quantum mechanics. While there are various ways of modelling quantum computation, the model used throughout this thesis pictures it in terms of *quantum circuits* - the quantum analogue to classical boolean logic circuits. Before outlining the main components of quantum circuits we will first introduce *qubits* in the next section.

2.1.1 Qubits and the Bloch sphere representation

The basic unit of information in most models of quantum computation is the qubit [15], the quantum analogue to classical bits. A single qubit is treated as a two-level quantum system, like say the spin orientation of an electron. A qubit in a pure quantum state can be specified by four real numbers, and represented by a unit-vector $|\phi\rangle$ in a two-dimensional Hilbert space $\mathcal{H}_2$ with an orthonormal basis denoted $\{|0\rangle, |1\rangle\}$ as:

$$|\phi\rangle = \alpha |0\rangle + \beta |1\rangle \qquad \text{where } \alpha, \beta \in \mathbb{C} \tag{2.1}$$

As in subsection (1.2.3) we will most often use density matrices to represent qubit states. The 2-dimensional identity matrix with the three *Pauli matrices*[1] (shown in eq. (2.2) below) span $\mathcal{L}(\mathcal{H}_2)$:

$$X := \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \qquad Y := \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix} \qquad Z := \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} \tag{2.2}$$

From the expansion of density matrices into trace and traceless parts as in eq. (1.13) it is possible to express the qubit density matrices ρ as:

$$\rho = \frac{1}{2} (\mathbb{1}_2 + \underline{r} \cdot \underline{\sigma}) \tag{2.3}$$

where the Bloch vector $\underline{r} = (r_x, r_y, r_z)^{\mathrm{T}} \in \mathbb{R}^3$ and $\underline{\sigma} = (X, Y, Z)^{\mathrm{T}}$. The set of all single qubit states - the *Bloch sphere* $\mathcal{Q}$ is defined as follows:

[1] These matrices are diagonalised in the Pauli Z eigenbasis $\{|0\rangle, |1\rangle\}$.

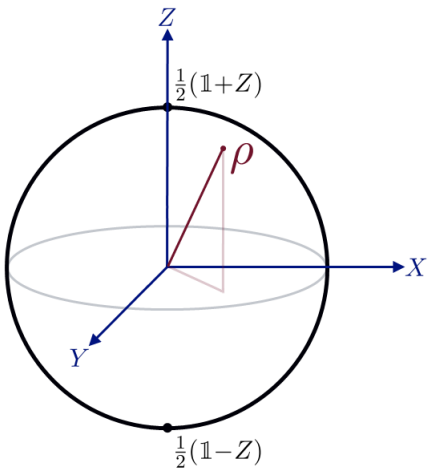


Figure 2.1: Representation of the Bloch sphere of qubit states. An arbitrary qubit state ρ represents a point in the sphere, with surface points (as in the figure) corresponding to pure states and internal sphere point corresponding to mixed states. The north and south "poles" of the sphere correspond to the basis states $|0\rangle$ and $|1\rangle$ and are the Pauli Z eigenstates.

Definition 2.1.1. ***The Bloch sphere*** $\mathcal{Q}$

The single qubit state space $\mathcal{Q}$ *is the closed convex set of density operators* ρ *on* $\mathcal{H}_2$ *with Bloch vectors* $\underline{r}$ *satisfying* $0 \leq |\underline{r}| \leq 1$.

Figure (2.1) above shows a geometric representation of the Bloch sphere.

2.1.2 Quantum circuits and universality

Computation on quantum circuits generally involves three stages; preparation of an input state in a *quantum register*, performing a sequence of logic gate operations, and finally measuring the output of the gates to obtain some bit-string as a result.

The quantum register serves as a "working memory" for the quantum device storing a finite set of qubits that are acted upon by the logic gates forming the quantum circuit. The Hilbert space of a quantum register storing N qubits corresponds to a tensor product of each single qubit Hilbert space as $\mathcal{H}_2 \otimes \mathcal{H}_2 \otimes \ldots \otimes \mathcal{H}_2 = \mathcal{H}_2^{\otimes N}$. If $|0\rangle$ and $|1\rangle$ denote the basis states for some qubit in the register, then the *computational basis* of the circuit is the set of register states $\{|\omega\rangle\}$ where ω is some N-bit string. The single qubit basis states $|0\rangle$ and $|1\rangle$ will be taken here to be the Pauli Z eigenstates.

Quantum operations acting as gate operations for the circuit are restricted to unitary operations using Stinespring's dilation theorem [16] - as any arbitrary quantum operation $\mathcal{E} : \rho \to \mathcal{E}(\rho)$ can be enacted by performing a unitary operation on a larger system composed of the quantum register coupled with some ancillary qubits, followed by projective measurements on just the ancillary qubits [17].

Gate operations in the quantum circuit correspond to unitary transformations of one or more of the qubits in the register, and are reversible. It is worth noting that gate operations may often be applied stochastically, as in the case of *adaptive computation* where measurement operations are performed after certain gate operations with the subsequent gates chosen based on those measurement results.

Measurements on individual qubits in the quantum circuit are always projective measurements in the Pauli-Z eigenbasis $\{|0\rangle, |1\rangle\}$ resulting in the output of a single classical bit. Single qubit measurements on different qubits commute. Multi-qubit measurements (like a Bell state measurement) can be performed by first performing a multi-qubit unitary, followed be a succession of single qubit measurements.

In analogy with classical circuits, quantum circuits can be pictured in terms of quantum versions of wires and gates with quantum wires serving to carry single qubit state information from one gate operation to the next in a temporal sense.

Any arbitrary N qubit quantum operation can often be approximated to a desired accuracy by performing sequences of operations from some smaller, restricted elementary set called a *universal set of quantum gates* [18]. An example of just such a set is the *standard universal set* [19] containing the single qubit *Hadamard* H, $\frac{\pi}{8}$-gate $P_{\frac{\pi}{8}}$, and the two qubit entangling control-not gate CX:

$$H = \frac{1}{\sqrt{2}}(X + Z) \qquad P_{\frac{\pi}{8}} = e^{i\frac{\pi}{8}} e^{-i\frac{\pi}{8}Z} \qquad CX = \begin{pmatrix} \mathbb{1} & \underline{0} \\ \underline{0} & X \end{pmatrix} \tag{2.4}$$

2.2 Quantum error correction

Physical implementations of quantum computers are difficult due to the fragility of quantum states and the difficulty in manipulating them. Any undesired change to the density matrix representing a quantum system is termed *quantum noise*. Given the non-local nature of quantum computation, even single qubit errors can quickly spread throughout the computation accumulating exponentially. Noise has the effect of destroying the "quantum-ness" of quantum states, diminishing the believed computational speed-up of quantum computers over their classical counterparts.

Errors in the computation can occur not only from non-ideal gate implementations, but also from interactions of the quantum system with the environment it is contained in. As the coupled quantum system and environment evolve unitarily, potentially non-unitary evolution takes place on the quantum system. Quantum errors, unlike digital classical ones are continuous and must undergo some digitisation process to give us the hope of correcting them.

Given the importance of trying to build quantum devices resistant to the effects of quantum noise considerable work has been done to develop methods for protecting quantum computational states against the effects of noise.

Within this section we will introduce some simple quantum noise models and outline quantum error correction, before discussing the stabiliser formalism, and concluding with fault tolerant quantum computing and the threshold theorem.

2.2.1 Some quantum noise models

In this section we will briefly introduce three noise models that will feature later in this thesis. All three noise models are trace-preserving, completely positive maps and so have a Kraus decomposition as in section (1.2.6).

Local depolarising

The single qubit depolarising channel takes input states ρ to the following output states:

$$\rho \to \rho' = (1-\varepsilon)\rho + \varepsilon\frac{\mathbb{1}}{2} \tag{2.5}$$

where ε represents some probability. We can rewrite eq. (2.5) as:

$$\rho' = \frac{1}{2}(\mathbb{1} + \underbrace{(1-\varepsilon)\underline{r}}_{\underline{r}'} \cdot \underline{\sigma}) \tag{2.6}$$

As we can see, the effect of adding the depolarising noise is to transform the Bloch vector $\underline{r}$ to $\underline{r}'$. The map takes input states from the Bloch sphere into a smaller sphere contained within the Bloch sphere.

Suppose we now have quantum noise affecting a two qubit input state, we can imagine depolarising noise effecting each of the individual qubits as being such a two qubit noise. We term a noise model like this as *local depolarising noise,* where each of the qubit forming the pair independently undergoes depolarising noise at a rate of ε. For this noise model we have no change to the input state at a rate of $(1-\varepsilon)^2$, one qubit depolarised but not the other at a rate of $\varepsilon(1-\varepsilon)$, or both depolarised at a rate ε^2. We may express two qubit local depolarising noise as a map taking the input state ρ_{AB} (where A and B denote sub-systems) to the following output state:

$$\rho'_{AB} = (1-\varepsilon)^2\rho_{AB} + \varepsilon(1-\varepsilon)\rho_A \otimes \frac{\mathbb{1}}{2} + \varepsilon(1-\varepsilon)\frac{\mathbb{1}}{2} \otimes \rho_B + \varepsilon^2\frac{\mathbb{1}}{4} \tag{2.7}$$

Joint depolarising

An alternative way of thinking about two qubit depolarising noise is to consider such noise acting on both qubits together rather than on the sub-qubits individually. We call this *joint depolarising noise,* its action being to take an input state ρ_{AB} to the following state (where again $\varepsilon$is some probability):

$$\rho'_{AB} = (1-\varepsilon)\rho_{AB} + \varepsilon\frac{\mathbb{1}}{4} \tag{2.8}$$

From section (1.2.3) we see that the output state ρ'_{AB} can be written in terms of a tensor Pauli-basis as:

$$\rho'_{AB} = \frac{1}{4}(\mathbb{1} + \underbrace{(1-\varepsilon)\underline{r}_{AB}}_{\underline{r}'_{AB}} \cdot \underline{\sigma}_{AB}) \tag{2.9}$$

where $\underline{r}_{AB}$ is a 15 element real vector, and $\underline{\sigma}_{AB}$ is a vector formed from the tensor product of each of the three Pauli matrices. As with eq. (2.6), we see that joint depolarising noise has the effect scaling non-identity Pauli components by a factor of $(1-\varepsilon)$.

Local dephasing

The final natural noise model we will look at is *local dephasing noise.* Single qubit dephasing noise acts with probability ε to perform a phase-flip (Pauli Z) on an input state ρ. We may express such a transformation as:

$$\rho \to \rho' = (1-\varepsilon)\rho + \varepsilon Z\rho Z \tag{2.10}$$

The dephasing noise channel takes the Bloch vector of the input state to the transformed Bloch vector $\underline{r}' = ((1-2\varepsilon)r_x, (1-2\varepsilon)r_y, r_z)^T$ i.e. the dephasing acts to take input states from the Bloch sphere to a subset represented by a sphere contracted along the Z axis - like a Z axis cigar. As with the local depolarising noise case, for a two qubit input state ρ_{AB} we can imagine dephasing noise acting independently on each qubit at rate ε. Hence the noise has no effect on the input state at a rate of $(1-\varepsilon)^2$, one qubit dephased but not the other at a rate of $\varepsilon(1-\varepsilon)$, or both dephased at a rate ε^2:

$$\begin{aligned}\rho'_{AB} = (1-\varepsilon)^2\rho_{AB} + \varepsilon(1-\varepsilon)(Z\otimes\mathbb{1})\rho_{AB}(Z\otimes\mathbb{1}) \\ +\varepsilon(1-\varepsilon)(\mathbb{1}\otimes Z)\rho_{AB}(\mathbb{1}\otimes Z) + \varepsilon^2(Z\otimes Z)\rho_{AB}(Z\otimes Z)\end{aligned} \tag{2.11}$$

Where $\varepsilon = \frac{1}{2}$ we say that the output state is *completely dephased,* corresponding to states that are mixtures of the Pauli Z eigenstates.

2.2.2 Overview of error correction

Classical digital computation devices need not concern themselves with error correction as the failure rate of most of the components composing them is on the order of one error in 10^{17} operations [1]. Communication devices like modems and optical disc readers on the other-hand use error-correcting codes to protect the information from the effects of noise.

Just as with classical error correcting codes, the idea behind quantum error correction is that in order to protect information against the effects of noise redundant information must be added. By *encoding* our logical states the information contained in them is spread over many more physical qubits, and may be repeatedly checked for corruption by the effects of noise. If any information is corrupted, there should in principle be enough redundancy that the original information can be retrieved by performing appropriate recovery and decoding operations on the encoded message.

Adding redundancy to quantum information is less straightforward than in classical coding due to the *No Cloning Theorem* [1]; an unknown quantum state cannot be duplicated. Additionally, as measurements of quantum states can destroy quantum superpositions damaging our logical states, quantum error correction methods must detect and correct quantum errors without modifying the quantum state in question and gaining any information about it.

To protect the quantum information on k qubits we can *encode* them in n qubits, that is we will take the 2^k k-qubit logical basis states and map them to 2^n n-qubit code basis states by first adding a suitable number of ancilla qubits and then performing an appropriate unitary encoding operation. Likewise by performing the inverse of the encoding unitary and discarding the $(n-k)$ ancilla states we can return to our original computational states.

Any computational state formed of a linear combination of computational basis states gets mapped to a code state formed from the linear combination of appropriate code basis states. The space of all code states C hence forms a Hilbert space - a subspace of the full n qubit tensor product Hilbert space.

Assuming that gate operations are ideal, and that quantum noise affects states in between gate operations, the cumulative errors affecting the system over some fixed period of time can be represented as a quantum channel $\mathcal{E}$ acting on a density matrix representing a coded state $\rho_{1\ldots n}$ as:

$$\rho_{1\ldots n} \to \tilde{\rho}_{1\ldots n} = \mathcal{E}(\rho_{1\ldots n}) = \sum_a E_a \rho_{1\ldots n} E_a^\dagger \qquad \text{where } \sum_a E_a^\dagger E_a = \mathbb{1} \tag{2.12}$$

The set of Kraus elements $\{E_a\}$ form a set of errors acting on the system, for example, if errors were affecting a single qubit then $\{E_a\}$ could correspond to bit-flip and phase-flip errors.

If a quantum error is correctable for some quantum code then there exists a recovery operation - a trace-preserving quantum channel $\mathcal{R}$ acting on an erroneous code state $\tilde{\rho}_{1\ldots n}$ as:

$$\tilde{\rho}_{1\ldots n} \to \mathcal{R}(\tilde{\rho}_{1\ldots n}) = \sum_k R_k \tilde{\rho}_{1\ldots n} R_k^\dagger \qquad \text{where } \sum_k R_k^\dagger R_k = \mathbb{1} \tag{2.13}$$

If $\mathcal{R}$ is a recovery operation for the set of errors $\{E_a\}$ then for the coded state $\rho_{1\ldots n}$:

$$(\mathcal{R} \circ \mathcal{E})(\rho_{1\ldots n}) = \rho_{1\ldots n} \tag{2.14}$$

It is worth noting that not every $\mathcal{E}$ and code C will have an associated $\mathcal{R}$. There may also be situations where a noisy channel $\mathcal{E}$ may have a recovery operation $\mathcal{R}$, but a very close channel $\mathcal{E}'$ may not. In such cases $\mathcal{R}$ may still be able attempt to correct $\mathcal{E}'$ albeit with some (high) probability of success.

A specific quantum code is typically only suitable for correcting a certain basis of errors $\{E_a\}$. We will now look at the relationship between a given basis of errors and the basis code states capable of correcting them.

In order to be able to correct two different quantum errors E_a and E_b we must have the ability to distinguish the action of one error E_a on a basis code state $|\psi_i\rangle$, from the action of the other error E_b on a different basis code state $|\psi_j\rangle$. For perfect distinguishability the two different erroneous code states $E_a|\psi_i\rangle$, $E_b|\psi_j\rangle$ must be orthogonal:

$$\langle\psi_i| E_a^\dagger E_b |\psi_j\rangle = 0 \qquad \text{where } i \neq j \tag{2.15}$$

The restrictions from eq. (2.15) alone are not sufficient to specify a code useful for quantum computation. In order to learn what error has occurred (or if an error has even occurred) we must perform measurements on the quantum system. These measurements however must not reveal any information about the code state prior to being "affected" by noise; doing so would damage the superpositions within the quantum state we wish to protect. By measuring all inner products $\langle\psi_i| E_a^\dagger E_b |\psi_i\rangle$ for all combinations of E_a and E_b (including where $E_a = \mathbb{1}$, i.e. no error) we may learn exactly what error has effected our system. In order for us to learn nothing about the uncorrupted code state these measurements must all be equal for all basis code states:

$$\langle\psi_i| E_a^\dagger E_b |\psi_i\rangle = \langle\psi_j| E_a^\dagger E_b |\psi_j\rangle \tag{2.16}$$

The literature [20, 21, 22] provides the full quantum error correction code restriction as:

$$\langle\psi_i| E_a^\dagger E_b |\psi_j\rangle = \alpha_{ab}\delta_{ij} \tag{2.17}$$

where $|\psi_i\rangle$, and $|\psi_j\rangle$ run though all basis code states, E_a and E_b over all basis errors (including the identity) and α_{ab} is some complex number, independent of i and j, that forms an element of a positive matrix α. As α_{ab} is independent of i and j, we see that eq. (2.16) is satisfied for all $|\psi_i\rangle$, and $|\psi_j\rangle$. The proof that a quantum code satisfying eq. (2.17) for a given set of errors $\{E_a\}$ is both necessary and sufficient for the existence of $\mathcal{R}$ capable of correcting $\{E_a\}$ can be found in [23].

Crucially, the ability to correct errors from the finite set $\{E_a\}$ means that we can then correct for any error with Kraus error elements formed from the linear combination of $\{E_a\}$. Any error channel $\mathcal{E}$ acting only on a single qubit has Kraus error elements E_a that can be expressed as a linear combination of the identity matrix $\mathbb{1}$ and the Pauli X, Y, Z matrices. We therefore only need $\mathcal{R}$ to be able to detect and reverse the action of the 3n individual qubit Pauli operators acting under conjugation to correct for any single qubit error. Naturally this scheme extends to systems of higher dimension, however it becomes progressively more complex with error syndrome measurements.

An example of a very basic error correcting code can be found in section (9.1) in the appendix.

2.2.3 Introduction to the stabiliser formalism

One of the most important results to come out of quantum error correction is the *stabiliser formalism* [3]. The formalism considers n-qubit systems where the set of states and operations is constrained. This restricted set - the stabiliser states can be efficiently specified in terms of a number of parameters that is polynomial in n. Despite having this compact description such states can still exhibit multiparticle entanglement, providing us valuable insights into the relationship between classical and quantum computation.

The stabiliser formalism also provides us a way of compactly describing an important class of quantum error correcting codes called *stabiliser codes*. Within this section we will introduce the stabiliser formalism, outlining stabiliser codes, before leading into concatenated coding for its relevance to fault-tolerant quantum computation. The ideas introduced from the stabiliser formalism will be important when discussing the Gottesman-Knill theorem and magic state architectures later in the chapter.

2.2.3.1 The Pauli group and stabiliser states

An operator S_i is said to *stabilise* a state $|\psi\rangle$ if:

$$S_i |\psi\rangle = |\psi\rangle \tag{2.18}$$

In other words, the state $|\psi\rangle$ is an eigenstate of S_i with eigenvalue of $+1$. By specifying a set of operators $\mathcal{S}$ and looking at their mutually stabilised states we can specify an associated subspace V_S of the Hilbert space. A particular state in V_S may be uniquely stabilised and therefore specified by a large enough set of operators in $\mathcal{S}$ (up to some global phase). In such cases it is often more efficient to work with a set of stabilising operators rather than a state.

The stabiliser states presented in the literature crucial to fault tolerant quantum computing are those specified from considering operators from the Pauli Group.

The single qubit Pauli group $\mathcal{G}_1$ is the following,

$$\mathcal{G}_1 := \{\pm\mathbb{1}, \pm X, \pm Y, \pm Z, \pm i\mathbb{1}, \pm iX, \pm iY, \pm iZ\} \tag{2.19}$$

and the Pauli group $\mathcal{G}_n$ on n qubits is:

$$\mathcal{G}_n := \mathcal{G}_1^{\otimes n} \tag{2.20}$$

The weight of an element M of $\mathcal{G}_n$ is defined as the number of qubits on which M acts as a non-identity operator. The 2^n dimensional matrix representation of the Pauli group $\mathcal{G}_n$ has the following properties:

1. For each $M \in \mathcal{G}_n$, $M^2 = \pm\mathbb{1}^{\otimes n}$.
2. Elements $M \in \mathcal{G}_n$ satisfying $M^2 = \mathbb{1}^{\otimes n}$ are hermitian, and those satisfying $M^2 = -\mathbb{1}^{\otimes n}$ are anti-hermitian.
3. Any pair of elements $M, N \in \mathcal{G}_n$, commute ($MN = NM$) or anti-commute ($MN = -NM$).

It is possible to specify every operator in a Pauli group by considering group products of a subset of operators. In general, a subset of elements $\{T_1, T_2, ..., T_r\}$, $r \leq n$, of a group $\mathcal{T}$ is said to *generate* $\mathcal{T}$ if every element of $\mathcal{T}$ can be written as a group product of elements from the subset and their inverses. Each element of $\{T_1, T_2, ..., T_r\}$ is called a *generator* and $\mathcal{T}$ may be denoted as $\langle T_1, T_2, ..., T_r\rangle$. A group of cardinality $|\mathcal{T}|$ has at most $log_2(|\mathcal{T}|)$ generators [1, 3].

The literature [1, 3] defines stabiliser groups $\mathcal{S}$ as:

Definition 2.2.1. ***The n qubit stabiliser group.***

Suppose we have a system of n qubits. A stabiliser group $\mathcal{S}$ is an abelian subgroup of $\mathcal{G}_n$ that does not contain the elements $-\mathbb{1}^{\otimes n}$ and $\pm i\mathbb{1}^{\otimes n}$.

Any vector stabilised by the generators of a stabiliser group would also be stabilised by any products of generators. A stabiliser group $\mathcal{S}$ can be specified in terms of its smallest generating subgroup.

Given a stabiliser group $\mathcal{S}$, its associated stabiliser subspace $V_{\mathcal{S}}$ is:

Definition 2.2.2. ***The n qubit stabiliser subspace.***

Suppose we have a stabiliser group $\mathcal{S} = \langle S_1, S_2, ..., S_r \rangle$ acting on n qubits. We will define the stabiliser subspace $V_{\mathcal{S}}$ associated to $\mathcal{S}$ as the set of states $|\psi\rangle \in \mathcal{H}^{\otimes n}$ that satisfy

$$S_i |\psi\rangle = |\psi\rangle \,, \ \forall \, S_i \in \mathcal{S} \tag{2.21}$$

If $\mathcal{S}$ has minimally r generators then $V_{\mathcal{S}}$ has dimension 2^{n-r}.

Stabiliser subspaces are useful in that we only need to specify a small set of stabiliser group generators to represent the set of states contained in the subspace. A straightforward example of a stabiliser group and its associated stabiliser subspace is that for three qubits found in section (9.2) in the appendix.

2.2.3.2 Unitary evolution in the stabiliser formalism

Having looked at stabiliser subspaces we will now look at their evolution under unitary operations. Suppose we apply a unitary operation U to a state within the stabiliser subspace $V_{\mathcal{S}}$, associated with the stabiliser set $\mathcal{S}$. Let $|\psi\rangle \in V_S$, then for any element S_i of $\mathcal{S}$ we have:

$$U |\psi\rangle = U S_i |\psi\rangle = \left(U S_i U^\dagger\right) U |\psi\rangle \tag{2.22}$$

The transformed state $U |\psi\rangle$ is therefore stabilised by $US_iU^\dagger$, that is it now stabilised by the group $\mathcal{S}' = \left\langle US_1U^\dagger, US_2U^\dagger, ..., US_rU^\dagger \right\rangle$, with an associated "transformed" stabiliser subspace $V_{\mathcal{S}'} = \{U |\psi\rangle \,|\, |\psi\rangle \in V_S\}$. We see that the evolution of the stabiliser states can be computed purely in terms of the evolution of the stabiliser generators of $\mathcal{S}$.

An important set of operators associated to the Pauli group is its *normaliser* group. These operators acting under conjugation, map Pauli group elements to Pauli group elements. In particular the normaliser to $\mathcal{G}_n$ can be generated by the n qubit Clifford group:

$$\mathcal{C}_n := \left\langle H_i, \, P_{\frac{\pi}{2}\,i}, \, CX_{ij} \right\rangle \tag{2.23}$$

where the indices i and j denote the qubits the operators act on, with identities on all others. Readers should recall from subsection (2.1.2), the Hadamard H and control-not CX gates, with the phase gate $P_{\frac{\pi}{2}}$ is given as:

$$P_{\frac{\pi}{2}} = e^{i\frac{\pi}{2}} e^{-i\frac{\pi}{2}Z} \tag{2.24}$$

As Clifford operators are in the normaliser to $\mathcal{G}_n$, the action of a Clifford operator on a stabiliser state requires us to simply update the generators of the associated stabiliser group, needing at most n operations to perform.

We conclude this section with the following theorem from the literature [3]:

Theorem 2.2.3. ***Normaliser from Clifford generators.***

Suppose U is in the normaliser to $\mathcal{G}_n$, then U may be composed from the product of $O(n^2)$ Clifford group generators (up to some global phase).

We have that computing any n-qubit quantum circuit preparing stabiliser states and involving only operations from the associated Clifford group can be done polynomially in n. Measurement in the stabiliser formalism is discussed in section (9.3) in the appendix.

2.2.3.3 The Gottesman-Knill theorem

An important result of the stabiliser formalism relevant to this thesis is the Gottesman-Knill theorem [1, 24]. Let us suppose we have an n -qubit quantum computer capable of adaptive computation only involving operations from the following list:

1. The ability to prepare qubits in the Pauli Z eigenstate $|0\rangle$. (2.25)
2. Can probabilistically apply any unitary operations from the Clifford group $\mathcal{C}_n$.
3. Have the ability to measure an eigenvalue of a Pauli operator on any qubit.

By adaptive computation we mean that the description of an operator in the computation to be performed at step t, may be dependent on all of the measurement outcomes in steps $1, \ldots, t-1$. The Gottesman-Knill theorem states that quantum computation restricted to only the above operations may be efficiently classically simulated on a classical computer.

We can see this result as follows: In the previous subsections we observed that a stabiliser state can be specified compactly with at most n-stabiliser generators. As we are restricted to only Clifford operators only, each Clifford gate operation requires us to simply update the set of generators. As mentioned in section (9.3), Pauli measurements of stabiliser states can also be performed efficiently.

Interestingly, stabiliser circuits based on the operations listed in eq. (2.25) can generate bipartite maximally entangled states and multi-partite entangled states despite being classically simulatable [25]. It is important to note that stabiliser circuits alone cannot perform universal quantum computation, unless the operations in eq. (2.25) additionally include the ability to perform a single qubit unitary that cannot be generated by Clifford operations [26]. The ability to take Clifford operations to a full universal set is explored further with magic state architectures in section (2.3).

2.2.3.4 Stabiliser codes and error correction

Given a stabiliser group $\mathcal{S}$, we can define an associated stabiliser subspace $V_{\mathcal{S}}$ that can be used as our coding subspace for some quantum error correcting code - such codes are

called *stabiliser codes*. Specifically, for a stabiliser $\mathcal{S}$ generated by r elements we can use its subspace $V_{\mathcal{S}}$ to encode $k = n - r$ logical qubits onto n code qubits. The normaliser is important in the context of stabiliser codes as the elements of the normaliser of $\mathcal{S}$, not contained in $\mathcal{S}$, act on encoded states as Pauli operators would on logical states.

In example (9.2) in the appendix it is straightforward to show that set of errors that satisfy the quantum error correction conditions in eq. (2.17) for the specified stabiliser subspace are the bit-flip errors from section (9.1) in the appendix.

2.2.4 Introduction to fault tolerant quantum computing

In our discussion so far we have only looked at instances where qubit dynamics, preparation, and measurements can be performed perfectly, with errors only occurring during "rest" intervals between operations. Physical implementations of quantum computers must also account for errors that can occur during such dynamical processes, including during error correction recovery procedures.

Fault tolerant quantum computing (FTQC) [27, 28, 29] shows that arbitrarily accurate quantum computation can be achieved despite the presence of faulty circuit elements provided that the quantum noise in the system is below some *noise threshold* [30, 28]. Within this section we will introduce the basic ideas in fault tolerant quantum computing, and show how we can arrive at the noise threshold for some restricted setting.

The essential idea in FTQC is that a computational circuit can be designed to use faulty physical components in a particular way, and then use error correction to suppress errors multiplying and propagating throughout the subsequent computation. Quantum computation circuits in particular are susceptible to multiplying errors through the use of multiparticle gates. For example, an error on a control qubit of a CX gate may be propagated to the target qubit too, therefore the single qubit error can become a two qubit error.

Computation can be made fault tolerant by performing encoded versions of operations on encoded qubits rather than decoding the qubits prior to performing the operation. That is, each element of the intended computation is replaced with a fault tolerant circuit simulating the action of the intended computation and suppressing error accumulation.

We will say that a computational circuit element is fault tolerant if, for $\leq t$ component failures, at most t errors are in the output of each logical qubit block outputted from the circuit element.

This definition is reliant on the error-correcting code used in the computation. For example, if our code could protect against two qubit errors then the definition is taken to require only two qubit errors on each logical qubit outputted. In order to be able

to perform fault tolerant computation we must be able to perform fault-tolerant error correction, encoded logical gate operations, measurement, and preparation of encoded states.

The key result from fault tolerant quantum computing is the threshold theorem introduced in the next subsection.

2.2.5 The fault tolerance threshold

Theorem 2.2.4. ***Threshold theorem.***

Suppose that the error rate of a gate-component forming a quantum circuit is some probability p. There exists some constant $p_{th} > 0$, such that if $p < p_{th}$ then the quantum circuit can be implemented by said faulty components to an accuracy ϵ, with a polylogarithmic overhead, and arbitrarily long quantum computation is possible.

Here we have made assumptions on the nature of the components forming the logical gates (i.e.. the gate architecture used) and on how they fail, namely we have considered a particular noise model effecting them. The threshold value for a given quantum system is dependent on a variety of factors including the gate architecture used, the error correction code used and the component level error corrected.

An error correction code like that described in section (9.1) in the appendix can protect against single qubit errors, but we will have failures where two or more errors occur simultaneously. Well designed encoded gates should output encoded blocks of qubits that if decoded would only contain errors that scale as $\mathcal{O}(p^2)$ [1].

An important concept in fault tolerant quantum computation is the idea of *concatenated codes.* It is possible to reduce the effective error rate in the quantum computation further, by recursively applying the fault tolerant scheme in constructing a hierarchy of quantum circuits. For example if a single qubit is encoded on n qubits, then these n qubits may be encoded again on n^2 qubits. If the effective error rate from one level of coding is p, then by encoding it again the effective error rate can be reduced to cp^2. This repeat encoding can be done ad-infinitum.

From the proof of the threshold theorem in the literature [1] we see that if the individual gate noise is less than p_{th}, then for fault tolerant computation of length T, we need on the order of $log_2(log_2(T))$ levels of concatenation acting on a circuit size of the order of $polylog(T)$.

2.3 Magic state based gate architectures

Inspired by results from fault-tolerant quantum computing that showed that operators from the Clifford group can be implemented fault-tolerantly[2], Bravyi and Kitaev presented in [2] a universal quantum computation gate architecture based on operations from the Clifford group. More specifically their architecture consists of operations from the following list:

1. The ability to perform stabiliser operations from list in eq. (2.25).
2. Can prepare ancillary qubits in a mixed state ρ that lies outside of the convex hull of Pauli eigenstates.

The idea behind these choices of operations is that while we can perform the operations from eq. (2.25) fault tolerantly using stabiliser codes, the ancilla state ρ is taken to be mixed from imperfections in the preparation procedure, i.e. from environmental noise, etc. When preparing n mixed ancilla qubits it is required that all of the ancillary qubits are independent from each other i.e. we can obtain the state $\rho^{\otimes n}$.

Bravyi and Kitaev in [2] looked to find the ancilla states ρ which in addition to the operations in eq. (2.25) could be used to efficiently simulate universal quantum computation.

As we know from section (2.2.3.3), we may, with suitable weighted dice rolls prepare any state within a "Bloch octahedron" using the set of operations in eq. (2.25). When ρ is within the Bloch octahedron our model is also classically efficiently simulatable, and hence accepted to be incapable of UQC. As it is conjectured that a quantum computer capable of universal computation cannot be efficiently classically simulated, it was argued that ρ must lie outside the octahedron in order to perform quantum computing. While not proving that universal quantum computation is simulatable for all states ρ outside the Bloch octahedrons, Bravyi and Kitaev showed that they could achieve universality for a large set of states using two schemes based on *magic state distillation*. The magic states are defined as:

Definition 2.3.1. ***The T-type and H-type magic states***

Consider the following two pure quantum states $|T\rangle$, $|H\rangle$:

$$|T\rangle\langle T| = \frac{1}{2}\left\{\mathbb{1} + \frac{1}{\sqrt{3}}(X+Y+Z)\right\} \qquad |H\rangle\langle H| = \frac{1}{2}\left\{\mathbb{1} + \frac{1}{\sqrt{2}}(X+Z)\right\} \tag{2.26}$$

The magic states are defined as the set of eight T-type quantum pure states $\{U|T\rangle, U \in \mathcal{C}_1\}$, and the twelve H-type quantum pure states $\{U|H\rangle, U \in \mathcal{C}_1\}$, where $\mathcal{C}_1$ is the single qubit Clifford group.

[2] Using concatenated methods described in section (2.2.4).

The magic states have two roles in the model simulating a universal quantum computer. The first is that it is possible to obtain a magic state from multiple copies of the mixed state ρ by using operations from eq. (2.25) provided that the *maximum fidelity*[3] between ρ and all the magic states are above certain thresholds. Secondly that the set of operations in eq. (2.25) together with the ability to supply ideal magic states allows us to implement non-Clifford single qubit unitaries V through a process of (potentially repeated) magic state preparation, and Clifford measurements/gate operations. The set of Clifford operators together with V form a universal set of gates. The exact technical details of how operators V are obtained from a supply of ρ are beyond the scope of this thesis and can be found in [2].

2.4 Summary of Chapter (2)

In this chapter we defined the Bloch sphere single particle state space $\mathcal{Q}$ and gave an overview of quantum circuits before introducing the idea of a universal set of quantum gates. We then gave an introduction to quantum error correction, the stabiliser formalism and error correction within the stabiliser formalism, leading to the Gottesman-Knill theorem. The next section gave a brief overview of fault tolerant quantum computing and introduced the threshold theorem. The chapter concluded with an introduction to magic state architecture based quantum computers for their relevance to the research presented later in the thesis.

[3]The *maximum fidelity* of ρ and a T, or H-type magic state labelled $|m\rangle$ is defined as: $F_m(\rho) \equiv \max_{U \in \mathcal{C}_1} \sqrt{\langle m| U^\dagger \rho U |m\rangle}$.

3 Classical simulation of noisy quantum computers

3.1 Bounds on the noise threshold

The threshold theorem provides us a goal to work towards for a physical realisation of a quantum computer. Wide efforts have therefore been made to determine values of the threshold p_{th} for a variety of different quantum computing architectures, and noise models. Calculating precise values of the threshold has proven to be extremely difficult, and so in many cases only estimates for the threshold value have been attained. The focus has therefore been on tightening the ranges of these noise values.

We can imagine the computational capabilities of noisy quantum computing to fall within three regions based on the error rate ρ_{noi} in the computing device; A lower bound region of noise rates satisfying $p_{noi} \leq p_{low}$, an upper bound region $p_{upp} \leq p_{noi}$, and finally an intermediate region $p_{low} < p_{noi} < p_{upp}$.

In the following sections we will provide the interpretations of these noise regions and state current best estimates of the threshold bound values.

3.2 Lower bound results

The lower noise threshold value p_{low} indicates the maximum error rate at which a particular fault tolerant error correction scheme can be used to allow a noisy quantum computer to efficiently simulate an ideal universal quantum computer.

Values for p_{low} have been calculated using three methodologies, each giving significantly different values from the others:

Numerical simulation methods often give the highest valued lower thresholds. They typically work by randomly applying simple errors to simulations of noisy quantum computers with error correction, and observing how resilient the simulation is to these errors.

The second method looks to rigorously prove that a specific quantum circuit, designed under some error correction scheme, is fault tolerant for a given error rate. This method

gives the lowest, most conservative lower bound threshold values, magnitudes lower than those obtained from numerical methods.

The final method is to estimate threshold values analytically under certain assumptions that are speculated to have negligible effects. This method yields intermediate values for the threshold between the rigorous calculations and numerical simulation values.

As mentioned in the previous chapter, computed threshold values highly depend on the chosen error correcting code, available fault-tolerant circuitry, and method of analysis. Current lower threshold values are as follows:

For concatenated 7-qubit codes, various numerical estimates place the threshold value on the magnitude of 10^{-3} [31]. Lower bounds for such codes have been rigorously proved to be 2.73×10^{-5} [32, 33]. Better values for the lower threshold bound have been rigorously proved for 25-qubit Bacon-Shor code at 1.9×10^{-4} [34].

An alternative approach to simply considering error correcting codes is to prepare complex ancilla states in lieu of gates on the data qubits. Erroneous ancilla states can easily be discarded, but achieving low error rates throughout the computation can be prohibitively expensive in terms of resources [35, 36]. Numerical simulations calculate the lower threshold for such systems at around 10^{-2}, with a rigorous analytical proof placing the value to be around 10^{-3} [37, 38].

Advances have been made using *surface codes* which operate as stabiliser codes constructed from a two dimensional lattice of physical qubits acting on nearest neighbours (or other geometries of physical qubits). Such topological quantum computers are inherently tolerant to the formation of local errors [39]. Error correction procedures based on surface codes have achieved lower threshold values of approximately 10^{-2} [40], with a better value of 1.85×10^{-1} obtained when the error model considered is restricted to depolarising noise [41] (see section (2.2.1)).

The following table summarises the lower threshold values for approaches highlighted:

Approach	**Method**	**Threshold value**
Concatenated 7-qubit codes	Numerical	10^{-3}
Concatenated 7-qubit codes	Analytical	2.73×10^{-5}
25-qubit Bacon-Shor code	Analytical	1.9×10^{-4}
Ancilla state preparation	Numerical	10^{-2}
Ancilla state preparation	Analytical	10^{-3}
Surface codes	Numerical	10^{-2}
Surface codes (Depolarising noise)	Numerical	1.85×10^{-1}

(3.1)

3.3 Upper bound results

Whilst the majority of effort has been put towards calculating better lower bounds, steps have also been taken towards calculating upper bound values. The central aim of the work presented in this thesis is to find new better upper bounds to the threshold.

The upper noise threshold value p_{upp} indicates the largest possible noise value at which we believe we can get worthwhile computation from a noisy quantum computer. Note that by worthwhile we do not mean full universal quantum computation but rather something lesser that is still better than classical computation.

The methodology for determining upper bound values, most relevant to this thesis looks at when quantum circuits built around specific gate architectures and affected only by certain noise models can be efficiently simulated by some classical algorithm. This is based on the view that quantum computers are capable of computational tasks beyond classical computers. If quantum computation is simulatable on a classical computer then there is no need to construct a quantum computer in the first place.

We will now outline a couple of approaches that show how various restricted quantum systems can be efficiently simulated by classical algorithms.

Stabiliser formalism based methods (as with Virmani et al. in [42]) show that noisy quantum systems with noise rates above the upper threshold bound enter regimes that can use the Gottesman-Knill theorem (see section (2.2.3.3)).

An important result following the approach in [42] is that by Buhrman et al. in [43], where a restricted model was considered composed of perfect Clifford operations and access to an arbitrary non-Clifford single qubit gate subject to depolarising noise. As already mentioned such a set of gates are universal for quantum computation.

It was shown that at depolarising noise rates of ≈ 0.453 any arbitrary single qubit gate could be expressed as a convex sum of single qubit Clifford gates and therefore follow the Gottesman-Knill result. Following on from this Plenio et al [5] again considered a system composed of Clifford and a non-Clifford resource. Here noise was added to Clifford resources and then commuted around to specific Clifford and non-Clifford resources. A variety of noise models were considered yielding noise rates of the range of approximately 0.0301 to 0.2605.

The most relevant method for this thesis is that using the Harrow and Nielsen algorithm [6], which will be outlined in more detail in section (3.5). Their approach considers quantum devices composed of single qubit gates and a restricted set of two qubit gates. If a quantum device is affected by sufficient gate noise then the two qubit gates forming the device lose their ability to generate quantum entanglement, and the enter the specified restricted set which can be efficiently simulated using the algorithm. In [6], they consider the CX gate as the two qubit gate, and show that probabilistic

depolarising noise acting immediately on both output qubits at a rate of ≈ 0.74 is sufficient to ensure the noisy CX gate can't generate entanglement. In other words they show that ≈ 0.74 is an upper bound value for the threshold. Under a more adversarial noise model this rate is lowered to 0.5. Virmani et al. in [42] sharpen this result to ≈ 0.67 joint depolarising noise acting on the CX gate.

3.4 Classical simulations of quantum systems

3.4.1 Types of classical simulation

There has been other interest in classically simulating quantum computers outside of determining the upper threshold bound values. In this section we will introduce two types of classical simulation specified by Jozsa et al. [44, 45, 46] that some of the key results in the following sections fit into.

Definition 3.4.1. ***Strong efficient classical simulation***

Suppose we have a quantum circuit $\mathcal{C}$ acting on a register of n qubits and consisting of at most $\text{poly}(n)$ *gates acting on any individual qubit. Let Λ denote the set of input states, and Π the set of output observables.*

$\mathcal{C}$ is strongly efficiently simulatable with respect to Λ and Π if we have a deterministic classical algorithm that can compute all expectation values $\text{tr}\left[\mathcal{C}(\rho)\pi\right]$ *to an accuracy of m digits within* $\text{poly}(n, m)$ *time, for every $\rho \in \Lambda$ and $\pi \in \Pi$.*

This notion of simulation can be made weaker by lessening the accuracy required, for example, rather than using an accuracy of m, a weaker notion of simulation may require computation of expectation values to an accuracy of $\log(m)$ digits.

The second type of simulation is defined as the following:

Definition 3.4.2. ***Weak efficient classical simulation***

A quantum circuit $\mathcal{C}$ acting on a register of n qubits is weakly efficiently simulatable if we have a probabilistic classical algorithm that can sample once from a probability distribution suitably close to the output probability distribution in $\text{poly}(n)$ *time.*

As we are sampling only once at the end, this simulation type is more "natural" in that it mimics quantum measurements in a lab situation. The ability to be able to strongly simulate a system, implies that we can also weakly simulate it [47], but there may exist quantum circuits that can only be weakly simulated and not strongly. As a weaker property, it may be more straightforward to consider weak simulation over strong.

3.4.2 Other examples of classical simulation

Within the literature the majority of simulation schemes fit into the strong simulation definition.

Looking back at the two schemes already mentioned; as it is easy and efficient to track the evolution of states and calculate all marginal probability distributions within the stabiliser formalism, the Gottesman-Knill theorem corresponds to a strong simulation.

The Harrow and Nielsen algorithm is based around repeated sampling during the evolution and measurement of states we therefore have a weak simulation.

Some other notable methods of classical simulation that will be worth briefly mentioned here are Matchgate Circuits [48, 44], Matrix Product States [49, 50], and methods using the Tensor Network formalism [49, 51]. It is worth noting that unlike those mentioned before, all these methods consider the topology of the qubits.

We will begin by first defining matchgates:

Definition 3.4.3. ***Matchgates and matchgate circuits***

A matchgate G is a two qubit unitary gate of the form:

$$G = \begin{bmatrix} a & 0 & 0 & b \\ 0 & w & x & 0 \\ 0 & y & z & 0 \\ c & 0 & 0 & d \end{bmatrix} \qquad where \begin{bmatrix} a & b \\ c & d \end{bmatrix}, \begin{bmatrix} w & x \\ y & z \end{bmatrix} \in SU(2) \tag{3.2}$$

Matchgate circuits are quantum circuits composed of nearest neighbour matchgates.

As with the Gottesman-Knill theorem, matchgates provide a compact way of describing certain states and following their evolution through the quantum circuit in an efficient way. A result proven by Valiant [48] shows that we can perform a strong classical simulation of poly-sized matchgates circuits acting only on nearest neighbour qubits for any arbitrary input. Just as with Clifford circuits and the Gottesman-Knill theorem, matchgate circuits can generate entanglement whilst also being efficiently classically simulatable [52]. If the nearest neighbour restriction is relaxed so that the matchgates act on nearest neighbour and next nearest neighbour qubits, then the matchgate circuit is can efficiently perform universal quantum computation.

Within the Matrix Product formalism n-qubit quantum states are be represented uniquely by a set of n-matrices. The idea being that the number of variables required to specify the n-matrices is fewer than the 2^n independent variables that would be required to specify an arbitrary state. The sizes of the matrices is dependent only the maximal Schmidt rank over the possible partitions of the n qubits, and if the rank is

polynomial in n, then performing single qubit gates and measurements, and two qubit gates on nearest neighbour qubits can be simulated efficiently.

Matrix product states are an example of the broader class of tensor network states. While a detailed description is beyond the scope of this thesis, in summary within the tensor network formalism quantum states and operators are represented by sets of tensors, with computational steps corresponding to contractions of these tensors. Just as with matrix product states, we may reduce the number of parameters we need to represent states by representing them with tensors with smaller ranks, reflecting the expected entanglement properties of states. If the number of parameters describing the these tensors are sufficiently small, then we have an efficient description of the state. By describing states using tensors, the computation of a quantum circuit has a can be mapped to a graph, with vertices representing gate operations and edges corresponding to the individual qubit "wires" connecting gates. The action of a circuit can then be determined by contracting the vertices in the network.

3.5 The Harrow and Nielsen algorithm

In this section we will go into detail about the Harrow and Nielsen algorithm [6] (shortened to H&N), as the algorithm we consider in our research is an application of the H&N algorithm in a modified context. We will begin by defining the set of operations the algorithm is limited to. In brief, the H&N algorithm shows that gate model quantum computers without entanglement can be efficiently simulated classically.

3.5.1 The separability preserving operations and subsets

The *separability preserving* (SP) operations are a wide class of two qubit quantum operations that are defined to have separable output states for separable input. Harrow and Nielsen [6] state that there is currently no known simple characterisation of separability preserving operations, which is why they chose to instead study *separable operations*, and, *SWAP + separable operations* - sub-sets of the separability preserving operations that have stronger properties than separability preserving operations [6]. It is important to note that the set of SP operations are convex and closed under composition.

3.5.1.1 Characterising separable operations

In this subsection we will look at the properties of separable operations. They were introduced in [53, 54], and have the following definition:

Definition 3.5.1. ***Separable Operations***

Let $\mathcal{E}$ correspond to a completely positive, trace-preserving map acting on a composite quantum system composed of two subsystems labelled A and B. $\mathcal{E}$ is said to be a separable operation if it has a Kraus representation as:

$$\mathcal{E}(\rho) = \sum_i (A_i \otimes B_i)\rho(A_i^\dagger \otimes B_i^\dagger) \tag{3.3}$$

By a SWAP + separable operation, we mean a two qubit operation, consisting of first swapping the two qubits then performing a separable operation on the two swapped qubits.

The complete set of separable operations plus SWAP + separable operations do not form a complete set of SP operations [42]. The set of SWAP + separable operations is distinct from the set of only separable operations, as SWAP + separable operations are capable of generating entanglement with the aid of local ancillary systems.

We will now discuss the properties of separable operations by looking at their associated Choi-Jamiołkowski (CJ) state as the properties of the operations translate into properties of the CJ states. As in section (1.2.4), we will follow the convention used by Harrow and Nielsen when describing separability of certain sub-systems as cuts across those specific subsystems.

Suppose we have a quantum system composed of the sub-systems R_A, A, B, R_B, where R_A, R_B are copies of A and B respectively. The subsystems A and B have Hilbert space dimensions d_A and d_B. The canonical, maximally entangled states of the subsystems $R_A A$, and BR_B are:

$$|\alpha\rangle = \frac{1}{\sqrt{d_A}}\sum_i |i_{R_A} i_A\rangle \qquad \text{and} \qquad |\beta\rangle = \frac{1}{\sqrt{d_B}}\sum_i |i_B i_{R_B}\rangle \tag{3.4}$$

Let $\rho_{\mathcal{E}}$ denote the CJ state associated to the quantum operation $\mathcal{E}_{AB}$ acting only on the subsystems A and B:

$$\rho_{\mathcal{E}} = \left(\mathbb{1}_{R_A} \otimes \mathcal{E}_{AB} \otimes \mathbb{1}_{R_B}\right)\left(|\alpha\rangle\langle\alpha| \otimes |\beta\rangle\langle\beta|\right) \tag{3.5}$$

For $\rho_{\mathcal{E}}$ to be a valid CJ state the reduced state $(\rho_{\mathcal{E}})_{R_A R_B}$ must be a maximally mixed state of $R_A R_B$. A straightforward characterisation of separable operators is presented in the following theorem (see [55]):

Theorem 3.5.2. ***Operator-separability theorem***

Suppose we have a composite quantum system composed of the sub-systems R_A, A, B, R_B. Let the operation $\mathcal{E}_{AB}$ (as in eq. (3.5)) be trace-preserving. $\mathcal{E}_{AB}$ is separable across the $R_AA{:}BR_B$ cut, if and only if $\rho_{\mathcal{E}}$ can be written in the form:

$$\rho_{\mathcal{E}} = \sum_i p_i \rho_i^{R_AA} \otimes \rho_i^{BR_B} \tag{3.6}$$

where p_i is a probability, and $\rho_i^{R_AA}$, $\rho_i^{BR_B}$, are density operators of the subsystems R_AA, and BR_B respectively.

In other words $\mathcal{E}_{AB}$ is separable when $\rho_{\mathcal{E}}$ is separable. It is important to note that not all separable states of the $R_AA{:}BR_B$ cut correspond to CJ operators $\rho_{\mathcal{E}}$ as in eq. 3.6. When $\mathcal{E}_{AB}$ is trace-preserving then $\mathrm{tr}_{AB}(\rho_{\mathcal{E}})$ must be the maximally mixed state of R_AR_B. In general, separable states σ of $R_AA{:}BR_B$ have $\mathrm{tr}_{AB}(\sigma)$ that are not maximally mixed.

We can finally list the characterisation as:

Theorem 3.5.3. ***The set of separable operation CJ operators*** **[55]**

The set of density operators σ of the composite system R_AABR_B such that $\sigma = \rho_{\mathcal{E}}$ as in 3.6 where $\mathcal{E}_{AB}$ is a trace-preserving quantum operation corresponds to: (a) those σ that are separable with respect to the $R_AA{:}BR_B$ cut, and (b) $\mathrm{tr}_{AB}(\sigma)$ is a maximally mixed state of R_AR_B.

Following on from the characterisation of separable operations, we may characterise SWAP + separable operations using the following CJ representation:

$$\rho(\mathcal{E}) = \sum_j p_j \rho_j^{AR_B} \otimes \rho_j^{R_AB} \tag{3.7}$$

In other words the CJ state representing this type of operations is separable across the $AR_B : R_AB$ cut.

3.5.1.2 Separable machines

Harrow and Nielsen in [6] studied a noisy quantum device restricted to separable operations and SWAP + separable operations. They termed such devices *separable machines*, defining them to be circuit model based quantum computers, capable of adaptable computation that are restricted to using only the following procedures:

1. Preparation of single qubit states in some fixed state.
2. Able to perform some arbitrary set of single qubit quantum operations i.e. unitary transformations, measurement, etc.
3. Able to perform operations from the arbitrary set of two qubit operations that can be expressed as convex combinations of (a) separable operations, and (b) SWAP + separable operations.

3.5.2 Overview of the algorithm

We will now provide a summary of the H&N algorithm. The main technical concern is with the accuracy required to adequately simulate the noisy quantum computer, and how increased accuracy of the simulation relates to the increase in computational overhead. We will however not go into any detail on this as Harrow and Nielsen's arguments regarding accuracy in [6] are directly applicable to our algorithm.

The separable machines looked at by Harrow and Nielsen were modelled as follows: The initial state of the quantum system was prepared in the computational basis state $|x\rangle$, after which a quantum circuit composed of single qubit, and two qubit separable/SWAP + separable gates was performed. This circuit was taken from $\{C_n\}$ - the uniform family of quantum circuits[1] containing $p(n)$ gates that act on $q(n)$ qubits, where $p(n)$ and $q(n)$ are polynomials in n. The computation was concluded by performing a computational basis measurement on some or all qubits yielding measurement outcomes y with an associated probably distribution $p_x(y)$.

The classical algorithm for input x, outputs result y, with the probability distribution $\tilde{p}_x(y)$. It was argued that for the classical simulation to be "accurate", the distribution $\tilde{p}_x(y)$ must approximate $p_x(y)$ sufficiently well. Note the algorithm as a weak simulation, aims not to recreate the whole distribution $p_x(y)$, but rather efficiently sample from the approximation $\tilde{p}_x(y)$. The H&N algorithm can achieve these accuracies with a classical computational overhead scaling as polynomial function of $p(n)$ not dependent on the circuit family $\{C_n\}$.

We will now go into more detail on the classical simulation procedure, starting by specifying the variables used in the classical simulation, before describing how these variables are updated when enacting gate operations and measurements.

Each qubit $j = 1, \ldots, q(n)$ of the separable machine is represented in the classical algorithm by a three element real vector $\vec{s}_j$. Similarly the multipartite state of the whole system is represented by a $3q(n)$-dimensional real vector $\vec{s}$ that can be constructed from each $\vec{s}_j$, as $\vec{s} \equiv (\vec{s}_1, \ldots, \vec{s}_{q(n)})$. Each sub-vector $\vec{s}_j$ is called *valid* if its elements are: (a)

[1] A quantum circuit C computes the function $f : \{0,1\}^n \to \{0,1\}$, its size corresponds to the number of logical gates forming the circuit. Here we say a circuit family consists of a set of quantum circuits $\{C_n\}$ for each circuit input length. If a family of circuits are computable with a running time polynomial in n then we say that they are uniform.

specified to l bits of precision, and (b) ensure that $\|\vec{s}_j\| \leq 1$. If every sub-vector $\vec{s}_j$ forming $\vec{s}$ is valid, then $\vec{s}$ is called valid too. When a given vector $\vec{s}$ is valid it has a corresponding density operator that can be expressed as:

$$\rho(\vec{s}) \equiv \frac{\mathbb{1} + \vec{s}_1 \cdot \sigma}{2} \otimes \cdots \otimes \frac{\mathbb{1} + \vec{s}_{q(n)} \cdot \sigma}{2} \qquad \sigma \equiv (X, Y, Z) \tag{3.8}$$

Within the classical simulation the valid vectors $\vec{s}$ are used as the variables of the simulation - used to represent the quantum states $\rho(\vec{s})$. Each $\vec{s}_j$ is a single qubit Bloch vector, and the space of all valid sub-vectors $\vec{s}_j$ corresponds to the Bloch sphere. Harrow and Nielsen state that at l bits of precision the simulation is effective enough to attain sufficient accuracy.

The computation is started on the separable machine by preparing an $q(n)$ qubit quantum state $|x\rangle$. Here x is a $q(n)$ digit long binary string that can be expanded as $x_1 \cdots x_{q(n)}$, with each x_j corresponding to the j^{th} qubit's computational basis state. If $x_j = 0$ then we have a corresponding $\vec{s}_j = (0, 0, 1)$ stored in the simulation, and if $x_j = 1$ then $\vec{s}_j = (0, 0, -1)$.

We can now describe how the simulation approximates the quantum gates forming the circuit of the separable machine. The transformation of quantum states corresponds to updating the $\vec{s}$ vectors in the simulation. All quantum gates are treated by the simulation as bi-partite, with single qubit gates as special cases where one of the qubits is acted upon trivially.

Let $\mathcal{E}_{AB}$ denote a separable two-qubit gate acting on qubits A and B. The simulation procedure takes the vector $\vec{s}$ as input, performs a stochastic operation on it before outputting another valid $3q(n)$-dimensional vector $\vec{s}'$. The vector $\vec{s}$ is then replaced with $\vec{s}'$ in the simulation, and the process is then repeated over for each of the $p(n)$ gates of the quantum system. After all of the gates a final output vector $\vec{s}$ is produced, and a simulation of the final measurement takes place.

Looking at the action of $\mathcal{E}_{AB}$ more explicitly; at some stage of the computation the state of the system is represented in the simulation by a valid vector $\vec{s}$ corresponding to the following density operator:

$$\rho(\vec{s}) = \frac{\mathbb{1} + \vec{s}_1 \cdot \sigma}{2} \otimes \cdots \otimes \frac{\mathbb{1} + \vec{s}_A \cdot \sigma}{2} \otimes \frac{\mathbb{1} + \vec{s}_B \cdot \sigma}{2} \otimes \cdots \otimes \frac{\mathbb{1} + \vec{s}_{q(n)} \cdot \sigma}{2} \tag{3.9}$$

The density operator for just qubits A and B can be expressed as:

$$\rho(\vec{s}_{AB}) = \frac{\mathbb{1} + \vec{s}_A \cdot \sigma}{2} \otimes \frac{\mathbb{1} + \vec{s}_B \cdot \sigma}{2} \tag{3.10}$$

The set of separable two qubit states as in eq. (3.10), form a convex set with pure states corresponding to the extrema of the set. Following on from section (1.2.3) we may represent the separable two qubit states geometrically as a convex shape in a

15-dimensional space. As the output of $\mathcal{E}_{AB}(\rho(\vec{s}_{AB}))$ remains separable we can use Caratheodory's Convex Hull Theorem[2] to express it as the convex sum of at most 16 points within the convex shape as:

$$\sum_i q_i \frac{\mathbb{1} + \vec{t}_A^{\,i} \cdot \sigma}{2} \otimes \frac{\mathbb{1} + \vec{t}_B^{\,i} \cdot \sigma}{2} \tag{3.11}$$

where i are positive integers up to 16, $\sum_i q_i = 1$, and $\vec{t}_A^{\,i}$, $\vec{t}_B^{\,i}$ are real three element vectors satisfying $\left\|\vec{t}_A^{\,i}\right\|, \left\|\vec{t}_B^{\,i}\right\| \leq 1$. The operators $\left(\frac{\mathbb{1}+\vec{t}_A^{\,i}\cdot\sigma}{2} \otimes \frac{\mathbb{1}+\vec{t}_B^{\,i}\cdot\sigma}{2}\right)$ each correspond to the extrema of the convex shape - pure states of the "Bloch sphere tensor Bloch sphere".

The simulation aims, for an input $\vec{s}$, to find a set of three-vectors $\vec{r}_A^{\,i}$ and $\vec{r}_B^{\,i}$ and at most a 16-element probability distribution p_i that can approximate the q_i, $\vec{t}_A^{\,i}$, $\vec{t}_B^{\,i}$'s in eq. (3.11) to l-bits of precision and ensure that the corresponding density operator is close enough to $\mathcal{E}_{AB}(\rho(\vec{s}_{AB}))$.

Writing $\mathcal{E}_{AB}(\rho(\vec{s}_{AB}))$ out as in eq. (3.11) is not straightforward, as finding the distribution q_i, and vectors $\vec{t}_A^{\,i}$, $\vec{t}_B^{\,i}$ is non-trivial, however in [6] it is stated that a brute force search over all valid probability distributions and vectors requires a polynomial number of operations as a function of n, which is sufficient for the simulation.

With a suitable set of $\vec{r}_A^{\,i}$, $\vec{r}_B^{\,i}$, and distribution p_i found, the simulation outputs a valid $q(n)$ vector $\vec{s}'$ as follows: Let $j = 1, \ldots, A, B, \ldots q(n)$, for $j \neq A$, B the algorithm sets $\vec{s}'_j = \vec{s}_j$, that is, the input vector's variables. With a probably p_i it selects one extrema from the convex set - one pair $\vec{r}_A^{\,i}$, $\vec{r}_B^{\,i}$, and sets $\vec{s}_A{}' = \vec{r}_A^{\,i}$, and $\vec{s}_B{}' = \vec{r}_B^{\,i}$. The new output vector $\vec{s}'$ is finally formed from the $\vec{s}'_j$ vectors.

To repeat, the action of a separable operation $\mathcal{E}_{AB}$ is to transform the input vector $\vec{s}$ as

$$(\vec{s}_0, \ldots, \vec{s}_A, \vec{s}_B, \ldots, \vec{s}_{q(n)}) \overset{\text{with prob } p_i}{\Longrightarrow} (\vec{s}_0, \ldots, \vec{r}_A^{\,i}, \vec{r}_B^{\,i}, \ldots, \vec{s}_{q(n)}) \tag{3.12}$$

The simulation is concluded with a computational basis measurement as follows:

Let $\mathcal{K}$ be the subset of qubits that are to be measured at the end of the quantum computation. As the third component of every $\vec{s}_j$ corresponds to the Pauli Z component of the block vector, for each qubit $k \in \mathcal{K}$ we will denote the third component of the associated $\vec{s}_k$ vector as s_k^3. A computational basis measurement of qubit k would then yield the result 0 with probability $\frac{1}{2}(1+s_k^3)$, and the result 1 with probability $\frac{1}{2}(1-s_k^3)$.

[2] If $x \in R^d$ lies in the convex hull of set P then there exists a subset P' of P consisting of $d+1$ or fewer points such that x lies in the convex hull of P'.

Summary of the algorithm

In summary, provided that the operations forming a quantum circuit are separable operations, or, SWAP + separable operations, the algorithm can simulate each one with at most polynomial in n overhead. Harrow and Nielsen proved in [6] that the algorithm can sample efficiently from $\tilde{p}_x(y)$, a distribution arbitrarily close to the intended circuit's measurement probability distribution $p_x(y)$. They note that their proof only requires the operations to only be separability preserving, and so the algorithm can extend to the broader class of operations.

3.6 Introduction to generalised probability theories

In the research presented in the following chapter we modify the Harrow and Nielsen algorithm by loosening the what the valid variables of the simulation are, choosing single particle state spaces that are not the Bloch sphere. This was inspired by ideas from generalised probability theories (GPT), that will be briefly introduced here. A good introduction to GPTs can be found in [56, 57].

At present it still remains a mystery exactly which key features of the quantum formalism are responsible for providing the power of quantum computation over classical. Indeed whether quantum computation is better than classical computation remains unproven, but accepted. Quintessential features of quantum mechanics absent in classical mechanics like entanglement, non-locality and so on are often pointed to as the source of this computational speed-up.

Our understanding of the connections between features of physical theories and information processing can be deepened by looking at a broader range of "physical" theories. The features of these hypothetical theories can be then contrasted with quantum and classical mechanics, and insight can be gained.

The extended Harrow and Nielsen algorithm presented in this thesis will be set in a GPT we construct. This GPT, despite being very similar in structure to noisy quantum mechanics in terms of unitary dynamics, will have an inherent restriction to the set of measurements allowed/chosen. The restricted measurements allow us to consider a redefined single particle state space within the GPT, that contains both quantum and non-quantum operators. These state spaces are chosen as sets of operators that give positive outcome probabilities for the restricted measurements, and are required to reflect the symmetries of the dynamics to ensure that the GPT is consistent and measurement outcome probabilities remain positive.

The new single particle state space construction allows us to consider a generalised notion of entanglement over these new state spaces, and our algorithm is able to efficiently classically simulate a class of noisy quantum computers that are unable to generate generalised entanglement with respect to our new state spaces, in analogy to the Harrow

and Nielsen algorithm simulating noisy quantum systems unable to generate quantum entanglement. In the cases where the noisy dynamics cannot generate generalised entanglement, our GPT essentially becomes a classical theory. We will specify these state spaces in detail in the the next chapter.

3.7 Summary of Chapter (3)

This chapter began by specifying three regions of values for the fault tolerant noise threshold; a region below the lower bound to the threshold in which noisy quantum computers can efficiently simulate ideal universal quantum computers using fault tolerant computing techniques, an upper bound above which we believe we can gain no meaningful or advantageous computation because it can be classically simulated efficiently, and an intermediately region in which noisy quantum computation is believed to be unable to efficiently simulate ideal universal quantum computers, whilst also not being efficiently classically simulatable. We highlighted the key results and methodologies to obtain threshold bound values in each region before focussing on classical simulation methods to determine the upper bound to the threshold. We defined classical simulation and mentioned some alternative approaches before discussing the main topic of the chapter - the Harrow and Nielsen algorithm.

We started by specifying the separability-preserving operations the algorithm is limited to, and then went into more detail on the algorithm, describing the algorithms state space, and its simulation of quantum dynamics and measurement. The section was concluded by mentioning Bi-entangling machines as extensions of those covered by Harrow & Nielsen. The chapter finally ended with a brief introduction to generalised probability theories and generalised entanglement which served as an inspiration for us to consider extensions of the Harrow & Nielsen's algorithm with modified single particle state spaces in the next chapter.

4 Introduction to remaining chapters

4.1 Introduction

We will start this chapter with a brief summary of the motivations behind the research, listing the key ideas, before introducing the technical problems studied.

The difficulty of simulating quantum mechanics has been a known computational problem for a long time [58]. As already mentioned in earlier chapters, an immediate problem is how one deals with the exponential growth in the number of parameters required to describe quantum states of increasing numbers of qubits or of greater degrees of freedom. This is a particular problem when considering many-body systems with complex entangled states.

Just as in Chapter (3), we are lead to the question of "when can quantum systems be efficiently classically simulated?" While this is a broad question with applications in many areas, in this thesis we are interested in applying it to the study of noisy quantum computation, in particular determining noise levels at which noise rates noisy quantum computation become classically simulatable.

Approaches to obtain such classical simulation thresholds bounds involve: (a) firstly identifying a quantum regime that is restricted enough to have efficiently classical simulatable dynamics, and then (b) determining the amount of noise required to take a noisy quantum system into that regime. Broadly speaking, previous approaches have followed two routes - there are those that show noisy quantum systems enter regimes restricted to Clifford operators, using the Gottesman-Knill result as in section (2.2.3.3), and those that consider regimes with limited entanglement, as with the Harrow and Nielsen algorithm in section (3.5).

The approach taken in our research is to consider an extension of the Harrow and Nielsen algorithm to use generalised notions of entanglement for systems that have restricted measurements. As we will discuss shortly, it is through these restricted measurements that we can replace the single particle state space of the H&N algorithm with a set of operators $\mathcal{S}$ containing both quantum and non-quantum states. The non-quantum separability considered here is specified with respect to the state space S, instead of the quantum Bloch sphere. If a noisy quantum computer with restricted measurement enters a regime in which all its dynamics map operators from $\mathcal{S}$ to other operators in $\mathcal{S}$, and the dynamics are not capable of generating entanglement with respect to $\mathcal{S}$, then it can be efficiently simulated classically. We will show that, for

certain choices of $\mathcal{S}$ we require less noise for efficient classical simulation, and that we may be able to efficiently classically simulate a slightly broader range of quantum regimes than other methods.

In this thesis we principally consider magic state based architectures that are restricted to Clifford operations (see section (2.3)). Restrictions to the set of available measurements arise in fault tolerant quantum computation for the following reasons: firstly, many error correction schemes require only measuring a certain set of observables, e.g. the computational basis states for a number of qubits; secondly, that measurements themselves are susceptible to noise and so are restricted. We will go into more detail about how the limited available measurements, and Clifford dynamics of magic state architectures restrict $\mathcal{S}$ later in this chapter.

The only Clifford operation that entangles operators from our chosen $\mathcal{S}$ in a generalised sense will be the bipartite CZ operation (see theorem (9.6)). The main technical problem of our research is to determine the amount of noise to make a noisy CZ gate generalised separable with respect to $\mathcal{S}$. Our aim was, for a given noise model, to find a choice of $\mathcal{S}$ requiring the least amount of noise on a CZ gate for generalised separability with respect to $\mathcal{S}$. While we were able to improve upon previous results, were often unable to find analytic results, and the optimal choice of $\mathcal{S}$.

The first half of this chapter will define $\mathcal{S}$, in relation to the restrictions on the available measurements. We will define a generalised non-quantum notion of entanglement with respect to $\mathcal{S}$, before considering an illustrative example of $\mathcal{S}$ - the *Bloch cube*. Next we will present the main technical problem of the research. The second half of the chapter will start by relating observations about *cube-entanglement* to quantum entanglement and prior work. These observations will provide important further motivation to the research. and go through the variations of the problem considered in this thesis. Section (4.5.4) provides an overview of these variations. The chapter ends by introducing new notation used through the remainder of the thesis.

4.2 Operator sets $\mathcal{S}$, and $\mathcal{S}$-separability

The simulation capabilities of our new algorithms are tied to the choice of *single particle state space* $\mathcal{S}$ for those algorithms. Here, a "state" is considered to be a description that allows us to sample measurement outcomes, albeit for a restricted choice of measurements. We begin by defining $\mathcal{S}$ before specifying how restrictions to the available measurements allow us to construct a particular $\mathcal{S}$-set.

Definition 4.2.1. ***The single particle state space $\mathcal{S}$***

To each particle forming a quantum system, we will associate a convex set $\mathcal{S}$ of unit trace 2×2 Hermitian operators ϱ that will represent the state of that particle (within the algorithm). $\mathcal{S}$ may contain operators outside the Bloch sphere.

As with the density operators in the Bloch sphere we expand ϱ in terms of identity and Pauli operators:

$$\varrho = \frac{1}{2}\left(\mathbb{1} + \underline{v} \cdot \underline{\sigma}\right) \qquad \text{where, } \underline{\sigma} = (X,\, Y,\, Z)^T \tag{4.1}$$

Here $\underline{v}$ is analogous to the Bloch vector, although crucially $|\underline{v}|$ may be larger than 1 for some ϱ. The probabilities associated measurement outcomes can be obtained by applying the Born rule[1] to ϱ.

Definition 4.2.2. ***$\mathcal{S}$ as the dual to a measurement set.***

Let $\mathcal{M}$ denote a set of single particle measurement operators. We can define a dual set $\mathcal{M}^*$ of operators to $\mathcal{M}$ through the Born rule, as follows:

$$\mathcal{M}^* := \{\rho \,|\, \mathrm{tr}\,[\rho] = 1,\, \mathrm{tr}\,[\rho M] \geq 0,\, \forall M \in \mathcal{M}\} \tag{4.2}$$

The set $\mathcal{M}^*$ is the largest a single particle state space $\mathcal{S}$ can be, as operators outside of $\mathcal{M}^*$ will return negative probabilities for measurements in $\mathcal{M}$.

The following example will help visualise how we can construct a set $\mathcal{M}^*$ for a given $\mathcal{M}$. We will look at how Pauli X measurement projectors bound the set of operators by two surfaces:

Example 4.2.3. ***Pauli X measurements.***

The projectors for this measurement take the form:

$$P_{X+} = \frac{1}{2}\left(\mathbb{1} + X\right), \qquad P_{X-} = \frac{1}{2}\left(\mathbb{1} - X\right) \tag{4.3}$$

Operators ρ within $\mathcal{S}$ can be expanded as:

$$\rho = \frac{1}{2}(\mathbb{1} + \alpha X + \beta Y + \gamma Z) \qquad \alpha,\, \beta,\, \gamma \in \mathbb{R} \tag{4.4}$$

The positivity of $\mathrm{tr}\,[\rho P_{X-}]$ determines the values that α, β, and γ can take:

[1] The probability of obtaining outcome i associated with the measurement operator Π_i is given by $\mathrm{Prob}\,(i) = \mathrm{tr}\,[\rho \Pi_i]$.

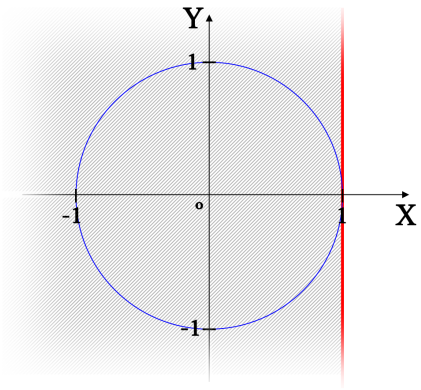


Figure 4.1: Representation of the set of operators on the $X-Y$ plane bounded by the projector P_{X-} under eq. (4.2) (as shaded). The blue circle represents the set operators representing the pure qubit states.

$$\mathrm{tr}\left[\frac{1}{2}\left(\mathbb{1}-X\right)\rho\right] = \frac{1}{2}\mathrm{tr}\left[(1-\alpha)\mathbb{1}+(\alpha-1)X+\ldots\right] = 1-\alpha \geq 0 \tag{4.5}$$

Hence for P_{X-} we require $\alpha \leq 1$ and are free to choose the values of β, γ. The set of valid operators on the $X-Y$ plane bounded by P_{X-} are plotted on figure (4.1). By considering the second projector element P_{X+} we can similarly bound the set of operators on the other side.

The bounds placed from considering a single measurement extend naturally as follows:

Example 4.2.4. ***The Bloch sphere.***

If $\mathcal{M}$ is the set of all quantum measurements, then $\mathcal{M}^*$ is $\mathcal{Q}$ the Bloch sphere.

Of particular importance to much of the motivation behind the research is the case where the $\mathcal{M}$ is the set of Pauli measurements as with magic state architectures. The corresponding set $\mathcal{M}^*$ is termed a *Bloch cube* in analogy to the Bloch sphere.

We will define a Bloch cube state space as:

Definition 4.2.5. ***The Bloch cube state space.***

If $\mathcal{M}$ is the set of Pauli measurements then the corresponding dual $\mathcal{M}^*$ becomes a cube containing both $\mathcal{Q}$ and non-quantum states outside $\mathcal{Q}$ as shown in figure (4.2)*. The extremal states of the Bloch cube have the following eight Bloch vectors:*

$$\{(1,1,1),\,(1,1,-1),\,(1,-1,-1),\ldots,\,(-1,-1,-1)\} \tag{4.6}$$

The surfaces of the Bloch cube represent positive operators ϱ for which the Born rule fires deterministically for one Pauli measurement, that is $\mathrm{tr}\left[\Pi\varrho\right] \in \{0,1\}$ where Π is a projector associated to a Pauli X, Y, or Z measurement. The vertices represent

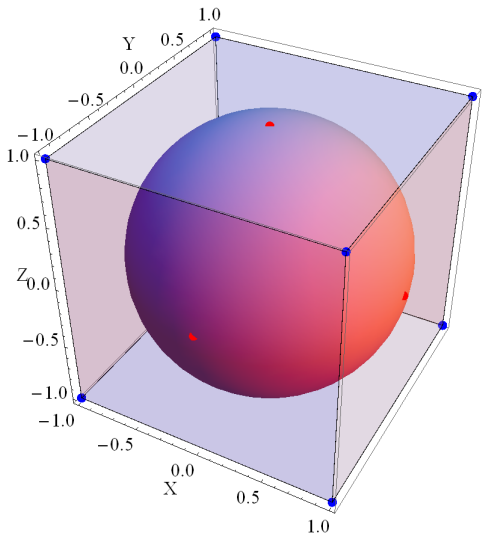


Figure 4.2: Representation of Bloch cube of states, and the Bloch sphere. The set of Bloch sphere forms a sub-set of Bloch cube set. The two sets intersect at the Pauli eigenstates highlighted by the red points.

---

operators that fire deterministically for any of the three Pauli measurements. The six Pauli eigenstates lie at the very center of each cube face.

Note that we can have $\mathcal{S} \subset \mathcal{Q}$ when we allow for noise on measurements/preparations as will be discussed later, and for appropriate choices of $\mathcal{M}^*$. For example, $\mathcal{S}$ may be taken to be the convex hull of the Pauli eigenstates (as the Bloch octahedron from section (2.2.3.3) and (2.3)) dual to the following set of operators:

$$\{M = \frac{1}{2}(\mathbb{1} + \underline{k} \cdot \underline{\sigma})\}$$

$$\text{where } \underline{k} \in \{(1,1,1),\,(1,1,-1),\,(1,-1,-1),\ldots,\,(-1,-1,-1)\}\,,\ \underline{\sigma} = (X,\,Y,\,Z) \quad (4.7)$$

For the most part we will assume our measurements/preparations are ideal and that our state space choices will satisfy $\mathcal{Q} \subseteq \mathcal{S} \subseteq \mathcal{M}^*$.

We can now define $\mathcal{S}$-*entanglement* through $\mathcal{S}$-*separable* multi-partite operators:

Definition 4.2.6. ***$\mathcal{S}$-separability.***

An operator ϱ_{sys} acting jointly over all n particles is defined to be $\mathcal{S}$-separable if it can be expressed as the following convex product:

$$\varrho_{sys} = \sum_{\alpha} p_\alpha \varrho_1^\alpha \otimes \varrho_2^\alpha \otimes \varrho_3^\alpha \otimes \ldots \otimes \varrho_n^\alpha \quad (4.8)$$

where ϱ_i^α is an operator taken from the convex set $\mathcal{S}$ and is associated to the ith particle, and $p_\alpha \in [0,\,1]$ with $\sum_\alpha p_\alpha = 1$. Operators ϱ_{sys} that cannot be expressed as (4.8) are called $\mathcal{S}$-entangled.

Note that with this definition, even single qubit quantum gates can $\mathcal{S}$-entangle operators, as they take operators from $\mathcal{S}$ to operators outside of $\mathcal{S}$. In this thesis, we will be only trying to simulate noisy quantum systems containing multi-partite $\mathcal{S}$-entangling gates, as we will assume that all single qubit gates are restricted so that they preserve $\mathcal{S}$ (as is the case with magic state architectures whose single qubit gates correspond to octahedral symmetry transformations of $\mathcal{S}$).

With our definition of $\mathcal{S}$-entanglement, we can now relate the extended algorithm back to Harrow and Nielsen's in Chapter (3.5). We will represent operators of $\mathcal{S}$ within the extended algorithm as vectors listing elements corresponding to the probabilities for the permitted measurement outcomes. These vectors may be embedded in a real vector space allowing us to also consider linear combinations of states and in particular convex combinations of states.

The extended algorithm can perform a weak classical simulation of a noisy quantum computer with restricted measurements, if the dynamics of the computer preserve $\mathcal{S}$-separability. The extended algorithm simulates single and bipartite $\mathcal{S}$-separable gates identically to the H&N simulation of quantum separable gate, albeit with a modified notion of "valid" vectors to reflect our redefined state space. Measurements are similarly performed. We will later show that some quantum-entangled states are $\mathcal{S}$-separable, so it is possible that our extension of the H&N algorithm could lead to a broader range of efficiently simulatable situations.

4.3 The main problem discussed in this thesis

In order to use the modified H&N algorithm it is required that all of the multi-partite quantum gates forming the noisy quantum device be *$\mathcal{S}$-separable*:

Definition 4.3.1. ***$\mathcal{S}$-separable gates.***

Multi-partite quantum gates are defined to be $\mathcal{S}$-separable if they are unable to output $\mathcal{S}$-entangled operators for all possible $\mathcal{S}$-separable inputs.

Our definition of $\mathcal{S}$-separable quantum operations is analogous to separable operations in the original H&N algorithm, it is worth noting though that unlike in the H&N algorithm single particle unitaries can entangle $\mathcal{S}$ in general. For a choice of $\mathcal{S}$ there will be gates forming the quantum device that are $\mathcal{S}$-entangling, each requiring a varying amount of noise to ensure they become $\mathcal{S}$-separable. For a given quantum system will select the gate U requiring the most noise to become $\mathcal{S}$-separable for a choice of noise model.

Prior to proceeding we will make the following two definitions:

Definition 4.3.2. ***Noisy multi-partite unitaries.***

Let U denote a multi-partite quantum gate, and N_ε some particular noise model whose noise strength is parametrised by $\varepsilon \in [0, 1]$. We will define a noisy multi-partite unitary U_ε as:

$$U_\varepsilon(\cdot) = N_\varepsilon \circ U(\cdot) \tag{4.9}$$

Definition 4.3.3. $\lambda_{U_\varepsilon,\mathcal{S}}$ ***noise parameter.***

For a given noisy quantum gate U_ε and set $\mathcal{S}$, we will define the noise parameter λ as

$$\lambda_{U_\varepsilon,\mathcal{S}} := \min\left\{\varepsilon | U_\varepsilon \in \mathcal{U}^{\mathcal{S}}\right\} \tag{4.10}$$

where $\mathcal{U}^{\mathcal{S}}$ is the set of (noisy) quantum gates that can only output $\mathcal{S}$-separable operators from $\mathcal{S}$-separable input. If there exists no $\varepsilon \in [0, 1]$ such that $U_\varepsilon \in \mathcal{U}^{\mathcal{S}}$ then we will set $\lambda_{U_\varepsilon,\mathcal{S}} = \infty$. The noise parameter $\lambda_{U_\varepsilon,\mathcal{S}}$ is implicitly depends on the noise model considered for U_ε.

This thesis is looking to solve variations of the following technical problem:

Problem 4.3.4.

Suppose that we can prepare each particle in the system from a state in the set $\mathcal{Q}$ and have available the local measurements from $\mathcal{M}$. We are looking to find single particle operator sets $\mathcal{S}$ satisfying $\mathcal{Q} \subset \mathcal{S} \subseteq \mathcal{M}^*$ that require the lowest $\lambda_{U_\varepsilon,\mathcal{S}}$ value for a given U_ε, because then we can efficiently classically simulate the system.

4.4 Contrasts between quantum and Bloch cube entanglement

This section will present observations about the relationship between non-quantum Bloch cube entanglement, quantum entanglement and non-locality, to provide better context for the main problem discussed in the previous section. The observations here, were originally presented in [7]. We will begin by defining *operator compatib*le sets of measurements, relating such sets to non-locality and classical simulation.

Definition 4.4.1. ***Operator compatibility.***

Suppose we have some set of measurement operators $\mathcal{M} = \{M_{ij}\}$ with associated measurement outcomes m_{ij}. For measurement i the probability of obtaining outcome m_{ij} is given by the Born rule as in eq. (1.29).
The set $\mathcal{M}$ is said to be "operator compatible" if there exists some (possibly non-positive operator) ϱ that fires deterministically for all possible measurement combinations, that is, for the outcome combination $(m_{1j}, m_{2j}, m_{3j}, \dots)$ we have $(\mathrm{tr}\left[M_{1j}\varrho\right], \mathrm{tr}\left[M_{2j}\varrho\right], \mathrm{tr}\left[M_{3j}\varrho\right], \dots)$ equalling some binary string.

An example of just such a set of operator compatible measurements would be Pauli X, Y, and Z measurements. We denote e, f, g as the eigenvalues of each measurement X, Y, Z respectively. As the set of measurements are operator compatible there must exist an operator ϱ that would fire deterministically for all combinations of the three measurements, namely $tr[M_e\varrho]$, $tr[M_f\varrho]$, and $tr[M_g\varrho] \in \{0, 1\}$, where M_e, M_f, M_g, represent the projector elements for the measurement results e, f, g. The operators $\varrho = \frac{1}{2}(\mathbb{1} + eX + fY + gZ)$ achieve these conditions under the Born rule and correspond to the corners of the Bloch cube.

In general, sets of quantum measurements are not operator compatible, but sets that are compatible are characterised geometrically by the following:

Lemma 4.4.2. ***Characterising operator compatibility.***

Suppose we have a quantum system of dimension d, and a set of N POVMs with n_i outcomes each. The set of POVMs can only be operator compatible if the following constraint is satisfied:

$$\sum_{i=1,\ldots,N} n_i \leq d^2 + N - 1 \tag{4.11}$$

We can see that a set of operator compatible measurements must satisfy this criterion by looking on the constraints placed on operators acted upon by these measurements. Let $\mathcal{A}$ denote a $d \times d$ dimensional trace-one Hermitian matrix, and the ith POVM to have elements labelled $(P_i)_j$. Following on from subsection (1.2.3), we see that $\mathcal{A}$ may be expanded in terms of trace and traceless Hermitian operators and be characterised by a $(d^2 - 1)$ element real vector $\underline{r}_{\mathcal{A}}$. If the set of POVMs are operator compatible then there must exist an operator $\mathcal{A}$ for which all the POVMs must return probability values as $\mathrm{tr}\left[(P_i)_j\mathcal{A}\right] \in \{0, 1\}$. Each $\mathrm{tr}\left[(P_i)_j\mathcal{A}\right]$ can be expressed as a dot product of two d^2 element real vectors, and so the set of all elements for all N POVM can be combined into a $(\sum_{i=1,\ldots,N} n_i) \times d^2$ dimensional matrix $\boldsymbol{P}$. This transform the problem into an linear algebra one of finding solutions to $\boldsymbol{P}\underline{r}_{\mathcal{A}} = \underline{b}$ where $\underline{b}$ is a binary string of outcome probabilities. Solutions can only exist to this problem when eq. (4.11) is satisfied.

In the non-trivial instance where $N > 1$, and where all N POVMs have d measurement outcomes, the condition for operator compatibility reduces to the following:

$$N \cdot d \leq d^2 + N - 1 \;\Rightarrow\; 0 \leq (d-1)(1+d-N) \;\Rightarrow\; N \leq 1 + d \tag{4.12}$$

A set of measurement is called tomographically complete if a state ϱ, may uniquely fix and be fixed by taking the traces of ϱ with the measurement operators of the tomographically complete set. The property of operator compatibility for sets of measurements

has importance to classical simulatability if those measurements are also tomographically complete, as matrices formed from Born measurement probabilities and definition (4.2.6)) are separable with respect to single particle state space (and usable with the extended H&N algorithm) if and only if they satisfy a local hidden variable model:

Lemma 4.4.3. ***Multiparticle states, separability and local hidden variables.***

Let $\mathcal{M}^*$ be a single particle state space, dual to a set of tomographically complete operator compatible measurements $\mathcal{M}$. A multiparticle state ρ is $\mathcal{M}^*$-separable if and only if there is a local hidden variable model (LHVM) [59] *for measurements from $\mathcal{M}$ on ρ.*

Proof. For simplicity we will only consider two particle systems, as it is straightforward to extend the arguments to multi-partite cases. As the measurements $\mathcal{M}$ are tomographically complete the probability distribution of measurement outcomes from $\mathcal{M}$ exactly characterises ϱ.

Let P_a and P_b denote POVMs corresponding to measurements in the a and b "directions". A bi-partite probability distribution that satisfies a LHVM by definition can be expressed as

$$\mathrm{tr}\left[\varrho((P_a)_i \otimes (P_b)_j)\right] = \sum_{\gamma} p(\gamma)p(i|a,\gamma)p(j|b,\gamma) \tag{4.13}$$

where $(P_a)_i$ and $(P_b)_j$ correspond to the i^{th} and j^{th} POVM elements, $p(\gamma)$ is a probability distribution over the hidden variable γ, and $p(i|a,\gamma)$, $p(j|b,\gamma)$ correspond to the probability distributions for the local measurements on each particle. These local measurement distributions can then be expressed as statistical mixtures of local deterministic distributions, where each measurement choice specifies a single deterministic outcome. The local deterministic distributions can then be matched, via the Born rule, to extrema points of $\mathcal{M}^*$ and by tomographic completeness the mixture of these extrema equals ρ . The state ϱ therefore, is separable with respect to $\mathcal{M}^*$ if and only if it satisfies a LHVM for the measurements in $\mathcal{M}$. □

Returning to the case of Pauli measurements, a state is then cube-separable if, and only if, it has a LHVM for the Pauli measurements. Following the proof above we can explicitly show this; suppose we had a two particle system with sub-systems denoted A and B, and we wished to perform some Pauli X axis measurement on system A and Y on system B. These measurements have associated POVM elements $(P_X)_i$ and $(P_Y)_j$ with measurement outcomes i, j respectively. For the state ρ, we may express the probability of getting the two measurement outcomes i and j as:

$$p(i,\, j|X,\, Y) = tr[(P_X)_i \otimes (P_Y)_j \rho] \tag{4.14}$$

If the state ρ is cube-separable, we may express (4.14) as:

$$p(i,\, j|X,\, Y) = tr[(P_X)_i \otimes (P_Y)_j \left(\sum_k p_k \sigma_k^A \otimes \sigma_k^B\right)]$$

$$p(i,\, j|X,\, Y) = \sum_k p_k \mathrm{tr}\left[(P_X)_i \sigma_k^A\right] \mathrm{tr}\left[(P_X)_j \sigma_k^B\right]$$

$$P(i,\, j|X,\, Y) = \sum_k p_k P(i|X,\, k) P(j|Y,\, k) \tag{4.15}$$

Where σ_i^A and σ_i^B are extremal single particle cube-states. Upon inspection of (4.15) we can clearly see that for the cube-separable state ρ we have a LHVM for the Pauli X, and Y measurements. The state ρ is separable with respect to the (tomographically complete) Pauli measurements if and only if it satisfies a LHVM for the Pauli measurements.

In summary, if there exists an operator (be it non-positive), that gives deterministic outcomes for combinations of measurement outcomes then that means that those measurements are operator compatible. This is important in the context of generalised probability theory, because if a set of measurements are not operator compatible then states cannot be uniquely specified by measurement outcomes from the set of measurements and there is the possibility of a distinction between non-locality and non-separability. This is also a point of discussion within quantum theory, as, while any separable state has a LHVM, there exist non-separable states that have a LHVM [60].

We may use a H&N type algorithm to efficiently classically simulate any dynamics preserving separability of state spaces $\mathcal{M}^*$ dual to operator compatible sets $\mathcal{M}$, if we have a LHVM for $\mathcal{M}$ and if the quantum dynamics considered cannot generate negative probabilities

Connections between quantum entanglement and cube-separability are presented in the following lemma:

Lemma 4.4.4. ***There exist quantum entangling, cube-separable operations.***

Two-qubit Bell states are cube-separable. As such, there exist quantum operations that are both cube-separable and quantum entangling.

Proof. As Bell states are cube-separable, any operation that creates a Bell-diagonal state, by say discarding a two qubit input and creating a Bell state in its place is cube-separable. □

Such operations would also be entanglement breaking (see section (9.4) in the appendix) as they would break any entanglement of the input with the other subsystems forming

the whole [61], and so may not be used to generate multiparticle entanglement without the aid of other operations. Without the ability to generate multiparticle entanglement these operations can be efficiently classical simulated using the approach in [42].

An alternative proof can be found in the appendix (9.5) and uses table notation introduced in section (4.6).

In [7] we go on to present further observations about Bloch cube entanglement. Whilst these observations are beyond the scope of this thesis, we will briefly summarise them as follows. We showed the existence of genuinely multiparticle quantum entangled states that are cube separable. The operations that could create such cube separable states could then be used to generate quantum entanglement of an increasing number of particles.

This result was then extended to show that we could construct cube separable quantum gates that are quantum entangling, non-entanglement breaking, and non-Clifford, meaning that we can classically simulate operations outside previous regimes.

4.5 Restrictions and variations

4.5.1 Restrictions for all cases

Determining the optimal $\mathcal{S}$ for problem (4.3.4) is hard as we have potentially infinitely many choices for $\mathcal{S}$. Additionally calculating $\lambda_{U_\varepsilon,\mathcal{S}}$ becomes difficult for choices of $\mathcal{S}$ with many extrema. Problem (4.3.4) is dependent on many variables, for example; the choice of noise model N, the set of measurements $\mathcal{M}$ available to the system, and the multi-partite quantum gate U requiring the most the most noise for $\mathcal{S}$-separability. As mentioned earlier, another point to consider is the action of single qubit unitaries on the $\mathcal{S}$, so operators in $\mathcal{S}$ remain in $\mathcal{S}$.

Simple symmetry arguments can be used to simplify problem (4.3.4) in the cases we have mentioned. We can split these arguments into symmetry arguments on the choices of $\mathcal{S}$, and those that can reduce the numbers of inputs needed to calculate $\lambda_{U_\varepsilon,\mathcal{S}}$.

4.5.2 Transformations on $\mathcal{S}$

In addition to restrictions from measurements, $\mathcal{S}$ is also limited by the set of single particle quantum gates available to the quantum system. As only $\mathcal{S}$-entanglement over multiple particles will be considered, $\mathcal{S}$ will be chosen in such a way that all the single particle gates be $\mathcal{S}$-separable without the need for noise. As a consequence $\mathcal{S}$ will reflect the symmetry of these operations, and only the multi-partite gates will be $\mathcal{S}$-entangling.

As already mentioned we will focus on magic state architecture based quantum devices. These devices are restricted to Pauli X, Y, Z, measurements, and are formed of the

single particle Clifford Group operations and the two particle $\mathcal{S}$-entangling control-not gate (CX). The single particle Cliffords represent octahedral symmetry transformations. Any choice of $\mathcal{S}$ satisfying $\mathcal{Q} \subseteq \mathcal{S} \subseteq \mathcal{M}^*$ must therefore also have octahedral symmetry.

4.5.3 General method and reducing the number of inputs

The multipartite CX gate is the only $\mathcal{S}$-entangling operation and thus requires noise to ensure it becomes $\mathcal{S}$-separable. For mathematical convenience we will instead consider the control-sign (CZ) gate as our multi-partite $\mathcal{S}$-entangling operation instead of the CX gate. We are free to make this choice as it has no bearing on the $\lambda_{U_\varepsilon,\mathcal{S}}$ value for each gate. We can see this in the following:

Let the noisy versions of the CZ and CX gates be denoted by CZ_ε and CX_ε respectively. The two gates are related as:

$$CX = (\mathbb{1} \otimes H) \cdot CZ \cdot (\mathbb{1} \otimes H) \tag{4.16}$$

Since $\mathcal{S}$ is chosen to have octahedral symmetry, the local Hadamard gates simply act to perform a symmetry rotations on the input and outputs of the CZ gate, and so the following holds true:

$$\lambda_{CZ_\varepsilon,\mathcal{S}} = \lambda_{CX_\varepsilon,\mathcal{S}} \tag{4.17}$$

Just like the CX gate, noise must be added to the CZ gate not only to ensure separability, but also to at very least to ensure measurement positivity. We go into further detail on this in section (9.6) in the appendix, where we show that the CZ gate acting on any state space $\mathcal{S}$ containing $\mathcal{Q}$ may return negative measurement probabilities for Pauli measurements.

For a given $\mathcal{S}$ and noise model, we calculate $\lambda_{CZ_\varepsilon,\mathcal{S}}$ by performing the following sequence of steps:

1. Select a tensor product of "pure"-extrema operators from $\mathcal{S}$.
2. Determine for our chosen input product the minimum ε value to ensure CZ_ε outputs a $\mathcal{S}$-separable product state.
3. Cycle through all other tensor product of pure states, noting ε for each input. After scanning through all product pairs set λ to be the largest ε value listed.

It is possible to simplify the calculation of $\lambda_{CZ_\varepsilon,\mathcal{S}}$ by using symmetry arguments to reduce the number of product inputs needed to be scanned over. Here will be detailed the two main symmetries used in the thesis to reduce to calculation.

Suppose for some particular state-space $\mathcal{S}$ and noise model N, we determine ε to be the minimum noise strength to ensure $\mathcal{S}$-separable CZ output for the $\mathcal{S}$-separable input

$\rho = \alpha \otimes \beta$. We can express the mapping from ρ to the output state as follows:

$$\rho = \alpha \otimes \beta \to N_\varepsilon \circ CZ(\alpha \otimes \beta) \in (\mathcal{S}-\text{separable}) \tag{4.18}$$

Let F denote some two qubit operation acting on the output of the noisy CZ gate. If F corresponds to a Swap gate it will keep the output $\mathcal{S}$-separable, and as it commutes with the CZ operation it allows us to pass it through the CZ operation to act directly on the input state. If F also commutes with N then eq. (4.18) becomes:

$$F \circ N_\varepsilon \circ CZ(\alpha \otimes \beta) \to N_\varepsilon \circ CZ(\beta \otimes \alpha) \in (\mathcal{S}-\text{separable}) \tag{4.19}$$

By swapping the subsystems in the input we still require the ε amount of noise to ensure $\mathcal{S}$-separable CZ output. There is therefore no need to consider the swapped product inputs and we can halve the number of input products we need to consider.

Similarly if F is the tensor product of single particle Clifford operations that act by conjugation, then we can pass it through the CZ gate to get another tensor product of single particle Clifford operations G acting on the input state. If our noise model N is such that it commutes with F then the action of F on the noisy CZ output state is

$$F\left(N_\varepsilon \circ CZ(\alpha \otimes \beta)\right) F^\dagger \to N_\varepsilon \circ CZ(G\left(\alpha \otimes \beta\right) G^\dagger) \tag{4.20}$$

The action of the single particle Clifford operations on operators in $\mathcal{S}$ correspond to octahedral symmetry transformations. By choosing appropriate sequences of G, it is possible to cycle the extrema in one octant of $\mathcal{S}$ through the analogous extrema in the other octants. If ε is large enough to ensure $\mathcal{S}$-separable CZ outputs for all tensor products of states in one octant of $\mathcal{S}$, then ε is large enough for product inputs in $\mathcal{S} \otimes \mathcal{S}$.

4.5.4 Variants of problem (4.3.4) considered in this thesis

In the upcoming chapters we will determine $\lambda_{CZ_\varepsilon,\mathcal{S}}$ for a variety of $\mathcal{S}$ and a variety of choices of noise model. In this section we will briefly outline what these variations are.

We will start by considering $\mathcal{S}$ to only be restricted by imperfect preparation and measurement of the states. In this case $\mathcal{S}$ is chosen to be a Bloch sphere that is contracted (corresponding to depolarised preparation) or expanded (corresponding to depolarised measurements) based on the level of noise on the preparation or measurement respectively. Three noise models are considered on the CZ operation; local depolarising noise, joint depolarising noise, and local dephasing noise.

For all of the other variations we will look at state spaces $\mathcal{S}$ also bound by allowed measurement restrictions, initially taking $\mathcal{S}$ to be the full dual $\mathcal{M}$ i.e. the Bloch cube. As already detailed in section (4.4), this choice is conceptually of interest as it can be shown that certain quantum entangled states are Bloch cube-separable. In Chapter (5)

we will go into more technical detail on how we calculate $\lambda_{CZ_\varepsilon,\mathcal{S}}$ for the Bloch cube specifically. We will allow for imperfect Pauli measurement and noisy preparation, and once again consider the three noise models; local depolarising noise, joint depolarising noise, and local dephasing noise.

As stated previously it is not necessary that we include the full dual space, and we can choose sets $\mathcal{S}$ satisfying $\mathcal{Q} \subset \mathcal{S} \subset \mathcal{M}^*$. In Chapter (7) we restrict ourselves to only joint depolarising noise, and attempt to identify the optimal choice of $\mathcal{S}$ that satisfies the restrictions in subsection (4.5.1) requiring the least $\lambda_{CZ_\varepsilon,\mathcal{S}}$. As there are seemingly infinite choices for $\mathcal{S}$ we show that by characterising the sets by various norms (factoring in imperfect measurement and preparation), we are able to to arrive at necessary bounds on the noise needed by the CZ gate for $\mathcal{S}$-separability that ensure Born rule measurement positivity. The noise required for Bloch cube separability forms an upper bound, specifying a region of noise rates for an optimal $\mathcal{S}$.

In the final research Chapter (6) we will choose $\mathcal{S}$ to be a truncated Bloch cube, the $\mathcal{S}$ requiring the least amount of joint depolarising noise that we found. We will go through a detailed calculation and numerical results for the amount of joint depolarising noise required to ensure CZ gate $\mathcal{S}$-separability. Finally arriving at a symbolic expression for the exact noise noise value for separability in the specific case of perfect measurements and preparations.

4.6 Pauli table notation and asides

Throughout this thesis we will make use of *Pauli table* notation to conveniently represent two particle operators. This notation becomes especially useful when considering the effects of Clifford operations on two particle operators. As the Pauli operators form an operator-basis for the Hilbert spaces of the individual particles, the set of tensor products of the Pauli operators form an operator basis for two particle systems. Thus any two particle operator $\mathcal{A}'$ can be written as:

$$\mathcal{A}' = \sum_{ij} \mathcal{A}_{ij} \sigma_i \otimes \sigma_j \tag{4.21}$$

where we may denote $\sigma_0 = \mathbb{1}$, $\sigma_1 = X$, $\sigma_2 = Y$, $\sigma_3 = Z$. These operators can be represented in a 4×4 table (denoted by curly brackets) whose elements correspond to the coefficients for the various Pauli tensor product combinations:

$$\left\{ \begin{array}{cccc} \mathcal{A}_{00} & \mathcal{A}_{01} & \mathcal{A}_{02} & \mathcal{A}_{03} \\ \mathcal{A}_{10} & \mathcal{A}_{11} & \mathcal{A}_{12} & \mathcal{A}_{13} \\ \mathcal{A}_{20} & \mathcal{A}_{21} & \mathcal{A}_{22} & \mathcal{A}_{23} \\ \mathcal{A}_{30} & \mathcal{A}_{31} & \mathcal{A}_{32} & \mathcal{A}_{33} \end{array} \right\} \tag{4.22}$$

These elements correspond to the coefficients of the following Pauli tensor product terms:

$$\left\{\begin{array}{cccc} \mathbb{1}\otimes\mathbb{1} & \mathbb{1}\otimes X & \mathbb{1}\otimes Y & \mathbb{1}\otimes Z \\ X\otimes\mathbb{1} & X\otimes X & X\otimes Y & X\otimes Z \\ Y\otimes\mathbb{1} & Y\otimes X & Y\otimes Y & Y\otimes Z \\ Z\otimes\mathbb{1} & Z\otimes X & Z\otimes Y & Z\otimes Z \end{array}\right\} \tag{4.23}$$

I.e. the entries in the table corresponding to the Pauli tensor $X\otimes Y$, correspond to the coefficients $\mathcal{A}_{12}$ in the decomposition of the operator $\mathcal{A}'$. Two particle operators of the form $\varrho_A\otimes\varrho_B$ where ϱ_A, $\varrho_B\in\mathcal{S}$ with Bloch vectors (x, y, z), and (A, B, C) respectively have the following Pauli table representation:

$$\frac{1}{4}\left\{\begin{array}{cccc} 1 & A & B & C \\ x & xA & xB & xC \\ y & yA & yB & yC \\ z & zA & zB & zC \end{array}\right\} \tag{4.24}$$

Where the "$\frac{1}{4}$" in front of the Pauli table corresponds to a normalisation factor that would multiply each of the coefficients. The CZ gate has the following effect on arbitrary two particle states $\varrho_A\otimes\varrho_B$:

$$\frac{1}{4}\left\{\begin{array}{cccc} 1 & A & B & C \\ x & xA & xB & xC \\ y & yA & yB & yC \\ z & zA & zB & zC \end{array}\right\} \rightarrow \frac{1}{4}\left\{\begin{array}{cccc} 1 & zA & zB & C \\ xC & yB & -yA & x \\ yC & -xB & xA & y \\ z & A & B & zC \end{array}\right\} \tag{4.25}$$

4.6.1 The effect of noise on two particle product states

The action of noisy CZ gates on product pure operators can be represented as:

$$\varrho_A\otimes\varrho_B \rightarrow CZ_\varepsilon(\varrho_A\otimes\varrho_B) = N_\varepsilon\circ CZ(\varrho_A\otimes\varrho_B) \tag{4.26}$$

where ϱ_A, ϱ_B are arbitrary operators from $\mathcal{S}$. This sequence has the following effect on the Pauli tables of input states for the three noise models introduced in section (2.2.1):

Joint depolarising noise

$$\frac{1}{4}\left\{\begin{array}{cccc} 1 & A & B & C \\ x & xA & xB & xC \\ y & yA & yB & yC \\ z & zA & zB & zC \end{array}\right\} \rightarrow \frac{(1-\varepsilon)}{4}\left\{\begin{array}{cccc} \frac{1}{(1-\varepsilon)} & zA & zB & C \\ xC & yB & -yA & x \\ yC & -xB & xA & y \\ z & A & B & zC \end{array}\right\} \tag{4.27}$$

Local depolarising noise

$$\frac{1}{4}\begin{Bmatrix} 1 & A & B & C \\ x & xA & xB & xC \\ y & yA & yB & yC \\ z & zA & zB & zC \end{Bmatrix} \rightarrow \frac{(1-\varepsilon)^2}{4}\begin{Bmatrix} \frac{1}{(1-\varepsilon)^2} & \frac{zA}{(1-\varepsilon)} & \frac{zB}{(1-\varepsilon)} & \frac{C}{(1-\varepsilon)} \\ \frac{xC}{(1-\varepsilon)} & yB & -yA & x \\ \frac{yC}{(1-\varepsilon)} & -xB & xA & y \\ \frac{z}{(1-\varepsilon)} & A & B & zC \end{Bmatrix} \tag{4.28}$$

Local dephasing noise

Where $\varepsilon' = 1 - 2\varepsilon$:

$$\frac{1}{4}\begin{Bmatrix} 1 & A & B & C \\ x & xA & xB & xC \\ y & yA & yB & yC \\ z & zA & zB & zC \end{Bmatrix} \rightarrow \frac{1+\varepsilon'}{4}\begin{Bmatrix} 1 & zA\varepsilon' & zB\varepsilon' & C \\ xC\varepsilon' & yB\varepsilon'^2 & -yA\varepsilon'^2 & x\varepsilon' \\ yC\varepsilon' & -xB\varepsilon'^2 & xA\varepsilon'^2 & y\varepsilon' \\ z & A\varepsilon' & B\varepsilon' & zC \end{Bmatrix} \tag{4.29}$$

4.6.2 Pauli measurements

It is straightforward to use Pauli table notation to calculate measurement probabilities, especially if these measurements correspond to local Pauli X, Y, or Z measurements. We will begin by outlining how to perform the trace of a product of operators represented by Pauli tables as this would be necessary when using the Born rule. Suppose we have two particle operators ρ and σ with Pauli tables:

$$\frac{1}{4}\begin{Bmatrix} \rho_{00}=1 & \rho_{01} & \rho_{02} & \rho_{03} \\ \rho_{10} & \rho_{11} & \rho_{12} & \rho_{13} \\ \rho_{20} & \rho_{21} & \rho_{22} & \rho_{23} \\ \rho_{30} & \rho_{31} & \rho_{32} & \rho_{33} \end{Bmatrix} \qquad \frac{1}{4}\begin{Bmatrix} \sigma_{00}=1 & \sigma_{01} & \sigma_{02} & \sigma_{03} \\ \sigma_{10} & \sigma_{11} & \sigma_{12} & \sigma_{13} \\ \sigma_{20} & \sigma_{21} & \sigma_{22} & \sigma_{23} \\ \sigma_{30} & \sigma_{31} & \sigma_{32} & \sigma_{33} \end{Bmatrix} \tag{4.30}$$

As each Pauli table represents a summation of Pauli operator tensor products and coefficients the product of σ and ρ gives a summation of Pauli operator tensor products and products of these various coefficients with some additional signs and imaginary terms gained from the Pauli products, i.e.

$$\begin{aligned} \sigma\rho &= \frac{1}{16}\left(\rho + (\sigma_{01}\rho)\mathbb{1}\otimes X + (\sigma_{02}\rho)\mathbb{1}\otimes Y + \ldots\right) \\ &= \frac{1}{16}((\sum_{ij}\sigma_{ij}\rho_{ij})\mathbb{1}\otimes\mathbb{1} + \rho_{01}\mathbb{1}\otimes X + \rho_{02}\mathbb{1}\otimes Y + \ldots) \end{aligned} \tag{4.31}$$

The only terms to come out of $\mathrm{tr}\left[\sigma\rho\right]$ are those coefficients of the $\mathbb{1}\otimes\mathbb{1}$ term as the Pauli operators are traceless:

$$\mathrm{tr}\left[\sigma\rho\right] = \frac{1}{4}\sum_{ij}\sigma_{ij}\rho_{ij} \tag{4.32}$$

We can now look at an example of using eq. (4.32) to calculate measurement probabilities using the Born rule:

Example 4.6.1. ***Pauli tables and Pauli measurements.***

Suppose a system's state ρ is represented by the Pauli table given in eq. (4.30), for the Pauli measurements X on system A, and Y on system B, the probability of getting the results $(-1)^n$ and $(-1)^m$ is given through the Born rule as:

$$\text{Prob}\left((-1)^n, (-1)^m | X, Y\right) = \text{tr}\left[\left(\frac{1}{2}\left(\mathbb{1} + (-1)^n X\right) \otimes \frac{1}{2}\left(\mathbb{1} + (-1)^m Y\right)\right)\rho\right] \quad (4.33)$$

The only terms from ρ that contribute to the trace are those corresponding to the Pauli products $X \otimes \mathbb{1}$, $\mathbb{1} \otimes Y$, and $X \otimes Y$. From eq. (4.32) we can see that by looking up these coefficients in the Pauli table representing ρ eq. (4.33) becomes:

$$= \frac{1}{4}\left(\mathcal{A}_{00} + (-1)^n \mathcal{A}_{10} + (-1)^m \mathcal{A}_{02} + (-1)^{n+m} \mathcal{A}_{12}\right) \quad (4.34)$$

4.7 Summary of Chapter (4)

We started this chapter by defining the single particle state space $\mathcal{S}$, detailing how restrictions on the available measurements bound the space of trace 1 operators to those that return positive measurement probabilities with respect to those measurements via the Born rule. In particular we define the Bloch cube state space - the cube of operators associated with Pauli X, Y, Z. measurements. Following on from the definition of $\mathcal{S}$, we defined a new notion of entanglement with respect to $\mathcal{S}$. We modified the H&N algorithm be replacing the algorithm's Bloch sphere single particle state space, with our larger $\mathcal{S}$ state space. If the operations of the noisy quantum computer only could generate $\mathcal{S}$-separable states then H&N's algorithm would translate and we would be able to classically simulate the noisy quantum computer.

In the next section we focused on some observations about the relationship between Bloch cube-separability, quantum-entanglement, and non-locality. We observed that we could construct operations that were cube-separable, quantum-entangling, and non-Clifford, suggesting that a H&N-like algorithm based around Bloch cubes could classically simulate regimes not covered by straightforward application of previous schemes.

We then moved onto detailing the exact technical problem of the thesis, that being, we are looking to find the $\mathcal{S}$ contained in a Bloch cube, but containing the Bloch sphere, giving the lowest value of $\lambda_{CZ_\varepsilon,\mathcal{S}}$. We detailed symmetries reducing the complexity of the problem, and stated the variations of the problem we will be looking at. The chapter concluded by introducing Pauli table notation, and showing how the three noise models we considered transform CZ output states. We finally described how to calculate Pauli measurements using the Pauli table notation.

5 The rescaled Bloch sphere and the Bloch cube

5.1 Introduction

This chapter will present the results obtained in [7].

Non-ideal measurements correspond to another type of restriction on measurements, allowing us to choose state-spaces $\mathcal{S}'$ for the algorithm that are enlargened/shrunken (termed here "rescaled") versions of state-spaces $\mathcal{S}$ corresponding to ideal measurements sets.

We will start by showing how non-ideal measurements and non-ideal preparations of states allow us to specify a new single particle state space. Then we will outline the method used in the calculations of $\lambda(CZ_\varepsilon, \mathcal{S})$ for such state spaces, before stating the symmetry arguments used to aid in these calculations.

We will next consider a straightforward extension of the H&N algorithm to the case where $\mathcal{S}$ is taken to be the Bloch sphere modified to allow for non-ideal preparation/measurements. We observe the connections between a rescaling parameter specifying "non-ideal"-ness and values of $\lambda_{CZ_\varepsilon,\mathcal{S}}$. This investigation was motivated by the hope that noise on measurements translates into a reduction in $\lambda_{CZ_\varepsilon,\mathcal{S}}$ over the H&N algorithm.

In the second half of this chapter we take $\mathcal{S}$ to be the full dual $\mathcal{M}^*$ for Pauli measurements giving us the Bloch cube state space. We first calculate $\lambda_{CZ_\varepsilon,\mathcal{S}}$ in the ideal case where the there is no noise on preparation/measurement and then again with noisy preparation/measurement allowing us to rescale the Bloch cube state space. The motivation for both of these cases was that again the geometry of the state space and/or rescaling would allow in a reduction of $\lambda_{CZ_\varepsilon,\mathcal{S}}$.

5.2 Non-ideal measurements

Non-ideal measurements can be imagined operationally as a noisy channel, followed by an ideal measurement. Here we will consider our particles to be subject to depolarising noise with probability p immediately prior undergoing a projective measurement associated to projectors Π. Where our particle is initially prepared in state

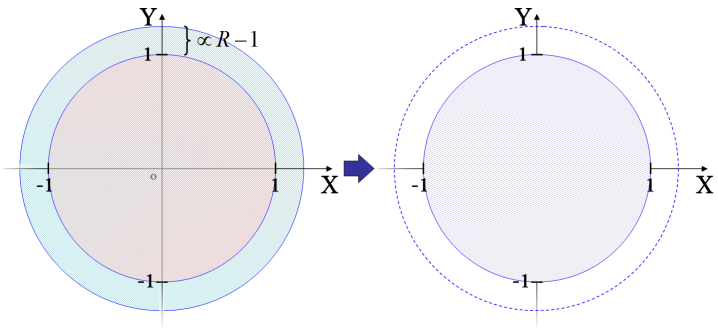


Figure 5.1: Cross-section on the $X-Y$ plane showing the action of the depolarising noise at rate p prior to measurement on the rescaled Bloch sphere, with radius R coloured dark blue. The lighter blue region inside shows a cross-section of the usual Bloch sphere. The states depicted on the left image are for the noise rate p are taken back to the Bloch sphere.

$\rho = \frac{1}{2}(\mathbb{1} + \underline{u}\cdot\underline{\sigma}) \in \mathcal{M}^*$, the effect of the noise is as follows:

$$\rho \to \rho' = (1-p)\rho + \frac{p}{2}\mathbb{1} = \frac{1}{2}(\mathbb{1} + (1-p)\underline{u}\cdot\underline{\sigma}) \tag{5.1}$$

As eq. (5.1) illustrates, the effect of the depolarising noise is to transform the Bloch vector $\underline{u}$, into $(1-p)\underline{u}$ shrinking any input operator's Bloch vector for any value $p > 0$. This allows us to accept input density operators from a larger set containing operators outside $\mathcal{M}^*$- it is simply required that these operators ensure that $\mathrm{tr}\,[\Pi\rho'] \geq 0$. Therefore for a system containing imperfect measurements the input density operator representing each particle will be taken from the set $\tilde{\mathcal{M}}^*$ containing operators $\tilde{\rho}$ of the form:

$$\tilde{\rho} = \frac{1}{2}\{\mathbb{1} + R\underline{u}\cdot\underline{\sigma}\} \qquad \text{where } R = \frac{1}{1-p} \tag{5.2}$$

For $R > 1$, $\tilde{\mathcal{M}}^*$ simply is an enlargement of $\mathcal{M}^*$, and where $R = 1$, there is no-rescaling and $\tilde{\mathcal{M}}^* = \mathcal{M}^*$. Figure (5.1) shows a cross-section of the admissible operator set for a noise at rate p prior to measurement in the case where $\mathcal{M}^* = \mathcal{Q}$ the Bloch sphere.

The parameter R can also be used to describe noise in the preparation of the input qubits where $0 < R < 1$, i.e. we can prepare particles and then individually depolarising them at a rate of $1-R$:

$$\rho \to \rho' = R\rho + \frac{(1-R)}{2}\mathbb{1} \tag{5.3}$$

In the rescaled sphere case we use the quantum PPT criterion to determine separability of the output states. In order to do this however, we need to perform an inverse map taking the output states back to quantum Bloch sphere state space as the PPT criterion is only usable with quantum Bloch spheres.

Let T_R denote maps taking operators from $\mathcal{M}^*$ to the rescaled set $\tilde{\mathcal{M}}^*$, and its inverse

as T_R^{-1}. The action of $T_R \otimes T_R(\cdot)$ represents faulty preparation of states or faulty measurements, rescaling the input states prior to the CZ_ε operation. The product inverse map $T_R^{-1} \otimes T_R^{-1}(\cdot)$ undoes any rescaling returning states back to the fault-less size (although still possibly $\mathcal{M}^*$-entangled from the CZ_ε operation).

Operationally the entire process of rescaling the input to the measurement can be viewed as the following sequence of operations acting on $\rho^A \otimes \rho^B \in \mathcal{M}^{*\otimes 2}$:

$$T_R^{-1} \otimes T_R^{-1} \circ CZ_\varepsilon \circ T_R \otimes T_R(\rho^A \otimes \rho^B) \tag{5.4}$$

Once again it's important to note that the application of specific local unitaries after the final $T_R^{-1} \otimes T_R^{-1}(\cdot)$ does not affect the separability of the output operators as they commute with $T_R^{-1} \otimes T_R^{-1}$ and would simply act to rotate any tensor product state. The T_R map has the following effect on the Pauli table

$$\begin{Bmatrix} \mathcal{A}_{00} & \mathcal{A}_{01} & \mathcal{A}_{02} & \mathcal{A}_{03} \\ \mathcal{A}_{10} & \mathcal{A}_{11} & \mathcal{A}_{12} & \mathcal{A}_{13} \\ \mathcal{A}_{20} & \mathcal{A}_{21} & \mathcal{A}_{22} & \mathcal{A}_{23} \\ \mathcal{A}_{30} & \mathcal{A}_{31} & \mathcal{A}_{32} & \mathcal{A}_{33} \end{Bmatrix} \rightarrow \begin{Bmatrix} \mathcal{A}_{00} & R\mathcal{A}_{01} & R\mathcal{A}_{02} & R\mathcal{A}_{03} \\ R\mathcal{A}_{10} & R^2\mathcal{A}_{11} & R^2\mathcal{A}_{12} & R^2\mathcal{A}_{13} \\ R\mathcal{A}_{20} & R^2\mathcal{A}_{21} & R^2\mathcal{A}_{22} & R^2\mathcal{A}_{23} \\ R\mathcal{A}_{30} & R^2\mathcal{A}_{31} & R^2\mathcal{A}_{32} & R^2\mathcal{A}_{33} \end{Bmatrix} \tag{5.5}$$

And similarly the inverse T_R^{-1}:

$$\begin{Bmatrix} \mathcal{A}_{00} & \mathcal{A}_{01} & \mathcal{A}_{02} & \mathcal{A}_{03} \\ \mathcal{A}_{10} & \mathcal{A}_{11} & \mathcal{A}_{12} & \mathcal{A}_{13} \\ \mathcal{A}_{20} & \mathcal{A}_{21} & \mathcal{A}_{22} & \mathcal{A}_{23} \\ \mathcal{A}_{30} & \mathcal{A}_{31} & \mathcal{A}_{32} & \mathcal{A}_{33} \end{Bmatrix} \rightarrow \begin{Bmatrix} \mathcal{A}_{00} & \frac{1}{R}\mathcal{A}_{01} & \frac{1}{R}\mathcal{A}_{02} & \frac{1}{R}\mathcal{A}_{03} \\ \frac{1}{R}\mathcal{A}_{10} & \frac{1}{R^2}\mathcal{A}_{11} & \frac{1}{R^2}\mathcal{A}_{12} & \frac{1}{R^2}\mathcal{A}_{13} \\ \frac{1}{R}\mathcal{A}_{20} & \frac{1}{R^2}\mathcal{A}_{21} & \frac{1}{R^2}\mathcal{A}_{22} & \frac{1}{R^2}\mathcal{A}_{23} \\ \frac{1}{R}\mathcal{A}_{30} & \frac{1}{R^2}\mathcal{A}_{31} & \frac{1}{R^2}\mathcal{A}_{32} & \frac{1}{R^2}\mathcal{A}_{33} \end{Bmatrix} \tag{5.6}$$

5.3 The rescaled Bloch sphere

5.3.1 Method and symmetry arguments

Within this section we will explore the effects on $\lambda_{CZ_\varepsilon,\mathcal{S}}$ of taking $\mathcal{S}$ to be a rescaled Bloch sphere by allowing for non-ideal measurements or preparation. For a particular noise model and rescaling parameter R, in order to determine whether the noisy CZ is separable with respect to the rescaled state-space, we must show that all outputs of the inverse scaling operation $T_R^{-1} \otimes T_R^{-1}(\cdot)$ are Bloch sphere separable. We require that the output states are Bloch sphere separable as we will use the PPT criterion [62, 63] to determine separability, as PPT outputs from the $T_R^{-1} \otimes T_R^{-1}(\cdot)$ correspond to $\mathcal{S}$-separable outputs from the noisy CZ gate. We use the PPT criterion here as it is the most straightforward way to determine separability of the state spaces in question.

As described in section (4.5.3), symmetry arguments can be used to simplify the calculation of $\lambda_{CZ_\varepsilon,\mathcal{S}}$ for a given noise model and $\mathcal{S}$ for a particular R parameter. We choose our CZ noise models to be local depolarising noise, joint depolarising noise, and local

dephasing noise. It is important to note that the application of local unitaries after the final $T_R^{-1} \otimes T_R^{-1}(\cdot)$ do not affect whether the output state is positive or PPT, as they commute with $T_R^{-1} \otimes T_R^{-1}$ and would simply act to rotate any tensor product state to another product state.

The two symmetry arguments from section (4.5.3) are used here to reduce the number of input product states. We may further reduce the number of input states by using symmetries particular to the Bloch sphere geometry. Suppose we applied local Pauli Z axis rotations on each output particle from the $T_R^{-1} \otimes T_R^{-1}(\cdot)$ operation, these unitaries can be commuted through to act upon the input product states rotating them around the Z axis. As these rotations are changes to the input states with no impact on the amount of noise needed for separability we are thus free to chose the input states $\rho = \alpha \otimes \beta$ up to arbitrary rotations about the Pauli Z, and will choose to only input product pure quantum states with no Pauli Y components i.e. with Bloch vectors ($cos(\vartheta)$, 0, $sin(\vartheta)$) and ($cos(\theta)$, 0, $sin(\theta)$) for the states α and β respectively. We may once again use the table notation from section (4.6) to describe the input states. The rescaled input state $T_R \otimes T_R(\alpha \otimes \beta)$ is proportional to a state with a table representation:

$$\left\{ \begin{array}{cccc} 1 & Rcos(\vartheta) & 0 & Rsin(\vartheta) \\ Rcos(\theta) & R^2cos(\theta)cos(\vartheta) & 0 & R^2cos(\theta)sin(\vartheta) \\ 0 & 0 & 0 & 0 \\ Rsin(\theta) & R^2sin(\theta)cos(\vartheta) & 0 & R^2sin(\theta)sin(\vartheta) \end{array} \right\} \tag{5.7}$$

Under the action of the CZ operation the transformed table is proportional to:

$$\left\{ \begin{array}{cccc} 1 & R^2sin(\theta)cos(\vartheta) & 0 & Rsin(\vartheta) \\ R^2cos(\theta)sin(\vartheta) & 0 & 0 & Rcos(\theta) \\ 0 & 0 & R^2cos(\theta)cos(\vartheta) & 0 \\ Rsin(\theta) & Rcos(\vartheta) & 0 & R^2sin(\theta)sin(\vartheta) \end{array} \right\}$$

Under the effect of the three noise models shown in section (4.6.1), the coefficients of the output state of the CZ operation gain "factors" $0 \leq r \leq 1$ which may vary for each Pauli term. The transformed table is therefore proportional to:

$$\left\{ \begin{array}{cccc} 1 & rR^2sin(\theta)cos(\vartheta) & 0 & rRsin(\vartheta) \\ rR^2cos(\theta)sin(\vartheta) & 0 & 0 & rRcos(\theta) \\ 0 & 0 & rR^2cos(\theta)cos(\vartheta) & 0 \\ rRsin(\theta) & rRcos(\vartheta) & 0 & rR^2sin(\theta)sin(\vartheta) \end{array} \right\} \tag{5.8}$$

Finally the action of the inverse rescaling map removes some of the R parameters, with partial transposition on the second system we are left with a table representation proportional to:

$$\left\{\begin{array}{cccc} 1 & rRsin(\theta)cos(\vartheta) & 0 & rsin(\vartheta) \\ rRcos(\theta)sin(\vartheta) & 0 & 0 & \frac{r}{R}cos(\theta) \\ 0 & 0 & \pm rcos(\theta)cos(\vartheta) & 0 \\ rsin(\theta) & \frac{r}{R}cos(\vartheta) & 0 & rsin(\theta)sin(\vartheta) \end{array}\right\} \tag{5.9}$$

We aim to show that all states outputted by the sequence of operations mentioned remain positive for both choices of sign in eq. (5.9). Symmetry arguments can reduce the number of input states that need to be considered: adding π to ϑ, flips the sign of $sin(\vartheta)$ and $cos(\vartheta)$, this is equivalent to applying the local unitary operation $Y \otimes Z$ to the final state represented by eq. (5.4). Similarly adding π to θ, changes the sign of $sin(\theta)$ and $cos(\theta)$ and is equivalent to the local unitary operation $Z \otimes Y$. As local unitary operations, neither $Y \otimes Z$ nor $Z \otimes Y$ have any effect on the final states separability that is whether it satisfies the PPT criterion, we can therefore make the restriction θ, $\vartheta \in [0, \pi]$. Applying the unitary $Z \otimes \mathbb{1}$ to the final state is equivalent to the transformation $\theta - \frac{\pi}{2} \to \frac{\pi}{2} - \theta$, and similarly the operation $\mathbb{1} \otimes Z$ to $\vartheta - \frac{\pi}{2} \to \frac{\pi}{2} - \vartheta$, hence we may, without loss of generality, restrict θ, $\vartheta \in [0, \frac{\pi}{2}]$.

With these restrictions to the input states in mind numerical calculations were performed to calculate the noise rates for CZ separability for varying R for each CZ noise model mentioned. These numerics followed the methodology in laid out in section (4.5.3) and (5.2), determining if, for a given input state, R value and noise rate, the output of the $T_R^{-1} \otimes T_R^{-1}(\cdot)$ operation satisfies the PPT criterion. The R and noise rate were separately incrementally increased and the whole process was repeated for all inputs. The input state requiring the most noise to satisfy the PPT criterion was identified and that noise rate was used as the noise threshold for that particular value of R.

From the numerics, for joint and local depolarising noise, we saw that for $0 < R < 1$, the product state with θ, $\vartheta = 0$ (corresponding to the product of two Pauli X eigenstates) required the most noise to become positive and PPT, for $R > 1$ the states $\vartheta = 0$, $\theta = \frac{\pi}{2}$, and $\vartheta = \frac{\pi}{2}$, $\theta = 0$, (corresponding a Pauli X eigenstate tensor a Pauli Z eigenstate) required the most noise on the CZ operation. It is important to note that we were not able to confirm these results analytically.

The local dephasing case behaved differently from the other two noise models, and will be described in more detail in the next section.

5.4 Rescaled Bloch sphere-separability noise threshold results

In this section we will present the results obtained from the numerical calculations. These results are presented in two forms; firstly as a relationship between the CZ gate noise threshold and the rescaling parameter R, and secondly, between the noise

threshold and the depolarisation noise on preparation/measurement. The connection between R and the depolarisation noise on preparation/measurement is given in equations (5.2) and (5.3).

5.4.1 Joint depolarising noise for rescaled Bloch spheres

Figure (5.2) contains a plot obtained from numerics showing the amount of joint depolarising noise needed to ensure CZ operations output separable rescaled Bloch sphere states for the rescaling parameter R. We can see from figure (5.2) the minimal value occurs for the rescaling parameter $R \approx 1.73$, corresponding to a local depolarisation rate of approximately 42.2% immediately prior to measurements. At this rescaling we require a joint depolarisation rate ε of approximately 53.6% on the CZ operation. This noise rate on the CZ is less than if we had considered no rescaling of the input state i.e. where $R = 1$. Hence by considering some noise on the measurement, we can reduce the amount of noise to make the CZ operation $\mathcal{S}$-separable.

5.4.2 Local depolarising noise for rescaled Bloch spheres

Figure (5.3) contains a plot obtained from the numerics showing the amount of local depolarising noise needed to ensure CZ operations output separable rescaled Bloch sphere states for the rescaling parameter R. We can see from figure(5.3) where we have the rescaling parameter $R \approx 1.16$, corresponding to a local depolarisation rate of approximately 13.8%, immediately prior to measurements, we require a local depolarisation rate ε of approximately 39.5% on the CZ operation. This noise rate on the CZ is less than if we had considered no rescaling of the input state. i.e. where $R = 1$. Again by considering some noise on the measurement, we can reduce the amount of noise to make the CZ operation $\mathcal{S}$-separable.

5.4.3 Local dephasing noise for rescaled Bloch spheres

For the local dephasing noise case our results were unlike those for depolarising noise in that we only were able to achieve separability of one output state, and only when either there was no rescaling, or there was total dephasing noise on the CZ output. We show this result in more detail as follows.

Suppose we chose the states $\alpha \otimes \beta$ to be the input states $\frac{1}{2}(\mathbb{1} + X) \otimes \frac{1}{2}(\mathbb{1} + Z)$. The sequence of operations expressed in eq. (5.4) produces a state proportional to the following output state for our choice of $\alpha \otimes \beta$, where we choose the noise model for the CZ operation to be a local dephasing operation:

$$\mathbb{1} \otimes \mathbb{1} + (1 - 2\varepsilon)RX \otimes \mathbb{1} + \mathbb{1} \otimes Z + \frac{(1 - 2\varepsilon)}{R} X \otimes Z \tag{5.10}$$

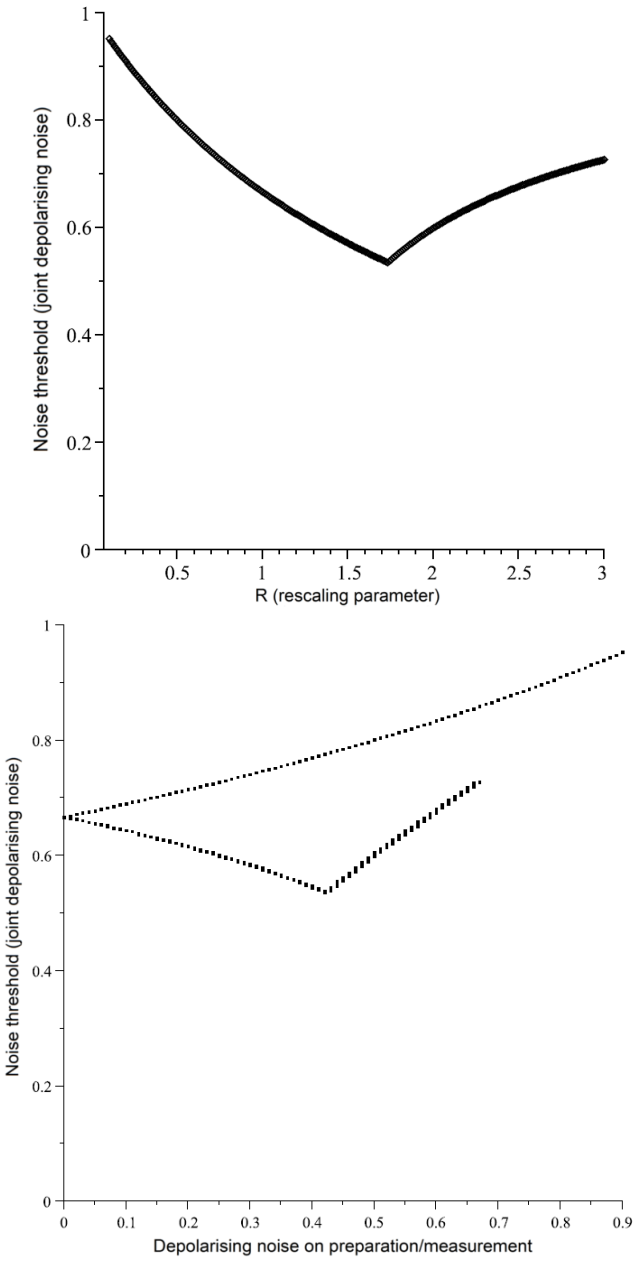


Figure 5.2: Plots of joint depolarising noise verses rescaling parameter R, and equivalent depolarisation noise on measurement/preparation.

(Top) Joint depolarising noise verses rescaling parameter R. By adding a slight amount of noise to measurements ($R > 1$) we can achieve CZ output separability with less joint depolarising noise acting on the CZ than for the no measurement noise case ($R = 1$).

(Bottom) Joint depolarising noise verses equivalent depolarisation rate on preparation ($R < 1$) for the higher curve, and noise on measurement ($R > 1$) for the lower curve. The vertical axis labels noise required for quantum-separable outputs from the CZ operation. Images taken from [7].

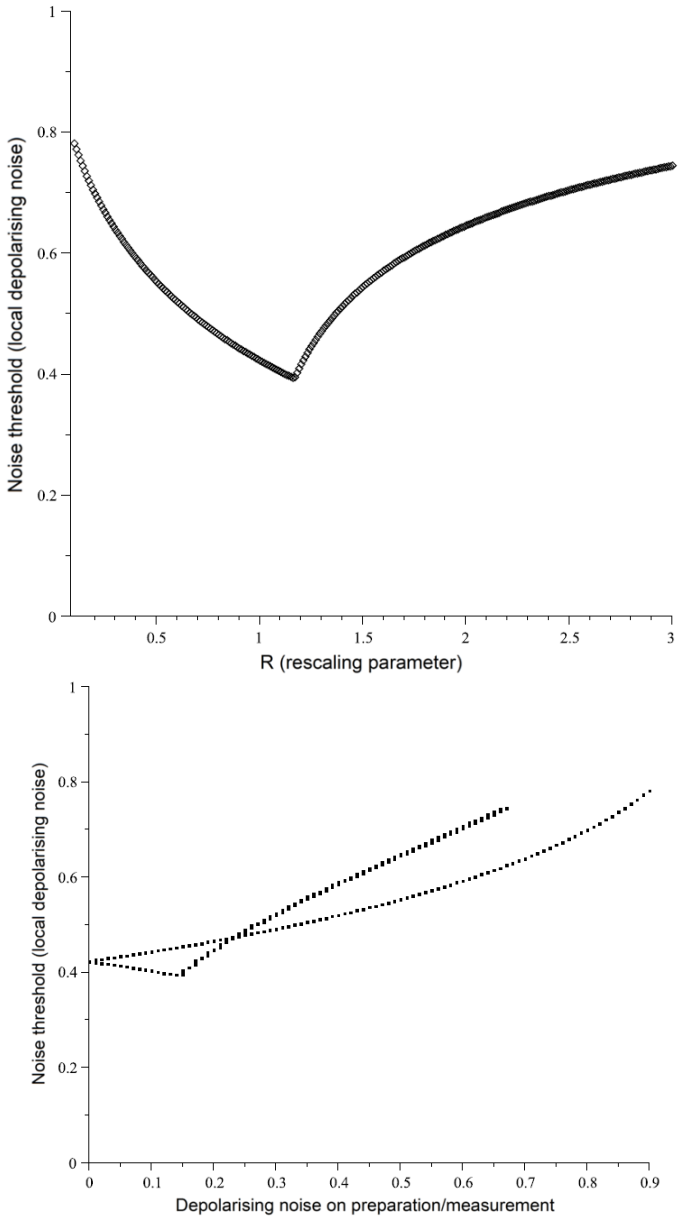


Figure 5.3: Plots of local depolarising verses rescaling parameter R, and equivalent depolarisation noise on measurement/preparation.

(Top) Local depolarising noise verses rescaling parameter R. By adding a slight amount of noise to measurements ($R > 1$) we can achieve CZ output separability with less local depolarising noise than for the no measurement noise case ($R = 1$).

(Bottom) Local depolarising noise verses equivalent depolarisation rate $1-R$ on preparation ($R < 1$) for the curve that starts higher, and measurement ($R > 1$) for the curve that starts lower. The vertical axis labels noise required for quantum-separable CZ outputs. Images taken from [7].

Partial transposition has no effect on the form of eq. (5.10), so we must simply prove that eq. (5.10) is positive in order to prove that the CZ operation is separable with respect to the rescaled Bloch sphere state space. Two of the eigenvalues of eq. (5.10) are:

$$(1-2\varepsilon)\left(R-\frac{1}{R}\right), \quad (1-2\varepsilon)\left(\frac{1}{R}-R\right) \tag{5.11}$$

As one eigenvalue is the negative of the other, they can only be positive if they are both zero, which requires that $R=1$ or $\varepsilon=\frac{1}{2}$. In all other cases the output state in eq. (5.10) is not positive and PPT meaning the CZ operation is not $\mathcal{S}$-separable.

5.5 Cube-separability noise threshold results

This half of the chapter will look at the effect on the calculation of $\lambda_{CZ_\varepsilon,\mathcal{S}}$ from taking $\mathcal{S}$ to be the full dual $\mathcal{M}^*$ - the Bloch cube for systems restricted to Pauli measurements and preparations. We will first calculate $\lambda_{CZ_\varepsilon,\mathcal{S}}$ without faulty measurements, and then also factor in non-ideal preparation/measurement into the calculation of $\lambda_{CZ_\varepsilon,\mathcal{S}}$ for Bloch cubes at the end. We will again consider the three types of natural noise from section (4.6.1) for both cases.

In [7] we construct an operation, (as described by a Choi-Jamiołkowski state) that was cube-separable but not a quantum separability preserving/entanglement breaking/Clifford operation, therefore describing classically simulatable operations outside previous regimes of operations that can be efficiently simulated classically. The hope here was that the noise required to take the CZ gate to a cube-separable operation would be less than that needed for CZ output quantum separability. If we could show we need less noise to make our CZ cube-separable, then we have shown that we need less noise than existing schemes to take a quantum computer to a system that can be efficiently simulated on a classical computer.

Unfortunately for the case of ideal preparation/measurement, with all of the noise models considered, the noise thresholds derived have been the same, or slightly worse, than those which could be achieved under existing schemes considering quantum separability.

The approach taken to calculate the noise threshold for Bloch cubes differed slightly from that taken with the rescaled Bloch sphere. The methodology followed that laid out in section (4.5.3) and used the symmetry arguments there to reduce the number of input states that we needed to consider.

As we were no longer dealing with Bloch spheres we could not use the PPT criterion to determine separability. We looked to consider the amount of noise required to ensure Pauli measurements on the CZ output states returned positive probability outcomes. Specific measurements were identified as requiring the least amount of noise for positivity, and we then manually identified and showed that Bloch-cube separability could

be achieved at those noise rates by constructing a LHVM. That is, we showed that the CZ output state could be expressed as a convex combination of Bloch cube product states.

5.5.1 Joint depolarising noise for cubes

As described in section (4.6.1) the effect of the joint depolarising noise acting immediately on the output state of the CZ operation takes the state to the following table:

$$\frac{(1-\varepsilon)}{4}\begin{Bmatrix} \frac{1}{(1-\varepsilon)} & zA & zB & C \\ xC & yB & -yA & x \\ yC & -xB & xA & y \\ z & A & B & zC \end{Bmatrix} \tag{5.12}$$

If we again consider the effect of a Pauli $X \otimes X$ measurement, the probability of obtaining $+1$ for the X measurement on the first particle, and -1 for the X measurement on the second particle can be expressed in a modified expression to (4.6.2):

$$P(+1,\,-1) = \frac{1}{4}\left(1 + (1-\varepsilon)(xC - zA - yB)\right) \tag{5.13}$$

When $xC = -1$, $yB = 1$, and $zA = 1$, we require ε to be at least $\frac{2}{3}$ to ensure that $P(+1,\,-1)$ is positive. We can show that this rate of noise is sufficient by showing that at this noise rate we can construct a LHVM for the system. Using the symmetry arguments from section (4.5), we only need to consider one input state for Bloch cubes. Here we choose the input state undergoing the CZ operation to have coefficients $x = y = z = A = B = C = 1$, and show that the cube-separable state can be constructed as follows:

$$\begin{Bmatrix} 1 & \frac{1}{3} & \frac{1}{3} & \frac{1}{3} \\ \frac{1}{3} & \frac{1}{3} & -\frac{1}{3} & \frac{1}{3} \\ \frac{1}{3} & -\frac{1}{3} & \frac{1}{3} & \frac{1}{3} \\ \frac{1}{3} & \frac{1}{3} & \frac{1}{3} & \frac{1}{3} \end{Bmatrix} = \frac{1}{3}\begin{Bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \end{Bmatrix} + \frac{2}{3}\begin{Bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{Bmatrix} \tag{5.14}$$

Where the second term itself may be constructed in terms of cube-separable states as:

$$\begin{Bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{Bmatrix} \propto \sum_{p,q,r,s\in\pm 1} \begin{Bmatrix} 1 & p & q & r \\ -q & -qp & -q^2 & -qr \\ -p & -p^2 & -pq & -pr \\ s & sp & sq & sr \end{Bmatrix} \tag{5.15}$$

The noise rate of $\frac{2}{3}$ is the same as that required in the Bloch sphere/quantum separability case where a $\frac{2}{3}$ joint depolarised noise rate is needed to take an EPR pair to a quantum-separable state.

5.5.2 Local depolarising noise for cubes

The effect of this noise acting immediately on the output of the CZ operation is to take the input state to the following table representation:

$$\frac{(1-\varepsilon)^2}{4}\begin{Bmatrix} \frac{1}{(1-\varepsilon)^2} & \frac{zA}{(1-\varepsilon)} & \frac{zB}{(1-\varepsilon)} & \frac{C}{(1-\varepsilon)} \\ \frac{xC}{(1-\varepsilon)} & yB & -yA & x \\ \frac{yC}{(1-\varepsilon)} & -xB & xA & y \\ \frac{z}{(1-\varepsilon)} & A & B & zC \end{Bmatrix} \tag{5.16}$$

Once again we consider looking at a Pauli $X \otimes X$ measurement. The probability of obtaining $+1$ for the X measurement on the first particle, and -1 for the second takes the form:

$$P(+1,\,-1) = \frac{1}{4}\Big(1 + (1-\varepsilon)(xC - zA) - (1-\varepsilon)^2 yB\Big) \tag{5.17}$$

Again, choosing the input state to have coefficients $xC = -1$, $yB = 1$, and $zA = 1$, our expression becomes:

$$P(+1,\,-1) = \frac{1}{4}(-2 + 4\varepsilon - \varepsilon^2) \tag{5.18}$$

The probability $P(+1,\,-1)$ remains positive where $\varepsilon \geq 2 - \sqrt{2} \approx 60\%$. At this noise rate we can show that the output state has a LHVM as follows: We are free to choose the input state to the CZ operation to have coefficients $x = y = z = A = B = C = 1$, the output state has a LHVM that can be constructed as follows:

$$\frac{1}{4}\mathcal{W} \propto (1 - 2(1-\varepsilon) - (1-\varepsilon)^2)\mathcal{X} + ((1-\varepsilon) - (1-\varepsilon)^2)\left(\mathcal{Y} + \mathcal{Y}'\right) + 3(1-\varepsilon)^2\mathcal{Z} \tag{5.19}$$

where each component corresponds to the tables:

$$\mathcal{W} = \begin{Bmatrix} 1 & (1-\varepsilon) & (1-\varepsilon) & (1-\varepsilon) \\ (1-\varepsilon) & (1-\varepsilon)^2 & -(1-\varepsilon)^2 & (1-\varepsilon)^2 \\ (1-\varepsilon) & -(1-\varepsilon)^2 & (1-\varepsilon)^2 & (1-\varepsilon)^2 \\ (1-\varepsilon) & (1-\varepsilon)^2 & (1-\varepsilon)^2 & (1-\varepsilon)^2 \end{Bmatrix}, \qquad \mathcal{X} = \begin{Bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{Bmatrix}$$

$$\mathcal{Y} = \begin{Bmatrix} 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{Bmatrix}, \qquad \mathcal{Y}' = \begin{Bmatrix} 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{Bmatrix}, \qquad \mathcal{Z} = \begin{Bmatrix} 1 & \frac{1}{3} & \frac{1}{3} & \frac{1}{3} \\ \frac{1}{3} & \frac{1}{3} & -\frac{1}{3} & \frac{1}{3} \\ \frac{1}{3} & -\frac{1}{3} & \frac{1}{3} & \frac{1}{3} \\ \frac{1}{3} & \frac{1}{3} & \frac{1}{3} & \frac{1}{3} \end{Bmatrix}$$

and each of the component tables have the following decomposition:

$$\mathcal{X} = \frac{1}{4}\begin{Bmatrix} 1 & 1 & 1 & 1 \\ -1 & -1 & -1 & -1 \\ -1 & -1 & -1 & -1 \\ -1 & -1 & -1 & -1 \end{Bmatrix} + \frac{1}{4}\begin{Bmatrix} 1 & -1 & -1 & -1 \\ 1 & -1 & -1 & -1 \\ 1 & -1 & -1 & -1 \\ 1 & -1 & -1 & -1 \end{Bmatrix} + \frac{1}{2}\begin{Bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \end{Bmatrix}$$

$$\mathcal{Y} = \frac{1}{2}\left\{\begin{array}{cccc} 1 & 1 & 1 & 1 \\ -1 & -1 & -1 & -1 \\ -1 & -1 & -1 & -1 \\ -1 & -1 & -1 & -1 \end{array}\right\} + \frac{1}{2}\left\{\begin{array}{cccc} 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \end{array}\right\}$$

$$\mathcal{Y}' = \frac{1}{2}\left\{\begin{array}{cccc} 1 & -1 & -1 & -1 \\ 1 & -1 & -1 & -1 \\ 1 & -1 & -1 & -1 \\ 1 & -1 & -1 & -1 \end{array}\right\} + \frac{1}{2}\left\{\begin{array}{cccc} 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \end{array}\right\}$$

$$\mathcal{Z} \rightarrow \text{see eq. (5.14)} \tag{5.20}$$

The noise rate of $\varepsilon = 2 - \sqrt{2} \approx 59\%$ is worse than that required to take an arbitrary entangled two qubit quantum pure state into the set of quantum-separable states. Using the PPT criterion this noise rate can be shown to be $\varepsilon = 1 - \frac{1}{\sqrt{3}} \approx 42\%$.

5.5.3 Local dephasing noise for cubes

The effect of this noise acting immediately on the output of the CZ operation is to take the state to the following table representation:

$$\frac{1}{4}\left\{\begin{array}{cccc} 1 & (1-2\varepsilon)zA & (1-2\varepsilon)zB & C \\ (1-2\varepsilon)xC & (1-2\varepsilon)^2 yB & -(1-2\varepsilon)^2 yA & (1-2\varepsilon)x \\ (1-2\varepsilon)yC & -(1-2\varepsilon)^2 xB & (1-2\varepsilon)^2 xA & (1-2\varepsilon)y \\ z & (1-2\varepsilon)A & (1-2\varepsilon)B & zC \end{array}\right\} \tag{5.21}$$

Measuring the Pauli operators $X \otimes X$, the probability of obtaining $+1$ for the X direction measurement on the first particle, and -1 for the second takes the form:

$$P(+1, -1) = \frac{1}{4}\Big(1 + (1-2\varepsilon)(zA + xC) + (1-2\varepsilon)^2 yB\Big) \tag{5.22}$$

Where the input product cube-pure state has coefficients $x = -C$, $z = A$, $y = B$, this expression takes the form:

$$P(+1, -1) = \frac{1}{4}(1 - (1-2\varepsilon) - (1-2\varepsilon)^2) \tag{5.23}$$

The noise rate of $\varepsilon = 1 - \frac{1}{\sqrt{2}} \approx 30\%$ is sufficient to ensure this measurement remains positive. For this amount of noise, a system whose input state is a product cube-pure state, with coefficients $x = y = z = A = B = C = 1$ leads to a system with a LHVM as follows:

$$\frac{1}{4}\mathcal{F} \propto \Big(1-2(1-\varepsilon)-(1-\varepsilon)^2\Big)\,\mathcal{G} + \Big((1-\varepsilon)-(1-\varepsilon)^2\Big)\,(\mathcal{H}+\mathcal{H}') + 3(1-\varepsilon)^2\mathcal{I} \quad (5.24)$$

where each component corresponds to the tables:

$$\mathcal{F} = \begin{Bmatrix} 1 & (1-2\varepsilon) & (1-2\varepsilon) & 1 \\ (1-2\varepsilon) & (1-2\varepsilon)^2 & -(1-2\varepsilon)^2 & (1-2\varepsilon) \\ (1-2\varepsilon) & -(1-2\varepsilon)^2 & (1-2\varepsilon)^2 & (1-2\varepsilon) \\ 1 & (1-2\varepsilon) & (1-2\varepsilon) & 1 \end{Bmatrix}, \qquad \mathcal{G} = \begin{Bmatrix} 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 \end{Bmatrix}$$

$$\mathcal{H} = \begin{Bmatrix} 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 \end{Bmatrix}, \qquad \mathcal{H}' = \begin{Bmatrix} 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 \end{Bmatrix}, \qquad \mathcal{I} = \begin{Bmatrix} 1 & \frac{1}{3} & \frac{1}{3} & \frac{1}{3} \\ \frac{1}{3} & \frac{1}{3} & -\frac{1}{3} & \frac{1}{3} \\ \frac{1}{3} & -\frac{1}{3} & \frac{1}{3} & \frac{1}{3} \\ \frac{1}{3} & \frac{1}{3} & \frac{1}{3} & \frac{1}{3} \end{Bmatrix}$$

Where the component tables have the following decomposition:

$$\mathcal{G} = \frac{1}{4}\begin{Bmatrix} 1 & 1 & 1 & 1 \\ -1 & -1 & -1 & -1 \\ -1 & -1 & -1 & -1 \\ 1 & 1 & 1 & 1 \end{Bmatrix} + \frac{1}{4}\begin{Bmatrix} 1 & -1 & -1 & 1 \\ 1 & -1 & -1 & 1 \\ 1 & -1 & -1 & 1 \\ 1 & -1 & -1 & 1 \end{Bmatrix}$$
$$+\frac{1}{4}\begin{Bmatrix} 1 & -1 & -1 & 1 \\ -1 & 1 & 1 & -1 \\ -1 & 1 & 1 & -1 \\ 1 & -1 & -1 & 1 \end{Bmatrix} + \frac{1}{4}\begin{Bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \end{Bmatrix}$$

$$\mathcal{H} = \frac{1}{2}\begin{Bmatrix} 1 & 1 & 1 & 1 \\ -1 & -1 & -1 & -1 \\ -1 & -1 & -1 & -1 \\ 1 & 1 & 1 & 1 \end{Bmatrix} + \frac{1}{2}\begin{Bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \end{Bmatrix}$$

$$\mathcal{H}' = \frac{1}{2}\begin{Bmatrix} 1 & -1 & -1 & 1 \\ 1 & -1 & -1 & 1 \\ 1 & -1 & -1 & 1 \\ 1 & -1 & -1 & 1 \end{Bmatrix} + \frac{1}{2}\begin{Bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \end{Bmatrix}$$

$$\mathcal{I} \rightarrow \text{see eq. (5.14)} \quad (5.25)$$

This value of $\varepsilon \geq 1 - \frac{1}{\sqrt{2}} \approx 30\%$, is the same as that when considering quantum-separability. A local dephasing noise at this rate is sufficient to make the CZ operation quantum separable.

5.6 Rescaled Bloch cube states

Earlier in this chapter in section (5.4) we found that adding some noise immediately prior to measurements allowed for less noise needed to make a noisy CZ operation separable with respect to a rescaled Bloch sphere state space. This rescaled state space was defined without any restrictions to the available measurements. Within this second half of the chapter we consider the effects of considering non-ideal measurement/preparations in addition to measurement restrictions in defining our single particle state-space.

The point being, that noise on the Pauli measurements might allow for a reduction in the noise needed to make the CZ operation separable with respect to the rescaled Bloch-cubes. As the three noise models we will look at are mixtures of Pauli operators, the arguments presented in section (4.5.3) still apply, and it is sufficient to only consider one product pure rescaled cube state.

Of particular relevance to magic state architectures are the rescaling noise factors that scale the Bloch-cube states such that they still contain the set of T-type and H-type magic states. The amount of noise required for separability of these state spaces would indicate the amount of noise needed to classically simulate stabiliser state based computation with access to pure magic states of those types (see section (2.3)). Where $R \geq \frac{1}{\sqrt{2}}$ our rescaled Bloch cube contains both sets of T-type and H-type magic states, where $\frac{1}{\sqrt{3}}$ the rescaled Bloch cube contains just the set of T-type magic states. The rescaling value $R = \frac{1}{\sqrt{3}}$ also corresponds to the local depolarising noise required to take a Bell state to a quantum-separable state. Given that Bell states are cube-separable (see section (4.4)), for the rescaling parameter $1 \geq R > \frac{1}{\sqrt{3}}$, the device can have access to quantum entangled pure Bell diagonal states.

We shall begin by describing the transformation of rescaled input cube states under the action of noisy CZ gates. The sequence of transformations on our product cube pure input states goes as; rescaling of the input states in terms of the rescaling parameter $R > 0$, performing the CZ operation, enacting a noise model characterised by the parameter $r < 1$, and finally undoing the rescaling operation.

As with the ideal preparation/measurement Bloch cube case our approach here was to consider Pauli measurements that require the least amount of noise for positivity. We then commented on whether we could obtain a LHVM and output separability at that noise rate.

5.6.1 Local depolarising noise for rescaled cubes

For the cube pure input states with the Bloch vectors (x, y, z) and (A, B, C), under the rescaling factor R our cube-pure inputs are transformed as follows:

$$\left\{\begin{array}{cccc} 1 & A & B & C \\ x & xA & xB & xC \\ y & yA & yB & yC \\ z & zA & zB & zC \end{array}\right\} \rightarrow \left\{\begin{array}{cccc} 1 & RA & RB & RC \\ Rx & R^2xA & R^2xB & R^2xC \\ Ry & R^2yA & R^2yB & R^2yC \\ Rz & R^2zA & R^2zB & R^2zC \end{array}\right\} \tag{5.26}$$

Under the action of the CZ operation eq. (5.26) is transformed to:

$$\left\{\begin{array}{cccc} 1 & R^2zA & R^2zB & RC \\ R^2xC & R^2yB & -R^2yA & Rx \\ R^2yC & -R^2xB & R^2xA & Ry \\ Rz & RA & RB & R^2zC \end{array}\right\} \tag{5.27}$$

The local depolarising noise at a rate of ε takes eq. (5.27) to:

$$\left\{\begin{array}{cccc} 1 & (1-\varepsilon)R^2zA & (1-\varepsilon)R^2zB & (1-\varepsilon)RC \\ (1-\varepsilon)R^2xC & (1-\varepsilon)^2R^2yB & -(1-\varepsilon)^2R^2yA & (1-\varepsilon)^2Rx \\ (1-\varepsilon)R^2yC & -(1-\varepsilon)^2R^2xB & (1-\varepsilon)^2R^2xA & (1-\varepsilon)^2Ry \\ (1-\varepsilon)Rz & (1-\varepsilon)^2RA & (1-\varepsilon)^2RB & (1-\varepsilon)^2R^2zC \end{array}\right\} \tag{5.28}$$

Following undoing the rescaling operation R:

$$\left\{\begin{array}{cccc} 1 & (1-\varepsilon)RzA & (1-\varepsilon)RzB & (1-\varepsilon)C \\ (1-\varepsilon)RxC & (1-\varepsilon)^2yB & -(1-\varepsilon)^2yA & (1-\varepsilon)^2\frac{1}{R}x \\ (1-\varepsilon)RyC & -(1-\varepsilon)^2xB & (1-\varepsilon)^2xA & (1-\varepsilon)^2\frac{1}{R}y \\ (1-\varepsilon)z & (1-\varepsilon)^2\frac{1}{R}A & (1-\varepsilon)^2\frac{1}{R}B & (1-\varepsilon)^2zC \end{array}\right\} \tag{5.29}$$

For cube-pure input states where x, y, z, A, B, $C \in \pm 1$, the table (5.29) becomes:

$$\left\{\begin{array}{cccc} 1 & (1-\varepsilon)R & (1-\varepsilon)R & (1-\varepsilon) \\ (1-\varepsilon)R & (1-\varepsilon)^2 & -(1-\varepsilon)^2 & \frac{(1-\varepsilon)^2}{R} \\ (1-\varepsilon)R & -(1-\varepsilon)^2 & (1-\varepsilon)^2 & \frac{(1-\varepsilon)^2}{R} \\ (1-\varepsilon) & \frac{(1-\varepsilon)^2}{R} & \frac{(1-\varepsilon)^2}{R} & (1-\varepsilon)^2 \end{array}\right\} \tag{5.30}$$

As before we need to ensure that all possible measurements available have must positive probabilities for our generalised theory to be consistent. From numerics we identify two pairs of Pauli measurements as requiring the most amount of noise for positivity. The probabilities of measuring, -1 for X, -1 for Y, are given by the Born rule as:

$$\frac{1}{4}\Big(1-(1-\varepsilon)^2-2(1-\varepsilon)R\Big) \geq 0$$

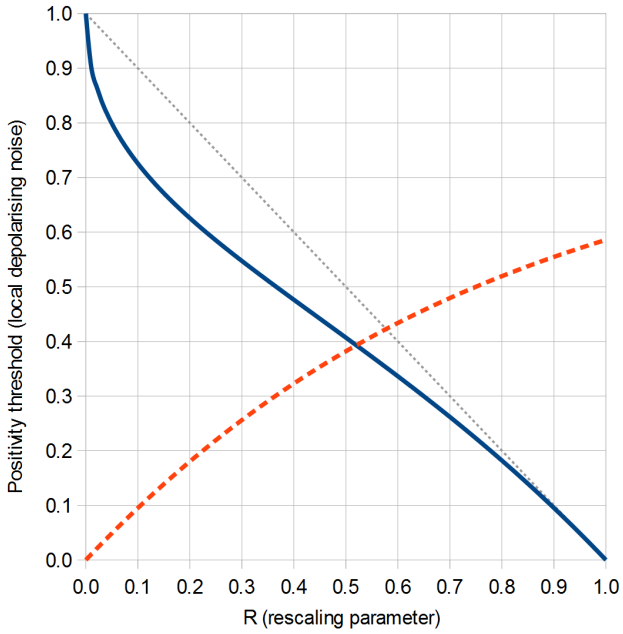


Figure 5.4: Local depolarising noise ε on the vertical axis against the rescaling parameter R on the horizontal axis. The increasing curve corresponds to eq. (5.31) derived from ensuring the positivity of -1 Pauli X, -1 Y, measurement. The decreasing curve corresponds to eq. (5.32) looking at the positivity of $+1$ for X, -1 for Z, measurements. The regions of valid values of ε, R ensuring both measurement probabilities remain positive lie below the two curve. The dotted line corresponds to $\varepsilon = 1 - R$. Image has been taken from [7].

$$\varepsilon \geq 1 - \sqrt{1 + R^2} - R \tag{5.31}$$

Similarly getting $+1$ for a X, and -1 for a Z measurement have the probabilities:

$$\frac{1}{4}\left(1 + (1-\varepsilon)R - (1-\varepsilon) - \frac{1}{R}(1-\varepsilon)^2\right) \geq 0$$

$$\varepsilon \geq 1 - \frac{R - 1 + \sqrt{(R-1)^2 + \frac{4}{R}}}{\left(\frac{2}{R}\right)} \tag{5.32}$$

These inequalities form the probabilities that require the smallest value of ε to ensure cube separability, and may be plotted as in figure (5.4).

We can see from figure (5.4) that the line corresponding to eq. (5.31), increases monotonically for $R > 1$, therefore make choosing $R > 1$ (allowing for imperfect measurements) unnecessary. We will therefore restrict ourselves to only looking at $R \leq 1$. As we can see from figure (5.4) the smallest ε required to ensure both states remain positive is given at the value of ε equalling approximately 39.29%, for noise on preparation

$1 - R$ equalling approximately 47.99%. This value of ε is smaller than the value of $\varepsilon = 42.27\%$, shown in earlier calculations to ensure that a CZ operation is quantum separable.

We do not believe [7] from numerical investigations that the value of ε= 39.29% could correspond to a set of valid LHVM for the set of measurements available. It is believed that eq. (5.31) only has a LHVM for $R \geq 54.49\%$ and $\varepsilon = 40.61\%$ approximately. This value of R is still sufficiently large to allow for the magic states to still remain within the rescaled Bloch cube, the H-type magic states are contained in a cube of size $R = \frac{1}{\sqrt{2}}$, and the T-type in $R = \frac{1}{\sqrt{3}}$.

The rescaling parameters $R = \frac{1}{\sqrt{2}}$, and $R = \frac{1}{\sqrt{3}}$, correspond to the local depolarising values $\varepsilon = 1 - \frac{\sqrt{3}-1}{\sqrt{2}} \approx 48.24\%$, and $\varepsilon = 1 - \frac{1}{\sqrt{3}} \approx 42.27\%$, respectively. This last value of ε means that magic state architectures with T-type magic state distillation as described in section (2.3) can become classically simulatable where CZ gates and Pauli eigenstate preparation are subject to local depolarising noise at a rate of approximately 42.27%.

5.6.2 Joint depolarising noise for rescaled cubes

We will now consider state transformations as in eq.'s (5.26) to (5.30) using joint depolarising noise instead of local depolarising. The final output state immediately prior to measurement can be expressed using the following table:

$$\left\{ \begin{array}{cccc} 1 & (1-\varepsilon)R & (1-\varepsilon)R & (1-\varepsilon) \\ (1-\varepsilon)R & (1-\varepsilon) & -(1-\varepsilon) & \frac{(1-\varepsilon)}{R} \\ (1-\varepsilon)R & -(1-\varepsilon) & (1-\varepsilon) & \frac{(1-\varepsilon)}{R} \\ (1-\varepsilon) & \frac{(1-\varepsilon)}{R} & \frac{(1-\varepsilon)}{R} & (1-\varepsilon) \end{array} \right\} \tag{5.33}$$

Once again numerical investigation helps us determine two Pauli measurement probabilities that allow us that are being important in calculating the values of ε and R for which we get a cube-separable output. The probability of measuring -1 for X, -1 for Y, using the Born rule can be gives us:

$$\frac{1}{4}\left(1 - (1-\varepsilon) - 2(1-\varepsilon)R\right) \geq 0$$

$$\varepsilon \geq 1 - \frac{1}{2R+1} \tag{5.34}$$

The second important measurement, measuring $+1$ for a Y, and -1 for a Z measurement gives the probabilities:

$$\frac{1}{4}\left(1 - (1-\varepsilon) - \frac{(1-\varepsilon)}{R} + (1-\varepsilon)R\right) \geq 0$$

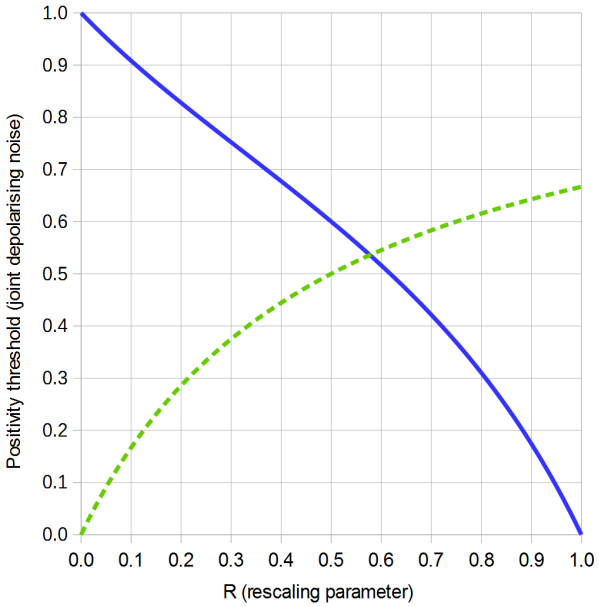


Figure 5.5: Joint depolarising noise ε on the vertical axis against the rescaling parameter R on the horizontal axis. The increasing curve corresponds to eq. (5.34) derived from ensuring the positivity of -1 Pauli X, -1 Y, measurement. The decreasing curve corresponds to eq. (5.35) looking at the positivity of $+1$ for Y, -1 for Z, measurements. The regions of valid values of ε, R ensuring both measurement probabilities remain positive lie above the two curves. Numerical investigations lead us to believe the intersection point of the two curves is achievable by a LHVM. This intersection point have ε, and R values of $\frac{1}{\sqrt{3}} \approx 58\%$, and $\frac{\sqrt{3}}{(2+\sqrt{3})} \approx 46\%$ respectively. Image has been taken from [7].

$$\varepsilon \geq 1 - \frac{1}{1 + \left(\frac{1}{R}\right) - R} \tag{5.35}$$

Plotting the inequalities in figure (5.5) shows that as the inequalities in eq. (5.34) are monotonically increase for larger R, to find the smallest noise value for separability we can restrict ourselves to $R < 1$.

We can see from numerical results that the inequality (5.34) for local depolarising noise on the preparation of cube states, is achievable by a LHVM for a value of $1 - R = 1 - \frac{1}{\sqrt{2}} \approx 29.29\%$, corresponding to the value of $\varepsilon = 1 - \frac{1}{\sqrt{2}+1}$. We therefore need a joint depolarising noise rate ε of approximately 58.58%. We can see that for this value of R that the pure T-type and H-type magic states remain within the rescaled Bloch cube. Therefore for a magic state architecture based quantum system with access to both T and H-type magic states, we can show that the system is classically efficiently simulatable if there is at least approximately 29.29% local depolarising noise in Pauli eigenstate preparations, and at least approximately 58.58% joint depolarising noise on the CZ gates forming the system.

5.6.3 Local dephasing noise for rescaled cubes

Under local dephasing noise the final output state has the table representation:

$$\left\{\begin{array}{cccc} 1 & (1-2\varepsilon)\,R & (1-2\varepsilon)\,R & 1 \\ (1-2\varepsilon)\,R & (1-2\varepsilon)^2 & -(1-2\varepsilon)^2 & \frac{(1-2\varepsilon)}{R} \\ (1-2\varepsilon)\,R & -(1-2\varepsilon)^2 & (1-2\varepsilon)^2 & \frac{(1-2\varepsilon)}{R} \\ 1 & \frac{(1-2\varepsilon)}{R} & \frac{(1-2\varepsilon)}{R} & 1 \end{array}\right\} \tag{5.36}$$

For any $R \neq 1$, we only have rescaled cube state separability of the output state when the dephasing is total. i.e. where $\varepsilon = \frac{1}{2}$. This is easier to see if we explicitly calculate the measurement probability for measurements of -1 for Z, and $+1$ for X, or -1 for Z, and -1 for X:

$$\frac{1}{4}\left(1 - 1 + (1-2\varepsilon)R - \frac{(1-2\varepsilon)}{R}\right) \geq 0 \tag{5.37}$$

$$\frac{1}{4}\left(1 - 1 - (1-2\varepsilon)R + \frac{(1-2\varepsilon)}{R}\right) \geq 0 \tag{5.38}$$

We can see from eq. (5.37) and eq. (5.38) that unless we have $\varepsilon = \frac{1}{2}$ - total dephasing, our generalised theory is not consistent with the measurement outcomes from the state as we would get negative probabilities for measurements.

5.7 Summary of Chapter (5)

This chapter looked at another type of measurement restriction - fault on input preparation/output measurement. We saw how such faults allowed us to define new state spaces characterised by a rescaling parameter that described the amount by which the unmodified state space was shrunk or enlarged. Rescaling was considered as another variation in the calculation of $\lambda_{CZ_\varepsilon,\mathcal{S}}$.

The first half of this chapter studied the Bloch sphere modified for faulty preparations/measurements. Continuing on from section (5.2), we began by describing the methodology used in calculating $\lambda_{CZ_\varepsilon,\mathcal{S}}$ for rescaled Bloch spheres. Symmetries specific to the Bloch sphere reducing the problem were stated, and the following subsections presented numerical results for the calculation of $\lambda_{CZ_\varepsilon,\mathcal{S}}$ for each noise model for varying rescaling parameter.

For both the joint depolarising noise, and local depolarising noise case it was observed that adding noise to the measurement reduced the amount of noise required for separable output. For local depolarising noise, we required slight noise on the measurements of $R \approx 1.16$ with a noise rate of $\approx 39.5\%$, and for joint depolarising, more noise on measurements with $R \approx 1.73$, and a corresponding noise rate of $\approx 53.6\%$. In the local dephasing case it was observed that adding any fault to preparation/measurement re-

quired maximal noise to ensure separable output. We were not able to obtain analytical solutions in any of the cases.

In the second half we focused on the Bloch cube - the largest state space returning positive probabilities for Pauli measurements. We started by calculating $\lambda_{CZ_\varepsilon,\mathcal{S}}$ analytically for each noise model in the instance where there was no preparation/measurement fault. In both the joint depolarising, and local dephasing cases we require the same amount of noise as the Bloch sphere case. In the local depolarising case we had worse results requiring more noise.

We then looked at the Bloch cubes modified to account for faulty preparations and measurements. In the local depolarising and joint depolarising noise cases we observed that the inclusion of noise in the preparation allows for a reduction in the noise needed for separability. For local depolarising noise we believed we had a valid LHVM for output states with noisy preparation of $1 - R \geq 54.49\%$ and noise rates of $\varepsilon \approx 40.61\%$. This value is less than the $\varepsilon \approx 42.26\%$ required for quantum separability. With joint depolarising, noise numerics showed that we had a valid LHVM output states with preparation noise of $1 - R \approx 29.29\%$ and a corresponding noise rates of $\approx 58.58\%$ less than the $\approx 66.67\%$ needed for quantum separability. As with the rescaled Bloch spheres, in the local dephasing case we observed that adding any fault to preparation/measurement required maximal noise to ensure separable output.

There was no improvement to the threshold value as hoped from considering the Bloch cube state space over Bloch sphere. For the depolarising noise models both Bloch cubes and spheres achieved similar threshold values when allowing for noise on preparation/measurement, although the sphere case had noise on measurements enlargening the initial state space, and the cube case had noise on preparation shrinking the initial state space. This result suggests that state spaces smaller/larger in particular "directions" are preferred, lending more towards lowering the noise threshold than others, for the noise models considered. What these directions are is unclear from the analysis here, and still leaves open the question if there exists a state space $\mathcal{S}$ between $\mathcal{Q} \subseteq \mathcal{S} \subseteq \mathcal{M}^*$ that gives lower threshold values.

6 The truncated cube state space

6.1 Introduction

The previous chapter looked at the most straightforward choice of $\mathcal{S}$ as the full dual to Pauli measurements with little gained in terms of the reduction of the noise needed for $\mathcal{S}$-separability of the CZ gate. We did show there exists a shape $\mathcal{S}$ requiring less joint depolarising noise for CZ $\mathcal{S}$-separability than the Bloch sphere. This $\mathcal{S}$ was the rescaled Bloch cube - a shrunken version of the Bloch cube, rescaled by a factor of $R = \frac{1}{\sqrt{2}} \approx 0.707$.

Previously we considered state spaces defined in terms of restrictions on measurements, either through inherent restrictions on the measurements available, or through noise on measurements. In this chapter we relax our state space definition, requiring that $\mathcal{S}$ satisfies $\mathcal{Q} \subset \mathcal{S} \subset \mathcal{M}^*$ where $\mathcal{M}^*$ is the Bloch cube. We maintain the restriction to Pauli measurements and require $\mathcal{S}$ satisfy the restrictions listed in section (4.5), but we no longer consider all possible operators that return positive probabilities as forming the state space. As there are infinitely many choices for $\mathcal{S}$ we instead choose to focus on one - the *truncated cube* state space.

While we believe that it is not the optimal choice for $\mathcal{S}$ requiring the least possible noise for CZ output separability, it does require less joint depolarising noise than any other $\mathcal{S}$ considered up to this point and is the best candidate we have. The results presented in this chapter can be found in [8]. It is important to note that within this chapter we won't be considering any noise on measurements and that for all truncated cube state spaces $R = 1$.

6.2 The truncated cube state spaces

We shall now define the set of *truncated cube* single particle state spaces. These spaces can be formed by taking the Bloch cube and "slicing-off" each of its eight vertices by varying degrees, quantifying the amount sliced off each vertex by a parameter c.

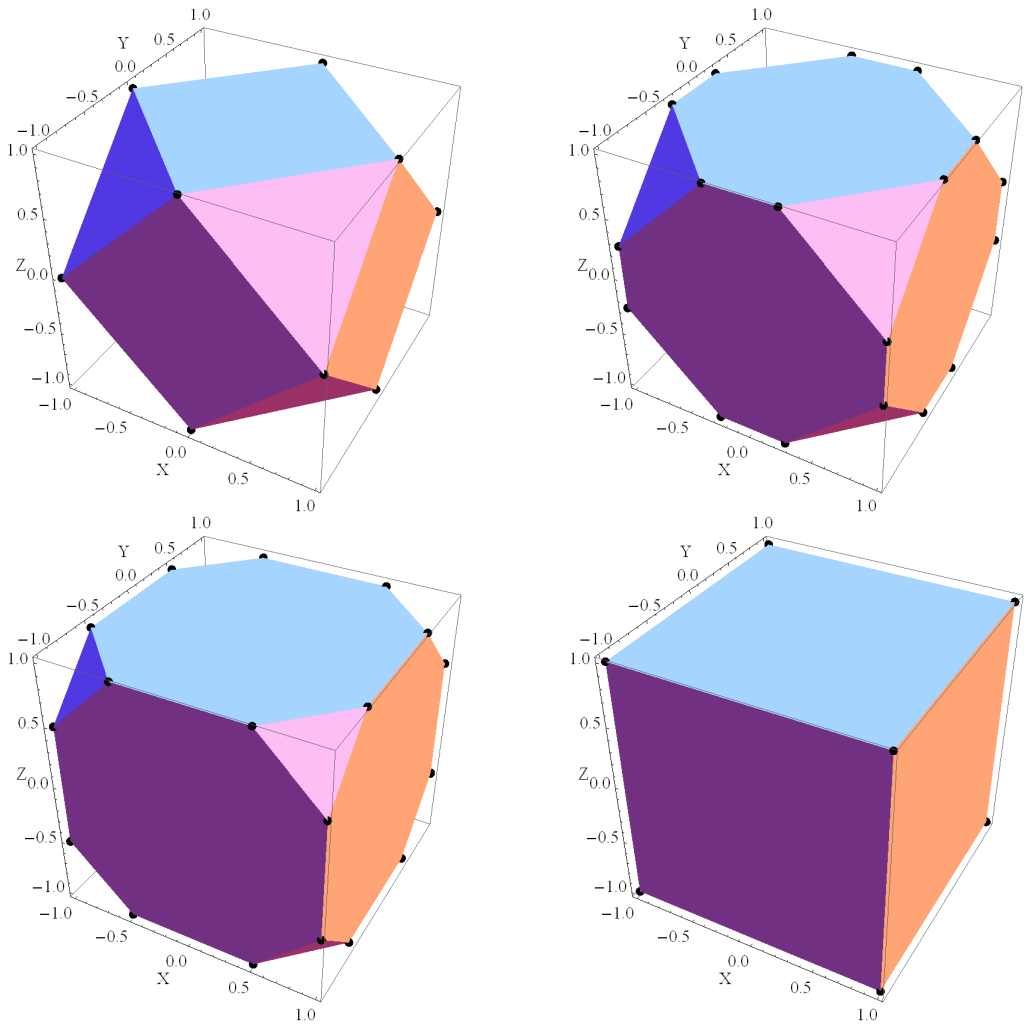


Figure 6.1: Plot of truncated cube state spaces with varying c; $c = 0$ (top-left), $c = 0.25$ (top-right), $c = 0.5$ (bottom-left), $c = 1$ (bottom-right).

Definition 6.2.1. The truncated cube state spaces.

Truncated cube state spaces (denoted $TRUNC(c)$) are defined to be the convex hulls of vertices represented by the following Bloch vectors with i, j, $k \in \{0, 1\}$:

$$\left((-1)^i c, (-1)^j, (-1)^k\right), \left((-1)^i, (-1)^j c, (-1)^k\right), \left((-1)^i, (-1)^j, (-1)^k c\right) \quad (6.1)$$

where the truncation parameter c, satisfies $0 < c \leq 1$.

Figure (6.1) shows plots of truncated cube state spaces for varying values of c. It is important to note that $TRUNC(c)$ becomes a Bloch cube when $c = 1$, and a cubeoctahedron for $c = 0$.

We will denote the number of vertices a given $TRUNC(c)$ has by N. This value can vary based on the value of c, with $N = 12$ for $c = 0$, $N = 24$ for $0 < c < 1$, and $N = 12$ for $c = 1$.

6.2.1 Input states from symmetry arguments

It is possible to reduce the number of product states we need to consider by considering the symmetry arguments detailed in section (4.5.2). We therefore only need to look at the noise needed to ensure separability for the following 6 product input states with the Pauli tables:

$$\varrho_1 : \frac{1}{4}\begin{Bmatrix} 1 & c & 1 & 1 \\ c & c^2 & c & c \\ 1 & c & 1 & 1 \\ 1 & c & 1 & 1 \end{Bmatrix} \quad \varrho_2 : \frac{1}{4}\begin{Bmatrix} 1 & 1 & c & 1 \\ c & c & c^2 & c \\ 1 & 1 & c & 1 \\ 1 & 1 & c & 1 \end{Bmatrix} \quad \varrho_3 : \frac{1}{4}\begin{Bmatrix} 1 & 1 & 1 & c \\ c & c & c & c^2 \\ 1 & 1 & 1 & c \\ 1 & 1 & 1 & c \end{Bmatrix}$$

$$\varrho_4 : \frac{1}{4}\begin{Bmatrix} 1 & 1 & c & 1 \\ 1 & 1 & c & 1 \\ c & c & c^2 & c \\ 1 & 1 & c & 1 \end{Bmatrix} \quad \varrho_5 : \frac{1}{4}\begin{Bmatrix} 1 & 1 & 1 & c \\ 1 & 1 & 1 & c \\ c & c & c & c^2 \\ 1 & 1 & 1 & c \end{Bmatrix} \quad \varrho_6 : \frac{1}{4}\begin{Bmatrix} 1 & 1 & 1 & c \\ 1 & 1 & 1 & c \\ 1 & 1 & 1 & c \\ c & c & c & c^2 \end{Bmatrix} \tag{6.2}$$

6.3 Pauli measurement positivity necessity bounds

We are interested in understanding the relationship between the c parameter and the amount of noise needed to ensure that the noisy CZ gate becomes separable with respect to $TRUN(c)$. We will look to determine what the optimal c value is requiring the least noise.

As before, we can use positivity to check that the noise rates are large enough so that at least there is enough noise to ensure positivity of local Pauli measurements probabilities as we checked in Chapter (7). In other word we are checking that the following is true for all relevant $\varrho \in TRUNC(c)^{\otimes 2}$:

$$\text{tr}\left[CUBE(\alpha = 1)^{*\otimes 2} CZ_\varepsilon(\varrho)\right] \geq 0 \tag{6.3}$$

From working through each ϱ and $CUBE(\alpha = 1)^{*\otimes 2}$ element, we obtain an inequality relating necessary noise ε to the truncated cube parameter c:

$$\varepsilon \geq \frac{1 + c^2}{2 + c^2} \tag{6.4}$$

Figure (6.2) shows a plot of the amount of noise needed to ensure Pauli measurement positivity verses truncation parameter c.

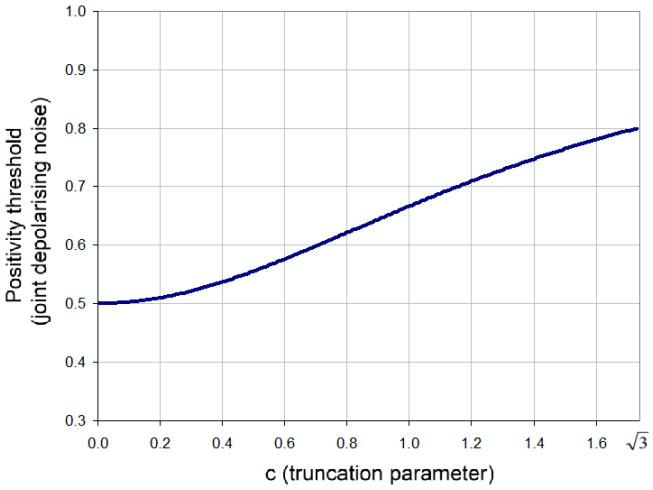


Figure 6.2: Plot of truncated cube state spaces with varying c values plotted against the amount of joint depolarising noise to ensure positivity of local Pauli measurements on the output of the noisy CZ gate.

6.4 Least noise rate as a linear programming problem

Up to this point we have shown various noise rates are sufficient by using educated guesses at what the convex sum of separable states that equal the noisy CZ gate output state are.

Here we will show that if $\mathcal{S}$ is a convex polytope (i.e. it has a finite number of extrema), and there is joint depolarising noise is acting on the CZ gate, then it is possible to determine the amount of noise needed to ensure CZ gate $\mathcal{S}$-separability, and work out the convex decomposition of the separable CZ output state by transforming the problem into a *linear programming problem* (LPP).

A good discussion of linear optimisation can be found in [64, 65].

In the next section we will be outlining the *simplex method* of solving LPP's, we will then present numerical results showing the necessary and sufficient rates of noise needed to ensure CZ separability with respect to $TRUN(c)$ state spaces.

After, we will show using the numerical results that we can arrive at a set of algebraic necessary bounds, and for specific values of c, attain exact algebraic convex decompositions of separable CZ output.

Suppose we have a single input product state $\varrho \in \mathcal{S}^{\otimes 2}$. The state ϱ transforms under the action of a noisy CZ gate effected by joint depolarising noise at a rate ε as:

$$\varrho \to \varrho' = (1-\varepsilon)CZ(\varrho) + \frac{\varepsilon}{4}\mathbb{1}^{\otimes 2} \tag{6.5}$$

If ε is sufficiently large then $\varrho' \in \mathcal{S}^{\otimes 2}$, i.e. ϱ' can be expressed as a convex sum of the N^2 extrema e_i of $\mathcal{S}^{\otimes 2}$ as:

$$(1-\varepsilon)CZ(\varrho) + \frac{\varepsilon}{4}\mathbb{1}^{\otimes 2} = \sum_j \eta_j e_j \qquad \text{where } 0 \leq \eta_j \leq 1,\ \sum_{j=1}^{N^2} \eta_j = 1 \tag{6.6}$$

Rather than using the usual Pauli table notation, we can represent each state in terms of a vector using the Pauli table elements as its elements. The general two particle state $\mathcal{A}'$ in eq. (4.22) may therefore be expressed as a 16 element column vector, transposed here as:

$$\Big(\mathcal{A}_{00}, \underbrace{\mathcal{A}_{01}, \mathcal{A}_{02}, \mathcal{A}_{03}, \mathcal{A}_{10}, \mathcal{A}_{11}, \mathcal{A}_{12}, \mathcal{A}_{13}, \mathcal{A}_{20}, \mathcal{A}_{21}, \mathcal{A}_{22}, \mathcal{A}_{23}, \mathcal{A}_{30}, \mathcal{A}_{31}, \mathcal{A}_{32}, \mathcal{A}_{33}}_{\underline{r}_{\mathcal{A}'}}\Big)^T \tag{6.7}$$

We will refer to a vector $\underline{r}_{\mathcal{A}'}$ formed from the last 15 elements of (6.7) as a *product Bloch vector of* $\mathcal{A}'$. Just as we can geometrically visualise normal Bloch vectors, we can visualise product Bloch vectors in a two-particle analogy, where each of the 15 dimensions corresponds to each Pauli tensor product direction i.e. the $X \otimes \mathbb{1}$ direction, $Y \otimes Z$, $\mathbb{1} \otimes X$ and so on.

We can use this vector form in eq. (6.7) to rearrange eq. (6.6) as

$$\underbrace{\varepsilon\left(CZ(\varrho) - \frac{\mathbb{1}^{\otimes 2}}{4}\right) + \sum_{j=1}^{N^2} \eta_j e_j}_{\boldsymbol{A}\underline{x}} = \underbrace{CZ(\varrho)}_{\underline{b}} \tag{6.8}$$

where the vector $\underline{x} = (\varepsilon, \eta_1, \eta_2, \ldots, \eta_{N^2})^{\mathrm{T}}$, and $\boldsymbol{A}$ is a $16 \times (N^2+1)$ dimensional matrix formed of vectors of the states as:

$$\left[\begin{pmatrix} \vdots \\ CZ(\varrho) - \frac{\mathbb{1}^{\otimes 2}}{4} \\ \vdots \end{pmatrix}, \begin{pmatrix} \vdots \\ e_1 \\ \vdots \end{pmatrix}, \begin{pmatrix} \vdots \\ e_2 \\ \vdots \end{pmatrix}, \begin{pmatrix} \vdots \\ e_3 \\ \vdots \end{pmatrix}, \cdots, \begin{pmatrix} \vdots \\ e_{(N^2)} \\ \vdots \end{pmatrix}\right] \tag{6.9}$$

We therefore have a system of linear equations, with potentially infinite solutions. In line with problem (4.3.4) in section (4.3), for each of the input states, we are looking to find a vector $\underline{x} \succcurlyeq \underline{0}$[1] solving $\boldsymbol{A}\underline{x} = \underline{b}$ such that ε is minimised. Note that where $0 < \varepsilon < 1$ and the $\eta_j' s$ satisfy the constraints in eq. (6.6) the minimal ε value would correspond to the amount of joint depolarising noise necessary to take the CZ output state back into the convex shape $\mathcal{S}^{\otimes 2}$. The ε values for each product input would finally be compared and the largest will be taken to be the $\lambda_{CZ_\varepsilon,\mathcal{S}}$ value. It turns out that problem of determining $\lambda_{CZ_\varepsilon,\mathcal{S}}$ as we have laid out here is a *linear programming problem* (LPP) that can be put into the *standard form*:

[1] Each element of $\underline{x}$ is greater than 0.

Definition 6.4.1. ***Standard form of linear programming problems.***

Linear programming problems look to optimise the value of an objective function $z \in \mathbb{R}$*, a linear function acting on a set of decision variables* $x_j \in \mathbb{R}$ *that themselves are subject to a finite number of linear constraints. In this thesis, a LPP in the following form, is said to be in standard form:*

$$\text{maximise}: \quad z = \underline{c}^{\mathrm{T}}\underline{x} \tag{6.10}$$

$$\text{subject to}: \quad \boldsymbol{A}\underline{x} = \underline{b} \tag{6.11}$$

$$\underline{x} \succcurlyeq \underline{0} \tag{6.12}$$

where $\underline{x}$ *is an* n *component vector of decision variables as* $(x_1, x_2, \ldots, x_n)^{\mathrm{T}}$*, the* cost vector $\underline{c}$ *a* n *component vector,* $\boldsymbol{A}$ *a* $m \times n$ *dimensional matrix, and* $\underline{b}$ *a* m *component vector that will be assumed to be* $\succcurlyeq \underline{0}$*.*
As $\underline{x} \in \mathbb{R}^n$*, and* $\underline{b} \in \mathbb{R}^m$*,* $\boldsymbol{A}$ *specifies a linear mapping from* $\mathbb{R}^n$ *to* $\mathbb{R}^m$*.*

We will, for our LPP set our equality constraints to be represented in the matrix form as in eq. (6.9), and choose an objective function dependent only on ε by choosing a cost vector $\underline{c}^{\mathrm{T}} = (1, 0, 0, 0, \ldots)$. As our decision variables correspond to a joint depolarising noise rate and convex coefficients, our choice of $\underline{x} = (\varepsilon, \eta_1, \eta_2, \ldots, \eta_{N^2})^{\mathrm{T}} \succcurlyeq \underline{0}$. Finally, the requirement that $\sum_{j=1}^{N^2} \eta_j = 1$ is encapsulated within the first of our equality constraints in eq. (6.11) (the first row of $\boldsymbol{A}$).

Our LPP as formulated here is still not in standard form; we require taking two additional steps:

1. In our LPP, we are looking to minimise our objective function $z = \underline{c}^{\mathrm{T}}\underline{x} = \varepsilon$, we can however use the objective function:

$$\text{maximise}: \quad z = \underline{c}^{\mathrm{T}}\underline{x} = (-1, 0, 0, 0, \ldots) \cdot (\varepsilon, \eta_1, \eta_2, \ldots, \eta_{N^2})^{\mathrm{T}} = -\varepsilon \tag{6.13}$$

2. If some element b_i of $\underline{b}$ is negative then we can simply multiply the i^{th} constraint by -1, that is, we can multiply the i^{th} row of $\boldsymbol{A}$ by -1.

6.5 Outline of the simplex method

6.5.1 Definitions and assumptions

Within this section we will outline how the simplex method can be used to solve LPPs. This was the specific method used later in the chapter. Readers interested in the results may wish to skip to section (6.6). Note that the theorems stated in this section can be found in the literature [65].

We will avoid discussing non-essential details by making a number of assumptions at the outset, starting with the following:

Assumption 6.5.1. ***Full rank assumption.***

For a LPP in standard form, with an equality constraint matrix $\boldsymbol{A}$ *of dimensional* $m \times n$*, it will be assumed that* $m < n$*, and that* $\text{rank}(\boldsymbol{A}) = m$*.*

We make the assumption (6.5.1) to avoid inconsistencies in the equality constraints, and instances where there are no sets of decision variables satisfying eq. (6.11). For our specific optimisation problem m is always 16, and the simplest[2] single particle state space satisfying octahedral symmetry and containing the entire Bloch sphere, would be the Bloch cube with a corresponding $\boldsymbol{A}$ that has $n = 64 + 1$ columns.

We will now proceed by defining the following types of solutions:

Definition 6.5.2. ***Solutions to systems of linear equalities.***

A solution *is defined to be any vector* $\underline{x} \in \mathbb{R}^n$ *satisfying a set of linear equalities* $\boldsymbol{A}\underline{x} = \underline{b}$ *as in eq.* (6.11)*.*

Definition 6.5.3. ***Feasible solutions and the feasible region of solutions***

Solutions to the LPP satisfying both the constraints eq.'s (6.11)*, and* (6.12)*, are called* feasible solutions*. The set of all possible feasible solutions is called the* feasible region*.*

Definition 6.5.4. ***The optimal objective function value and optimal solution.***

The optimal objective function value z_{opt} *is the least upper bound value the objective function* (6.10) *can take for the entire feasible region. An optimal solution* $\underline{x}_{opt}$ *to an LPP is the feasible solution that attains the optimal objective function value, that is* $z(\underline{x}_{opt}) \geq z(\underline{x})$ *for all feasible solutions* $\underline{x}$ *in the feasible region.*

We will now define a very important sub-set of feasible solutions that will later be shown to be key in finding the optimal solution to the LPP using the simplex method.

[2] With fewest vertices.

Definition 6.5.5. ***Basic solutions, degeneracy and basic feasible solutions.***

Suppose we have a system of linear equalities $\boldsymbol{A}\underline{x} = \underline{b}$*, with* $\text{rank}(\boldsymbol{A}) = m < n$*. We can form a smaller square matrix* $\boldsymbol{B}$ *out of* m *columns from* $\boldsymbol{A}$*, denoting the rectangular matrix formed from the remaining* $(n - m)$ *as* $\boldsymbol{R}$*. We are free to rearrange columns of* $\boldsymbol{A}$*, elements of* $\underline{x}$ *and* $\underline{b}$*, and can thus rewrite* $\boldsymbol{A}\underline{x} = \underline{b}$ *as:*

$$\boldsymbol{A}\underline{x} = (\boldsymbol{B} + \boldsymbol{R})\,\underline{x} = (\boldsymbol{B}\underline{x}_B + \boldsymbol{R}\underline{x}_R) = \underline{b} \tag{6.14}$$

Here $\underline{x} = (\underline{x}_B, \underline{x}_R)^{\mathrm{T}}$*, where* $\underline{x}_B$ *and* $\underline{x}_R$ *denote vectors of dimension* m *and* $(n - m)$ *respectively.*
If $\boldsymbol{B}$ *is a non-singular matrix and* $\underline{x}_R$ *is a* $(n - m)$ *element zero vector* $\underline{0}_R$*, then*

$$\underline{x}_B = \boldsymbol{B}^{-1}\underline{b} \tag{6.15}$$

and the solution $\underline{x} = (\underline{x}_B, \underline{0}_R)^{\mathrm{T}}$ *is called a basic solution of the system. The elements* x_{B_i}*,* $i \in \{1, 2, \ldots, m\}$*, forming* $\underline{x}_B$ *are termed basic variables, and* x_{R_l}*,* $l \in \{1, 2, \ldots, n - m\}$*, non-basic variables.*

If one or more elements of $\underline{x}_B$ *are zero then the basic solution* $\underline{x}$ *is degenerate.*

If basic solutions also satisfy the constraint in eq. (6.12) *then they are termed feasible basic solutions, and like basic solutions may also be degenerate.*

It is worth noting that the solutions achieving the optimum may not be unique. Degenerate feasible solutions can have the same objective function value. While it is possible to take degeneracy into consideration, we will make the following assumption to avoid unnecessary technical detail in this outline:

Assumption 6.5.6. All basic feasible solutions are non degenerate

It will be assumed that every basic feasible solution to the LPP is a non degenerate basic feasible solution.

Note that degeneracy was taken in account when performing the calculations.

6.5.2 The fundamental theorem of linear programming

We will now present the theorems central to linear programming as an optimisation method, as they provide structure to the feasible region of solutions.

Theorem 6.5.7. ***The fundamental theorem of linear programming.***

For an LPP in standard form, with an equality constraint matrix $\boldsymbol{A}$ *of dimensional* $m \times n$*, and rank* m*, we have that:*

1. *If there exists a feasible solution, then there exists a basic feasible solution.*
2. *If there exists an optimal feasible solution, then there exists an optimal basic feasible solution.*

In other words the optimal solution occurs at a basic feasible solution. The proof of theorem (6.5.7) can be found in the literature [65].

Theorem (6.5.7) transforms the task at hand into one where we are searching over the basic feasible solutions to the system for the one that is optimal. For an $\boldsymbol{A}$ matrix of dimensional $m \times n$, we have at most

$$\left(\begin{array}{c} n \\ m \end{array} \right) = \frac{n!}{m!(n-m)!} \tag{6.16}$$

ways of selecting sets of m columns, that is, we have this many basic solutions to eq. (6.11). It is worth noting that not all of these basic solutions may be feasible and so this must be taken to consideration.

Theorem 6.5.8. ***Convexity of feasible regions.***

The set of feasible solutions forms a convex set.

Proof. Let $\underline{x}_1$ and $\underline{x}_2$ be two feasible solutions to the linear constraints in eq.'s (6.11), and (6.12). Let $\underline{x}_3 = t\underline{x}_1 + (1-t))\underline{x}_2$, by linearity $\underline{x}_3$ is also a feasible solution to eq.'s (6.11), and (6.12). Any feasible solution can be expressed as a probabilistic mixture of other solutions, so the feasible region must be convex. □

Theorem 6.5.9. ***Basic feasible solutions and feasible region extrema.***

The basic feasible solutions (BFS) to a LPP correspond to extremal points of the associated feasible region, and vice-versa.

We can see from this theorem that we are geometrically searching over the vertices of the feasible region for optimal extrema for the vertex.

Most LPP optimisation methods in the literature [65] break the task down into two phases; the first to identify any basic feasible solution from which to start the search procedure, and second, the actual search procedure itself, looking to identify adjacent basic feasible solutions with larger objective function values. The second phase is then repeated until the optimal objective function value is arrived at.

We may simplify things here by skipping the first phase entirely as we already know a basic feasible solution - if we have full joint depolarising noise acting on the CZ gate then the output will be a maximally mixed product state. That is, we have our initial BFS by setting $\varepsilon = 1$ and choosing $\boldsymbol{B}$ to be formed from vectors formed from the first column of $\boldsymbol{A}$ and 15 e_j vectors that average over to give:

$$\left(\frac{1}{4}, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\right)^{\mathrm{T}}$$

Their corresponding n_j values can be weighted depending on their contribution so that they sum to 1. We can see it is a BFS as it satisfies eq.'s (6.8), and (6.11), (6.12). When we performed the calculation numerically, we let the solver choose its own starting basic feasible solution automatically.

The next section will look at the second phase, identifying better BFS, and the optimal solution.

6.5.3 Improving basic feasible solutions

We will look at the value of the objective function z where $\underline{x}$ is taken to be a basic or basic feasible solution. As a basic solution $\underline{x}$ can be expressed in terms of basic variables as $\underline{x} = (\underline{x}_B, \underline{0}_R)^{\mathrm{T}}$, the objective function thus becomes

$$z = \underline{c}^{\mathrm{T}}\underline{x} = \underline{c}_B^{\mathrm{T}}\underline{x}_B + \underline{c}_R^{\mathrm{T}}\underline{0}_R = \underline{c}_B^{\mathrm{T}}\underline{x}_B \tag{6.17}$$

where $\underline{c}_B^{\mathrm{T}}$ is called, the *reduced cost vector*, and is a m component vector of elements of $\underline{c}^{\mathrm{T}}$ corresponding to the non-zero basic variables of $\underline{x}$.

From assumption (6.5.1), as $\mathrm{rank}(\boldsymbol{A}) = m$ we can choose m columns from $\boldsymbol{A}$ to form a set of basis vectors. Any m element vector can therefore be expressed as a linear sum of this basis set of columns.

Suppose we selected as our basis the m columns forming the $\boldsymbol{B}$ matrix associated to some basic feasible solution (as in eq. (6.14)). We will denote the columns forming $\boldsymbol{B}$ as $\underline{\beta}_i$ where $i \in \{1, 2, \ldots, m\}$. A column vector $\underline{a}_j$ can be written in terms of our basis as:

$$\underline{a}_j = \sum_{i=1}^{m} y_{ij}\underline{\beta}_i \qquad \forall \mathrm{j} \in \{1, 2, \ldots, \mathrm{n}\} \tag{6.18}$$

Let $\underline{y}_j = (y_{1j}, y_{2j}, \ldots, y_{mj})^{\mathrm{T}}$, eq. (6.18) can be expressed as

$$\underline{a}_j = \boldsymbol{B}\underline{y}_j \rightarrow \underline{y}_j = \boldsymbol{B}^{-1}\underline{a}_j \tag{6.19}$$

Given a starting BFS as in eq. (6.15) we will obtain an improved BFS by replacing the r^{th}column of $\boldsymbol{B}$ denoted $\underline{\beta}_r$ with a suitable non-basic column of $\boldsymbol{A}$ - $\underline{a}_j$ to give a new BFS solution with a better objective function value. If the component y_{rj} associated

to $\underline{a}_j$ and $\underline{\beta}_r$ as in eq. (6.18) is not equal to zero then $\underline{a}_j$ is suitable to replace $\underline{\beta}_r$ and form a new column basis.

We will look at conditions on the choice of $\underline{a}_j$ and $\underline{\beta}_r$ to ensure feasibility:

From eq. (6.18) we have:

$$\underline{\beta}_r = \frac{1}{y_{rj}}\underline{a}_j - \sum_{\substack{i=1 \\ i\neq r}}^{m} \frac{y_{ij}}{y_{rj}}\underline{\beta}_i \qquad \text{where } y_{rj} \neq 0 \tag{6.20}$$

By replacing the column the basic variables vector $\underline{x}_B = (x_{B_1}, \ldots, x_{B_m})^{\mathrm{T}}$ associated to the basic feasible solution $\underline{x}$ transforms from

$$\boldsymbol{B}\underline{x}_B = \sum_{i=1}^{m} x_{B_i}\underline{\beta}_i = \underline{b} \tag{6.21}$$

to the following:

$$\left(\frac{x_{B_r}}{y_{rj}}\right)\underline{a}_j + \sum_{\substack{i=1 \\ i\neq r}}^{m} \left(x_{B_i} - x_{B_r}\left(\frac{y_{ij}}{y_{rj}}\right)\right)\underline{\beta}_i = \underline{b} \tag{6.22}$$

We have in effect arrived at a new basic solution $\underline{\tilde{x}}$ with decision variables $\tilde{x}_i$ as:

$$\tilde{x}_i = \begin{cases} x_{B_i} - x_{B_r}\left(\frac{y_{ij}}{y_{rj}}\right) & i = 1, 2, \ldots, m \\ \left(\frac{x_{B_r}}{y_{rj}}\right) & i = j \\ 0 & i = m+1, \ldots, n \end{cases} \tag{6.23}$$

Note that where $i = r$, $\tilde{x}_r = 0$. The new basic solution $\underline{\tilde{x}}$ takes the form:

$$\underline{\tilde{x}} = (\tilde{x}_1, \tilde{x}_2, \ldots, \tilde{x}_{r-1}, 0, \tilde{x}_{r+1}, \ldots, \tilde{x}_m, 0, \ldots, 0, \tilde{x}_j, 0, \ldots, 0)^{\mathrm{T}} \tag{6.24}$$

In order for $\underline{\tilde{x}}$ to be a valid basic feasible solution it must satisfy eq. (6.12), that is, we require:

$$\begin{cases} x_{B_i} - x_{B_r}\left(\frac{y_{ij}}{y_{rj}}\right) \geq 0 & i = 1, 2, \ldots, m \\ \left(\frac{x_{B_r}}{y_{rj}}\right) \geq 0 & i = j \end{cases} \tag{6.25}$$

We will therefore choose a column $\underline{\beta}_r$ to replace, by first choosing an $\underline{a}_j$ and $\underline{\beta}_r$and where $\left(\frac{x_{B_r}}{y_{rj}}\right) \geq 0$, and secondly where

$$\left(\frac{x_{B_r}}{y_{rj}}\right) = \min_{i=1,2,\ldots,m}\left\{\left(\frac{x_{B_i}}{y_{ij}}\right) \,:\, y_{ij} > 0\right\} \tag{6.26}$$

so as to ensure feasibility of the new solution.

We are free to rearranged the elements of $\underline{\tilde{x}}$ so the sub-vector $\underline{\tilde{x}}_B$ becomes

$$\underline{\tilde{x}}_B = (\tilde{x}_1, \tilde{x}_2, \ldots, \tilde{x}_{r-1}, \tilde{x}_j, \tilde{x}_{r+1}, \ldots, \tilde{x}_m)^{\mathrm{T}} \tag{6.27}$$

$$\tilde{x}_{B_i} = \begin{cases} x_{B_i} - x_{B_r}\left(\frac{y_{ij}}{y_{rj}}\right) & i = 1, 2, \ldots, m \;\&\; i \neq r \\ \left(\frac{x_{B_r}}{y_{rj}}\right) & i = r \end{cases} \tag{6.28}$$

Choosing columns with better objective function values

The transformation from one basic feasible solutions to another is pointless if the new basic feasible solution has an objective function value equal to or less than that of the original basic feasible solution. We will now show that we can place restrictions on the choices of $\underline{a}_j$ so as to ensure we move to a solution vertex with a better objective function values. Let us consider the objective function in eq. (6.17), with a new basic feasible solution it is transformed as

$$z = \underline{c}_B^{\mathrm{T}} \underline{x}_B \rightarrow \tilde{z} = \underline{\tilde{c}}_B^{\mathrm{T}} \underline{\tilde{x}}_B \tag{6.29}$$

Here for $i = 1, 2, \ldots, m$ and $i \neq r$, $\tilde{c}_{B_i} = c_{B_i}$, and for $i = r$, $\tilde{c}_{B_r} = c_j$. We can therefore expand $\tilde{z}$ as:

$$\begin{aligned} \tilde{z} &= \sum_{i=1}^{m} \tilde{c}_{B_i} \tilde{x}_{B_i} = \sum_{\substack{i=1 \\ i \neq r}}^{m} c_{B_i} \tilde{x}_{B_i} + c_j \tilde{x}_{B_r} \\ &= \sum_{\substack{i=1 \\ i \neq r}}^{m} c_{B_i} \left(x_{B_i} - x_{B_r} \left(\frac{y_{ij}}{y_{rj}} \right) \right) + c_j \left(\frac{x_{B_r}}{y_{rj}} \right) \\ &= \underline{c}_B^{\mathrm{T}} \underline{x}_B - \left(\frac{x_{B_r}}{y_{rj}} \right) \underline{c}_B^{\mathrm{T}} \underline{y}_j + c_j \left(\frac{x_{B_r}}{y_{rj}} \right) \\ &= z + \left(\frac{x_{B_r}}{y_{rj}} \right) (c_j - \underbrace{\underline{c}_B^{\mathrm{T}} \boldsymbol{B}^{-1} \underline{a}_j}_{z_j}) \end{aligned} \tag{6.30}$$

The numbers $c_j - z_j$, (where $z_j = \underline{c}_B^{\mathrm{T}} \boldsymbol{B}^{-1} \underline{a}_j$) are termed *reduced cost coefficients* with respect to the matrix $\boldsymbol{B}$, we see that $\tilde{z} > z$ only when $c_j - z_j > 0$. We will therefore choose a new column $\underline{a}_j$ such that the corresponding $c_j - z_j > 0$ and also the choice also satisfies $y_{rj} > 0$, $y_{ij} > 0$ to ensure feasibility from eq. (6.25). As $x_{B_r} > 0$ from the assumed non-degeneracy of our basic feasible solutions we have $\tilde{z} > z$.

Finally the following two theorems summarise the simplex method:

Theorem 6.5.10. ***The steps taken to arrive at better basic feasible solutions.***

Suppose we have a basic feasible solution $\underline{x} = (\underline{x}_B, \underline{0}_R)^{\mathrm{T}}$ *with a corresponding* $\boldsymbol{B}$ *matrix and objective function* z*. If (a) there exists a column from* $\boldsymbol{A}$ *not in* $\boldsymbol{B}$ *satisfying* $c_j - z_j > 0$ *with (b) at least one* $y_{ij} > 0$*, then it is possible to arrive at a new basic feasible solution with a larger objective function value* $\tilde{z}$ *by replacing one of the columns in* $\boldsymbol{B}$ *with* $\underline{a}_j$*. As we are only considering non-degenerate basic feasible solutions* $\tilde{z} > z$*.*

Theorem 6.5.11. ***The optimal basic feasible solution.***

Suppose we have a basic feasible solution $\underline{x} = \left(\underline{x}_B^{\mathrm{T}}, \underline{0}_R^{\mathrm{T}}\right)^{\mathrm{T}}$ *with objective function value* $\underline{z} = \underline{c}_B^{\mathrm{T}} \underline{x}_B^T$*. If* $z_j > c_j$ *for all columns* $\underline{a}_j$ *forming* $\boldsymbol{A}$*, then* $z > \tilde{z}$ *in every case and* $\underline{x}$ *is said to be optimal.*

6.6 Numerical solutions for truncated cubes

In this section we will use computational implementations of the simplex method to calculate the amount of joint depolarising noise required for CZ-separability. The single particle state space $\mathcal{S}$ will be set to $TRUNC(c)$. We remind the reader that we are interested in the relationship between c and the amount of noise needed for separability, in particular if there exists some c value that can lead to lower amounts of noise.

For each value of c, and chosen product input state, we have a new LPP. In order to calculate a numerical value for $\lambda_{CZ_\varepsilon, TRUNC(c)}$ we follow the procedure:

1. Select a numerical value for c, and a product input state from eq. (6.2).
2. Perform the necessary operations on the product input states to construct the standard form matrices and vectors for the LPP.
3. Input the matrices and vectors into a computational implementation of the simplex method.
4. The numerical solver outputs for our chosen input state and c value, an optimal solution to **that particular** LPP. This solution is given in terms of a list of optimal basic variables $\underline{x}_B$ and associated $\boldsymbol{B}$ matrix columns. We will label the optimal basic solution and basic matrix obtained from the numerics and associated with input state k as $\underline{x}_{B_{num,\,k}}$ and $\boldsymbol{B}_{num,\,k}$ respectively. The vector $\underline{x}_{B_{num,\,k}}$, $\boldsymbol{B}_{num,\,k}$ chosen c value, and product input state ϱ_k are then recorded. Most importantly, it is worth remembering that the first element of the basic solution is ε our joint depolarising noise rate, the rest are convex coefficients.
5. Steps $1 - 4$ are repeated for the same c value, but with other input states from (6.2).

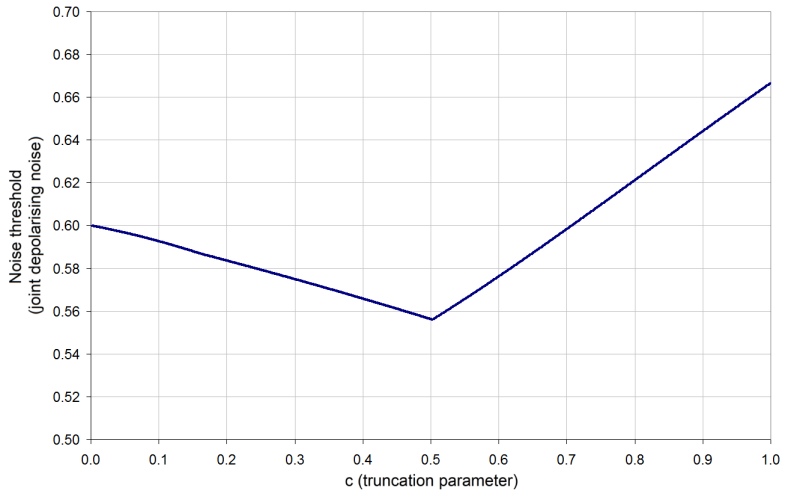


Figure 6.3: Plot of truncated cube state spaces with varying c values plotted against the amount of joint depolarising noise to ensure CZ gate separability with respect to the $TRUNC)(c)$ state space.

---

6. Finally the ε values for each input state are compared and the largest ε value as is selected as being the $\lambda_{CZ_\varepsilon,TRUNC(c)}$ value, i.e. the least amount of noise required to ensure CZ separability for all possible product input states for that value of c.

Figure (6.3) shows a numerical results plot of the amount of joint depolarising noise needed for separability with the truncation parameter c. We see from the numerics that for the lowest value of $\lambda_{CZ_\varepsilon,TRUNC(c)}$ is approximately 0.556 around $c = 0.5$.

6.6.1 Determining expressions linking c and $\lambda_{CZ_\varepsilon,TRUNC(c)}$

For the remainder of the chapter we will improved on the numerical result by showing how we can arrive at a symbolic expression relating c to $\lambda(CZ_\varepsilon, TRUNC(c))$. We remind the reader that each input state ϱ_k from eq. (6.2) requires different amounts of noise to ensure CZ output separability and that the input ϱ_k requiring the most noise would change depending on the value of c

Rather than performing steps 5 and 6 to in section (6.6) determine a numerical value of $\lambda_{CZ_\varepsilon,TRUNC(c)}$, the recorded numerical vector $\underline{x}_{B_{num,\,k}}$ and $\boldsymbol{B}_{num,\,k}$ matrix associated with each product input state can be used to determine symbolic necessary bounds like:

$$\varepsilon \geq \Gamma_{\varrho_k}(c) \qquad \text{where } \mathrm{c}' \leq \mathrm{c} \leq \mathrm{c}'' \tag{6.31}$$

Where for a certain range of values of c, Γ is a function of c, and is associated to ϱ_k. As usual, ε is the amount of noise needed to ensure the CZ gate output from the ϱ_k input remains separable.

Bringing together each function Γ_{ϱ_k} we have:

$$\lambda(CZ_\varepsilon, TRUNC(c)) = \max\left(\Gamma_{\varrho_1}, \Gamma_{\varrho_2}, \Gamma_{\varrho_3}, \Gamma_{\varrho_4}, \Gamma_{\varrho_5}, \Gamma_{\varrho_6}\right) \tag{6.32}$$

We will explain how we can arrive at each Γ_{ϱ_k} function as following: To every ϱ_k we can associate an *optimal face* of the convex set $TRUNC(c)^{\otimes 2}$ corresponding to the face at which the output state enters $TRUNC(c)^{\otimes 2}$ given enough joint depolarising noise. The optimal face can be viewed as a subset of some 14 dimensional hyperplane in the 15 product Bloch space. Every extrema of $TRUNC(c)^{\otimes 2}$ must lie either on this hyperplane or on one side of it.

Using the results for step 4 in section (6.6) for a particular ϱ_k and truncation parameter c, it is possible to guess what the corresponding optimal face is for ϱ_k and a range of truncation parameter values c, and in doing so obtain the associated Γ_{ϱ_k} function.

For a particular ϱ_k we identify the columns of $\boldsymbol{B}_{num,\,k}$ with vertices of $TRUNC(c)^{\otimes 2}$ for a range of values of c. As we want to relate ε to to some function dependent on a symbolic c, we replace the numerical entries of the columns of $\boldsymbol{B}_{num,\,k}$ with symbols. For example if we have $\underline{x}_{B_{num,\,k}}$ and $\boldsymbol{B}_{num,\,k}$ generated for some input state ϱ_k with truncation parameter $c = 0.2$, we will replace all "0.2" entries in $\boldsymbol{B}_{num,\,k}$ with the symbol "c", and all entries "0.4" with "c^2", thereby constructing a symbolic version of $\boldsymbol{B}_{num,\,k}$, that we will call $\boldsymbol{B}_{s,\,k}$.

In order to specify the face we need to determine what the normal $\underline{n}$ to the hyperplane containing the optimal face is. To find $\underline{n}$ we take $\boldsymbol{B}_{s,\,k}$ and perform the following steps:

As we are picturing this optimal face geometrically we are only interested in the product Bloch vector elements from each column of $\boldsymbol{B}_{s,\,k}$. We will therefore start modifying $\boldsymbol{B}_{s,\,k}$ by removing its first row of elements. The first column will also be removed, as the first column of $\boldsymbol{B}_{s,\,k}$ corresponds to $CZ(\varrho) - \frac{\mathbb{1}^{\otimes 2}}{4}$ which is not a vertex of $TRUNC(c)^{\otimes 2}$.

The remaining 15 columns correspond to Bloch vectors to vertices of $TRUNC(c)^{\otimes 2}$. We will select the first of these (denoted $\underline{\zeta}$) and subtract it from the others thereby giving 14 vectors on the hyperplane containing the optimal face. These vectors can then be used to construct the rectangular matrix $\boldsymbol{D}_{s,\,k}$ of dimension 15×14. We are looking for solutions to the homogeneous system of linear inequalities

$$\boldsymbol{D}_{s,\,k}^{\mathrm{T}}\underline{n} = \underline{0} \tag{6.33}$$

In other words we are looking to find the nullspace of $\boldsymbol{D}_{s,\,k}^{\mathrm{T}}$, which would contain the vector normal to the hyperplane. This can be done easily by a symbolic mathematics software package using guassian elimination. Let $(1-\varepsilon)\underline{r}_{CZ(\varrho_k)}$ correspond to the product Bloch vector associated to the noisy CZ output state for ϱ_k input. The normal can then be used to arrive at a symbolic expression for ε by taking the dot product of

the normal with $((1-\varepsilon)\underline{r}_{CZ(\varrho_i)} - \underline{\zeta})$:

$$\underline{n} \cdot ((1-\varepsilon)\underline{r}_{CZ(\varrho_k)} - \underline{\zeta}) \geq 0 \tag{6.34}$$

Finally this expression can be rearranged to give a expressions as eq. (6.31) for ϱ_k and range of c values.

6.6.2 Example: calculating Γ_{ϱ_1}

To clarify this process of calculating Γ_{ϱ_k} functions we will look at an example for one input; suppose we choose $c = 0.7$, and have a product input state ϱ_1 represented by the Pauli table (left) that under a noisy CZ gate transforms as follows (right):

$$\frac{1}{4}\begin{Bmatrix} 1 & 0.7 & 1 & 1 \\ 0.7 & 0.49 & 0.7 & 0.7 \\ 1 & 0.7 & 1 & 1 \\ 1 & 0.7 & 1 & 1 \end{Bmatrix} \overset{CZ}{\rightarrow} \frac{1}{4}\begin{Bmatrix} 1 & 0.7 & 1 & 1 \\ 0.7 & 1 & -0.7 & 0.7 \\ 1 & -0.7 & 0.49 & 1 \\ 1 & 0.7 & 1 & 1 \end{Bmatrix} \tag{6.35}$$

After constructing the necessary matrices we submit the LPP to the simplex solver which outputs a basic variable vector $\underline{x}_{B_{num,\,1}}$ (rounded to 4 decimal places):

$$\begin{aligned}(&0.5833,\ 0.2208,\ 0.0708,\ 0.1100,\ 0.0001,\ 0.1343,\ 0.0429,\ 0.0456,\\ &0.0858,\ \underbrace{0.0000}_{\approx 10^{-15}},\ 0.0267,\ 0.0687,\ 0.0871,\ 0.0441,\ 0.0229,\ 0.0402)^{\mathrm{T}}\end{aligned} \tag{6.36}$$

It is important to take note of the 10^{th} element of $\underline{x}_{B_{num,\,1}}$ which is not equal exactly to zero. We see that for this choice of input state and value of c the amount of noise needed for separability is approximately 58.3%, the remaining 15 elements sum to 1 as required by convex coefficients. The matrix $\boldsymbol{B}_{num,\,1}$ is given in eq. (9.14), in section (9.7) in the appendix.

Before proceeding we will replace the entries of the modified $\boldsymbol{B}_{num,\,1}$ with symbols i.e. entries like "0.7"$\rightarrow c$ and "0.49"$\rightarrow c^2$ to give $\boldsymbol{B}_{s,\,1}$. As we are only interested in the product Bloch vectors components of each column of $\boldsymbol{B}_{s,\,1}$ corresponding to vertices of $TRUNC(c)^{\otimes 2}$, we will remove the first row comprising of only 0, and 1's, and the first column as it does not correspond to a vertex of the optimal face.

We will take the first column $\underline{\zeta}$ of the modified $\boldsymbol{B}_{s,\,1}$ and subtract it from the remaining rows to arrive at 14 vector directions on the hyperplane containing our newly symbolic optimal face. These columns are used to construct the matrix $\boldsymbol{D}_{s,\,1}$ as in eq. (9.15) found in section (9.7) in the appendix. The normal to the hyperplane is given by determining the null space of $\boldsymbol{D}_{s,\,1}^{\mathrm{T}}$:

$$\underline{n} = \mathrm{nullspace}(\boldsymbol{D}_{s,\,1}^{\mathrm{T}}) = (0,-1,0,-1,0,1,0,0,0,0,0,0,0,0,0)^{\mathrm{T}} \tag{6.37}$$

Finally we can calculate eq. (6.34) for this product input state to arrive at Γ_{ϱ_1}, valid where approximately $0.47 < c \leq 1$:

$$\varepsilon \geq \Gamma_{\varrho_1} = \frac{2c}{1+2c} \tag{6.38}$$

Repeating this procedure for the other product states and values of c we arrive at the functions:

$$\Gamma_{\varrho_1} = \Gamma_{\varrho_4} = \begin{cases} \frac{c^2-1}{c^2-2} & \text{for } 0 \leq c \lesssim 0.376 \\ \frac{2c}{1+2c} & \text{for } 0.46 \lesssim c \leq 1 \end{cases} \tag{6.39}$$

$$\Gamma_{\varrho_2} = \frac{1+c^2}{2+c^2} \quad \text{for } 0 \leq c \leq 1 \tag{6.40}$$

$$\Gamma_{\varrho_3} = \Gamma_{\varrho_5} = \begin{cases} \frac{3-2c+c^2}{5-c} & \text{for } 0 \leq c \lesssim c' \\ \frac{1+c^2}{2+c^2} & \text{for } c' \lesssim c \leq 1 \end{cases} \quad \text{where } c' \approx 0.326 \tag{6.41}$$

$$\Gamma_{\varrho_6} = \begin{cases} \frac{6-2c^2}{10+c-c^2} & \text{for } 0 \leq c \lesssim k & \text{where } k \approx 0.161 \\ \frac{4\left(3+2c+c^2\right)}{20+16c+9c^2+2c^3+c^4} & \text{for } k \lesssim c \lesssim k' & \text{where } k' \approx 0.507 \\ \frac{5+c+c^2-c^3}{9+2c+c^2-c^3} & \text{for } k' \lesssim c \lesssim k'' & \text{where } k'' \approx 0.624 \\ \frac{2c}{1+2c} & \text{for } k'' \lesssim c \leq 1 & \end{cases} \tag{6.42}$$

For Γ_{ϱ_1} and Γ_{ϱ_4} in eq. (6.39), it appears that optimal face varies greatly for different c values between $0.376 < c < 0.46$, for this reason we have not included any expressions for the functions for those c values. From our numerical results we observed that the amount of noise needed for CZ output separability for ϱ_1, and ϱ_4 inputs, for these c values was low, and therefore we could confidently disregard the necessary bounds from this case. It is worth noting that Γ_{ϱ_2} in eq. (6.40) matches exactly the necessary bound obtained in eq. (6.4) corresponding to positivity of Pauli measurements. Figures (6.4) and (6.5) show plots of these necessary bounds plotted against numerical results for their associated product inputs.

Let us now consider eq. (6.32); we see that for $0 \leq c \leq 1$ that the key necessary bounds are from the functions Γ_{ϱ_6}, and, Γ_{ϱ_1} or Γ_{ϱ_2}, that is it is necessary that:

$$\lambda_{CZ_\varepsilon, TRUNC(c)} = \begin{cases} \frac{6-2c^2}{10+c-c^2} & \text{for } 0 \leq c \leq k & \text{where } k \approx 0.161 \\ \frac{4\left(3+2c+c^2\right)}{20+16c+9c^2+2c^3+c^4} & \text{for } k \leq c \leq l & \text{where } l \approx 0.502 \\ \frac{1+c^2}{2+c^2} & \text{for } l \leq c \leq 1 & \end{cases} \tag{6.43}$$

The minimum value of eq. (6.43) is ≈ 0.55601, and occurs at $l \approx 0.502298$, where the functions $\frac{4\left(3+2c+c^2\right)}{20+16c+9c^2+2c^3+c^4}$ and $\frac{1+c^2}{2+c^2}$ intersect. While we cannot exactly determine the

convex coefficients of the optimal solution at $c = l$, it is possible to calculate these coefficients for c values near l for each product input state, thereby showing that noise rates for these values of c are sufficient, as will be discussed in the next subsection.

6.6.3 Calculating exact convex coefficients

We wish to arrive at a symbolic version of $\underline{x}_B$ (denoted $\underline{x}_{B_{s,\,k}}$) for some particular c and product input state, so we can obtain exact expressions for the convex coefficients forming $\underline{x}_B$.

Just as we constructed $\boldsymbol{B}_{s,\,k}$, we can construct a symbolic version of the $\underline{b}$ equality vector from eq. (6.11) by replacing the numerical entries with symbols. For example, suppose $c = 0.7$, the associated $\underline{b}$ to the product input state ϱ_1 is:

$$(1, 0.7, 1, 1, 0.7, 1, -0.7, 0.7, 1, -0.7, 0.49, 1, 1, 0.7, 1, 1)^{\mathrm{T}} \tag{6.44}$$

Replacing entries "0.7"$\rightarrow c$ and "0.49"$\rightarrow c^2$, we obtain the symbolic version denoted $\underline{b}_{s,\,1}$:

$$\Big(1, c, 1, 1, c, 1, -c, c, 1, -c, c^2, 1, 1, c, 1, 1\Big)^{\mathrm{T}} \tag{6.45}$$

The symbolic matrix $\boldsymbol{B}_{s,\,1}$ and $\underline{b}_{s,\,1}$ must satisfy the following:

$$\boldsymbol{B}_{s,\,1}\underline{x}_{B_{s,\,1}} = \underline{b}_{s,\,1} \tag{6.46}$$

As $\boldsymbol{B}_{s,\,1}$ is a square matrix we may invert it arriving at an expression for $\underline{x}_{B_{s,\,1}}$:

$$\underline{x}_{B_{s,\,1}} = \boldsymbol{B}_{s,\,1}^{-1}\underline{b}_{s,\,1} \tag{6.47}$$

Hence for the specific value of c, the last 15 elements of $\underline{x}_{B_{s,\,1}}$ correspond exactly to the convex coefficients. In order to show sufficiency we must arrive at exact expressions for the convex coefficients for each product input.

Section (9.8) in the appendix shows $\boldsymbol{B}_{s,\,k}$ matrices and convex coefficient vectors constructed for each product input where $c = 0.5$.

It is important to note that the techniques used in this chapter can be extended to other shapes $\mathcal{S}$.

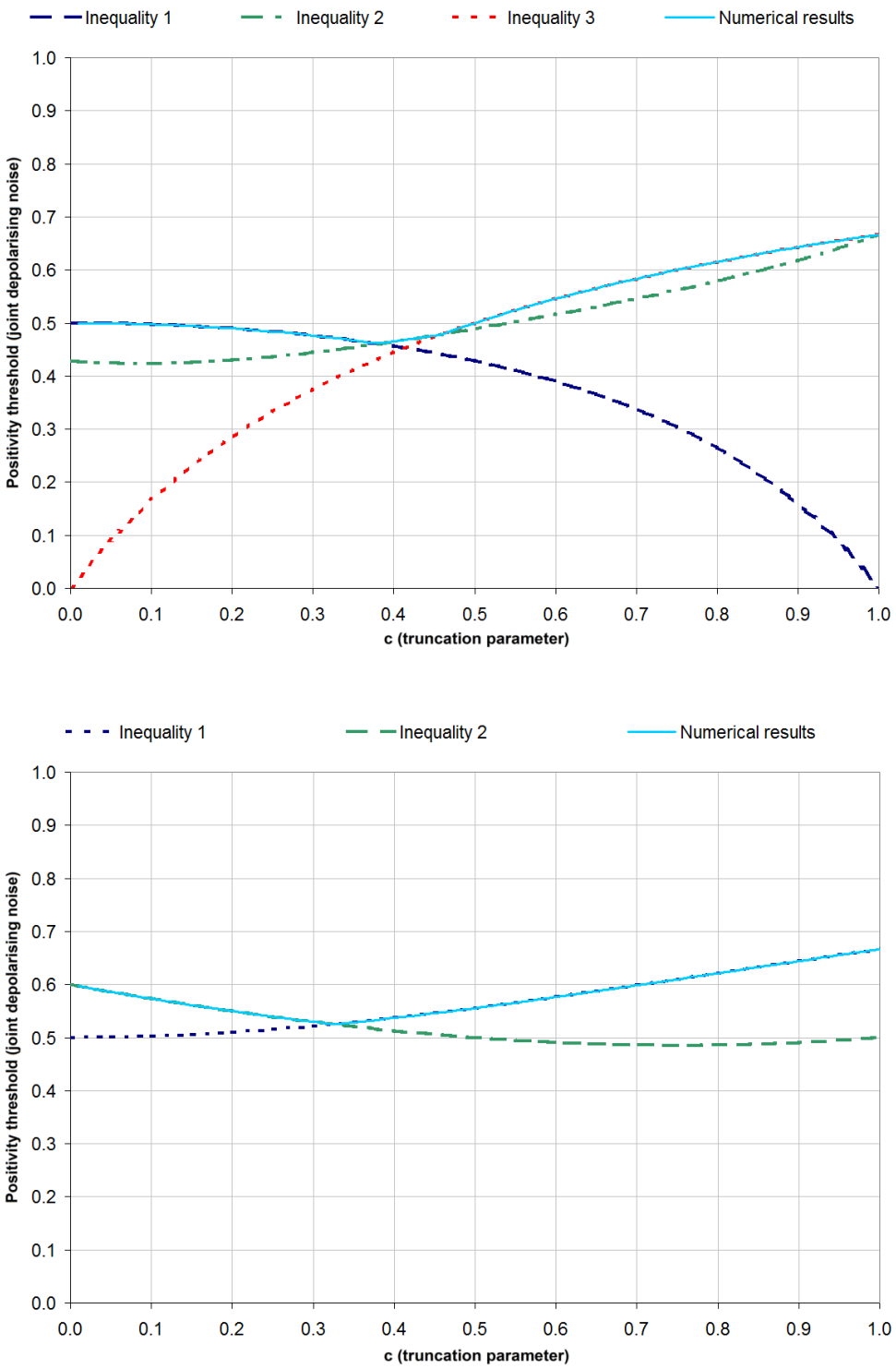


Figure 6.4: Plot optimal face necessary bounds and numerical results of truncated cube state spaces. Results for ϱ_1 and ϱ_4 (top). Here the non-solid lines (dark blue, green, and red) correspond to the necessary bounds from eq. (6.39), the solid line (light blue) corresponds to the numerical results for the amount of joint depolarising noise needed to ensure CZ output separability for ϱ_1 or ϱ_4 input. The exact necessary bound characterising the region of approximately $0.376 < c < 0.46$ is unclear. Results for ϱ_3 and ϱ_5 (bottom). The numerical results match the two necessary bounds from eq. (6.41) exactly.

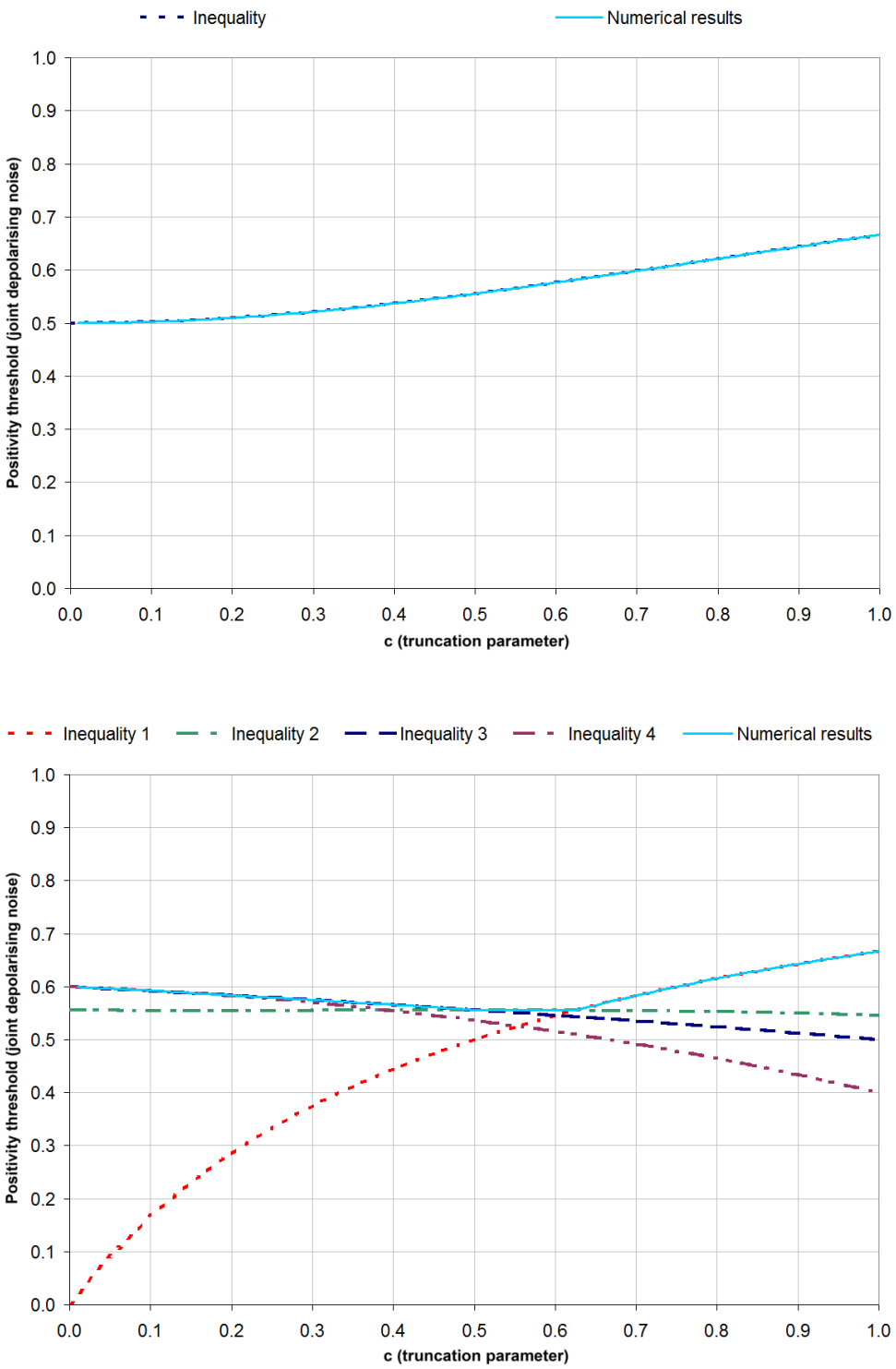


Figure 6.5: Results for ϱ_2 (top). Here the numerical results and necessary bounds in eq. (6.40) match exactly. Results for ϱ_6 (bottom). The numerical results match the three necessary bounds in eq. (6.42) exactly.

6.7 Summary of Chapter (6)

In this chapter we looked at the truncated cube state space - our best candidate for a choice of $\mathcal{S}$. While we believe it not to be optimal, it required the least amount of joint depolarising noise to ensure CZ output separability with no rescaling, over all state spaces we looked at. We started the chapter by defining the state space, parametrising it in terms of its truncation, and parameter c. We then specified the six input states for the truncated cube from the symmetry arguments. Relating back to the necessary norms in Chapter (7) we came up with a bound for cube positivity as a function of the truncation parameter c.

The next section looked at determining values of $\lambda_{CZ_\varepsilon,\mathcal{S}}$, for a varying value of c, we went on and showed that the problem could be re-expressed as a linear programming problem, solvable using the simplex method as outlined in the following section. We then detailed the procedure by which we could obtain numerical values for $\lambda_{CZ_\varepsilon,\mathcal{S}}$ for specific truncation values.

Using the numerical results we were able to make educated guesses to obtain symbolic relationships between the noise needed for separability for a specific input, and the truncation parameter. This method was then used again to obtain a convex decomposition of the sufficiently noisy CZ output state, and symbolic expressions for the convex coefficients. Although we did not obtain an analytic solution, we did show that our result was sufficient, and obtained an exact value of $\lambda_{CZ_\varepsilon,\mathcal{S}}$, of approximately 0.55601 when $c \approx 0.502298$. This is lower than the approximately 0.667 required for Bloch sphere and cube separability.

7 Positivity bounds

7.1 Introduction

In the previous chapter we looked at the truncated cube as a choice of $\mathcal{S}$. This set improved upon some of the previous chapters results in that for certain amounts of truncation it was found we needed less joint depolarising noise for $\mathcal{S}$-separability of the CZ gate.

In this chapter we will look to see if we can obtain bounds that would allow us narrow the choices of $\mathcal{S}$ corresponding to the lowest joint depolarising noise threshold values. In particular we are interested in seeing if state spaces smaller/larger in certain directions allow for reductions in the threshold value. To this end we will look at state spaces $\mathcal{S}$ satisfying $\mathcal{V} \subseteq \mathcal{S} \subseteq \mathcal{W}$, where $\mathcal{V}$ is some subset, possibly within the Bloch sphere, and $\mathcal{W}$ is a superset, possibly containing all quantum qubit states. By choosing $\mathcal{V}$ and $\mathcal{W}$ with a variety of specific geometries we will look to try and identify key directions.

As we will see in the next section we will specify sets $\mathcal{V}$ and $\mathcal{W}$ by considering norms on the Bloch vectors of the elements of $\mathcal{S}$. The positivity of operators with respect to $\mathcal{W}$ can be used to construct necessary bounds on the noise threshold. In the second half of the chapter we will arrive at another set of necessary bounds by looking at the structure of Pauli tables representing states. Finally we combine all previously obtained bounds to specify a region of noise threshold values containing the threshold value for an optimal $\mathcal{S}$.

Like in Chapter (5) we will interpret the shrinking and enlargening of state spaces with noise on the preparation/measurement.

7.2 Specifying norms

For a positive operatorϱ to be in a given convex set $\mathcal{S}^{\otimes 2}$, it is necessary that it satisfies the following relation:

$$\mathrm{tr}\left[\varrho\sigma\right] \geq 0 \qquad \forall\sigma \in \mathcal{S}^{*\otimes 2} \tag{7.1}$$

Here $\mathcal{S}^*$ denotes the dual set of operators to $\mathcal{S}$ (see section 4.2). We will use the following shorthand notation to represent these inequalities:

$$\mathrm{tr}\left[\varrho\mathcal{S}^{*\otimes 2}\right] \geq 0 \tag{7.2}$$

If ϱ is an output state from the noisy CZ gate then eq. (7.2) forms a necessary test on whether ϱ is an element of $\mathcal{S}^{\otimes 2}$, that is, a necessary test on if ϱ is $\mathcal{S}$-separable. This works in an analogous way to measurement probability positivity, and like measurement positivity does not guarantee separability with respect to $\mathcal{S}$.

As before we can express a positive operator $\varrho \in \mathcal{S}$, as $\varrho = \frac{1}{2}(\mathbb{1} + \underline{r} \cdot \underline{\sigma})$, where $\underline{\sigma}$ is a vector formed of Pauli operators as (X, Y, Z). In this chapter we will, as a shorthand, sometime express operators in terms of their $\underline{r}$ vectors, i.e. in statements like $\underline{r} \in \mathcal{S}$. In principle there are infinite choices for $\mathcal{S}$, and it is impossible to characterise them completely. We will instead characterise $\mathcal{S}$ sets in terms of norms on the $\underline{r}$ vectors. The reason for using them is they are strongly related to the symmetries and the constraints. We will consider the following norms on the absolute values on a number of components:

$$\begin{aligned}
&l_\infty \text{ norm} && \|\underline{r}\|_\infty := \max\{|x|, |y|, |z|\} \\
&\text{Ky-fan 2-norm} && \|\underline{r}\|_k := \max\{|x| + |y|, |y| + |z|, |x| + |z|\} \\
&l_1 \text{ norm} && \|\underline{r}\|_1 := |x| + |y| + |z| \\
&l_2 \text{ norm} && \|\underline{r}\|_2 := \sqrt{x^2 + y^2 + z^2}
\end{aligned} \tag{7.3}$$

The "longest" $\underline{r}$ vector lengths of operators in $\mathcal{S}$ are denoted by:

$$\begin{aligned}
\alpha(\mathcal{S}) &:= \max\{\|\underline{r}\|_\infty \mid \underline{r} \in \mathcal{S}\} \\
\beta(\mathcal{S}) &:= \max\{\|\underline{r}\|_k \mid \underline{r} \in \mathcal{S}\} \\
\gamma(\mathcal{S}) &:= \max\{\|\underline{r}\|_1 \mid \underline{r} \in \mathcal{S}\} \\
\delta(\mathcal{S}) &:= \max\{\|\underline{r}\|_2 \mid \underline{r} \in \mathcal{S}\}
\end{aligned} \tag{7.4}$$

For example, let $\mathcal{R}$ be the truncated cube state space (see section (6.2)), with truncation parameter $c = 0.5$. We have norm values$\alpha(\mathcal{S}) = 1$, $\beta(\mathcal{S}) = 2$, $\gamma(\mathcal{S}) = 2 + c$, $\delta(\mathcal{S}) = \sqrt{2 + c^2}$ given for $\mathcal{R}$.

Just as there is infinite freedom in choosing $\mathcal{S}$, there is also considerable freedom in choosing $\mathcal{V}$ and $\mathcal{W}$. The norms in the list in eq. (7.4) (above), in addition to the octahedral symmetries mentioned in section (4.5.2), and convexity of $\mathcal{S}$ guarantee the existence of various subsets and supersets of $\mathcal{S}$ as will be shown in the following sections. These subsets and supersets give an indication of the possible choices of $\mathcal{V}$ and $\mathcal{W}$ and allow us to form necessary bounds on the amount of noise needed for $\mathcal{S}$-separability.

7.3 Specifying subsets and supersets from norms

In this section we will specify sets in terms of the norms listed in eq. (7.4), using them to construct subsets and supersets of those sets. These supersets and subsets will be used in the following way, we will begin by first specifying an arbitrary convex operator

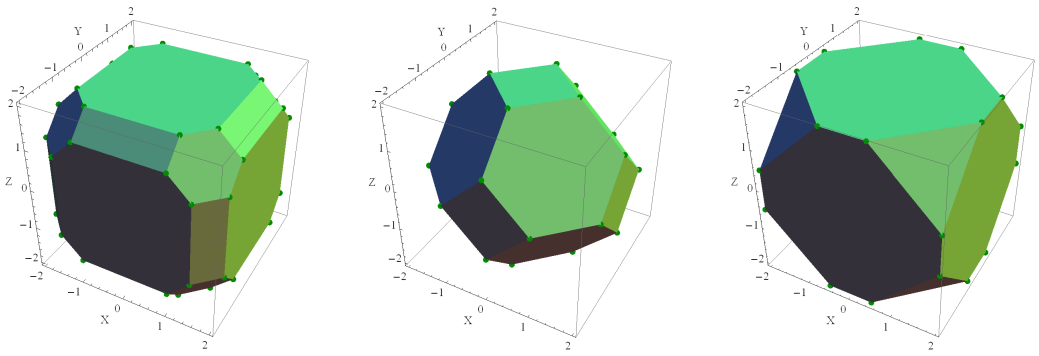


Figure 7.1: Representation of different $\mathcal{T}$ sets. (Left) The $\mathcal{T}$ set generated for ϱ represented by the $\underline{r}$ vector (1, 1.5, 2), (Middle) (2, 1, 0), (Right) (2, 0.5, 2).

set $\mathcal{T}$ that has octahedral symmetry, before detailing how we can characterise the size of $\mathcal{T}$ in terms of the norms. We will specify $\mathcal{T}$ as the convex hull of operators arrived at by performing octahedral symmetry operations on ϱ, where ϱ is some positive operator. Figure (7.1) presents three $\mathcal{T}$ sets generated through this method for different starting ϱ operators. Here $\mathcal{T}$ is visualised geometrically in the familiar Bloch sphere picture. The operator $\frac{\mathbb{1}}{2}$ corresponding to the vector (0, 0, 0) at the center of the shape will be sometime be referred to as the origin.

Guaranteed subsets of $\mathcal{T}$ will be specified by looking at the faces of the geometric representation of $\mathcal{T}$ and then forming polytopes of operators using the points at the centers of these faces. Supersets will be arrived at by looking at the sets of operators with vectors satisfying the various norms listed in eq. (7.4).

7.3.1 Subset and supersets specified by $\alpha(\cdot)$

A set $\mathcal{T}$ characterised by the norm $\alpha = \alpha(\mathcal{T})$ must contain an octahedron subset - a convex hull of extrema with the following Bloch vectors:

$$(\pm\alpha, 0, 0), (0, \pm\alpha, 0), (0, 0, \pm\alpha) \tag{7.5}$$

The octahedron specified by $\alpha(\mathcal{T})$ will be denoted by $OCT(\alpha)$. It is important to note that the vertices of this octahedron point in the directions of the Pauli eigenstates.

We can specify a superset by looking at all of the positive operators with vectors $\underline{r}$ satisfying the inequalities $\|\underline{r}\| \leq \alpha(\mathcal{T})$. It is clear that the set $\mathcal{T}$ must be a contained in a superset denoted $CUBE(\alpha)$, where $CUBE(\alpha)$ is a Bloch cube state space characterised by the norm value $\alpha = \alpha(\mathcal{T})$, with extremal vertices of l_∞ norm length $\alpha = \alpha(\mathcal{T})$. Figure (7.2) presents an plot of $OCT(\alpha)$ and $CUBE(\alpha)$ generated for the Bloch vector (1, 1.5, 2).

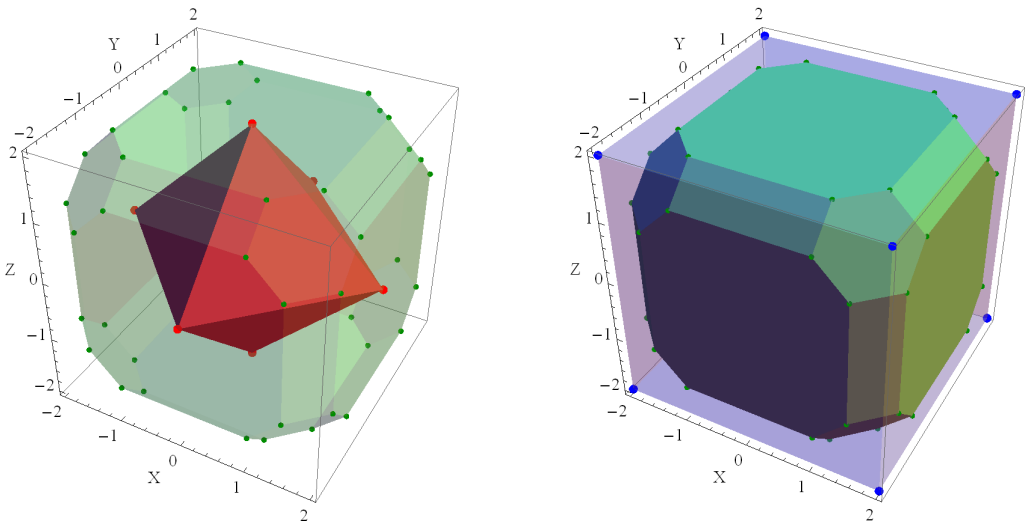


Figure 7.2: Representation of a subset and superset induced by $\alpha(\mathcal{T})$. $\mathcal{T}$ is generated for the vector $\underline{r} = (1, 1.5, 2)$, (Left) shows $\mathcal{T}$ containing the subset $OCT(\alpha)$ (Right) $\mathcal{T}$ is contained in the superset $CUBE(\alpha)$.

7.3.2 Subset and supersets specified by $\gamma(\cdot)$

Let us consider a ray from the origin in the direction of (1, 1, 1), that is, in the direction of one of the T-type magic states. This vector will intersect with a face of $\mathcal{T}$ and if we average over the vertices forming this face we end up with a center point of the face. Performing octahedral symmetry operations this the point gives a set of eight vectors, the convex hull of which specifies a cube contained within $\mathcal{T}$. The vectors of this cube have Bloch vectors satisfying:

$$|x| = |y| = |z| = \frac{\gamma(\mathcal{T})}{3} \tag{7.6}$$

where γ corresponds to the maximum l_1 norm length in eq. (7.3). We denote this internal cube as $CUBE(\frac{\gamma}{3})$. The set of all operators associated with vectors $\underline{r}$ satisfying $\|\underline{r}\| \leq \gamma(\mathcal{T})$ form an octahedron containing $\mathcal{T}$. We will denote this superset as $OCT(\gamma)$. Figure (7.2) presents a plot of $CUBE(\frac{\gamma}{3})$ and $OCT(\gamma)$ for a set $\mathcal{T}$ generated with the vector (1, 1.5, 2).

7.3.3 Subset and supersets specified by $\beta(\cdot)$

The set $\mathcal{T}$ contains at least one point with an $\underline{r}$ vector as $(c\beta, (1-c)\beta, z)$ where $c \in [0, 1]$. Hence by symmetry and convexity it contains $(\frac{\beta}{2}, \frac{\beta}{2}, 0)$. Then by symmetry it contains the other points. The convex hull of these points forms a subset of $\mathcal{T}$ - a cubeoctahedron. If the $\underline{r}$ vectors of $\mathcal{T}$ have a maximal Ky-fan length of $\beta = \beta(\mathcal{T})$ then this cubeoctahedron subset will be denoted $CUBOCT(\frac{\beta}{2})$ and will have vertices

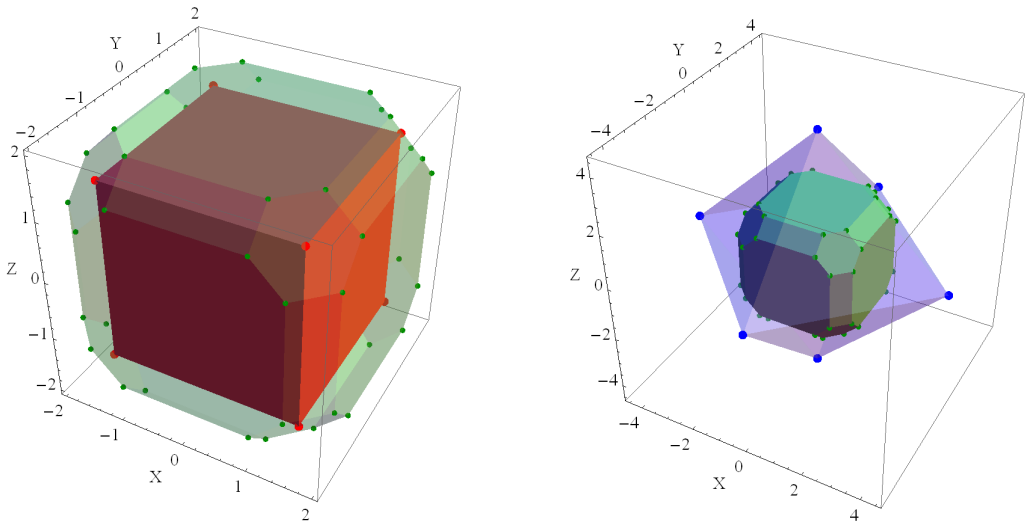


Figure 7.3: Representation of a subset and superset induced by $\gamma(\mathcal{T})$. $\mathcal{T}$ is generated for the Bloch vector $r = (1, 1.5, 2)$, (Left) shows $\mathcal{T}$ containing the subset $CUBE(\frac{\gamma}{3})$ (Right) $\mathcal{T}$ is contained in the superset $OCT(\gamma)$.

---

represented by operators with $\underline{r}$ vectors satisfying:

$$|x| = |y| = \frac{\beta}{2},\ z = 0 \qquad |x| = |z| = \frac{\beta}{2},\ y = 0 \qquad |y| = |z| = \frac{\beta}{2},\ x = 0 \tag{7.7}$$

The set of all operators associated with $\underline{r}$ vectors satisfying $\|\underline{r}\| \leq \beta(\mathcal{T})$ form a rhombic dodecahedron containing $\mathcal{T}$. We will denote this superset as $RHOM(\beta)$. Figure (7.4) presents an plot of $CUBOCT(\frac{\beta}{2})$ and $RHOM(\beta)$ generated for the $\underline{r}$ vector (1, 1.5, 2).

7.3.4 Characterisation through $\delta(\cdot)$ and summary

We can finally characterise $\mathcal{T}$ using the Euclidean norm as follows; whilst we did not specify a subset, a superset of $\mathcal{T}$ was taken to be sphere of radius equal to the longest $\underline{r}$ vector Euclidean norm length for $\mathcal{T}$, in other words the superset is the set of all operators with $\underline{r}$ vectors satisfying $\|\underline{r}\| \leq \delta(\mathcal{T})$.

The subsets and supersets characterised by the various norms are summarised as:

Norm	**Superset**	**Subset**	**Dual to superset**
$\alpha(\mathcal{S})$	$CUBE(\alpha)$	$OCT(\alpha)$	$OCT(\frac{1}{\alpha})$
$\beta(\mathcal{S})$	$RHOM(\beta)$	$CUBOCT(\frac{\beta}{2})$	$CUBOCT(\frac{1}{\beta})$
$\gamma(\mathcal{S})$	$OCT(\gamma)$	$CUBE(\frac{\gamma}{3})$	$CUBE(\frac{1}{\gamma})$
$\delta(\mathcal{S})$	$SPHE(\delta)$	-	$SPHE(\frac{1}{\delta})$

(7.8)

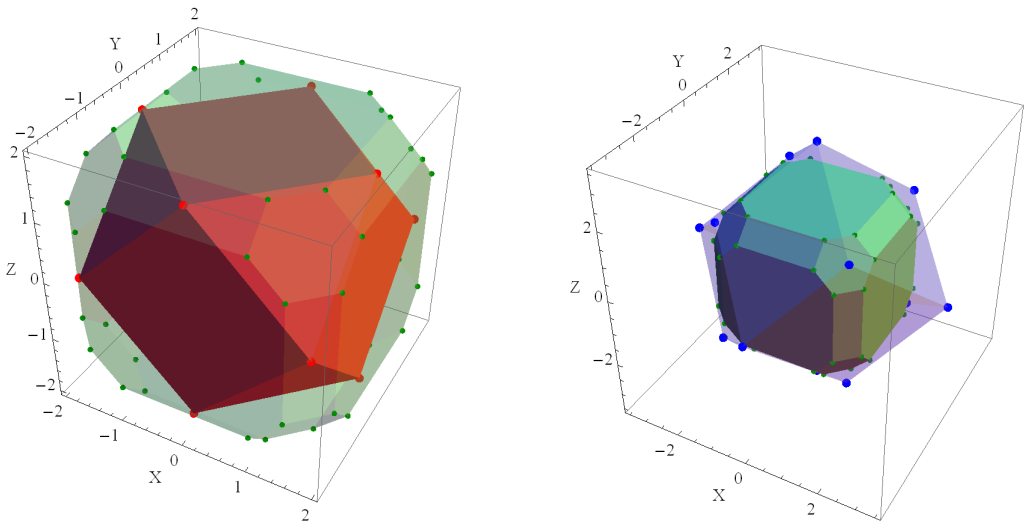


Figure 7.4: Representation of a subset and superset induced by $\beta(\mathcal{T})$. $\mathcal{T}$ is generated for the Bloch vector $\underline{r} = (1, 1.5, 2)$, (Left) shows $\mathcal{T}$ containing the subset $CUBOCT(\frac{\beta}{2})$ (Right) $\mathcal{T}$ is contained in the superset $RHOM(\beta)$.

7.3.5 Necessary constraints from superset-subset positivity

The supersets and subsets can be used with eq. (7.2) to arrive at lower bounds on $\lambda_{CZ_\varepsilon,\mathcal{S}}$. The noise acting on CZ_ε must be sufficient to ensure that any product input operator taken from product subsets remain separable with respect to the supersets. Using eq. (7.2) taking σ to be an operator from a subset, and $\mathcal{S}$ to be a superset we can arrive at necessary conditions as:

$$\text{tr}\left[(SUPSET^*)^{\otimes 2} CZ_\varepsilon (SUBSET^{\otimes 2})\right] \geq 0 \tag{7.9}$$

The quantum noise acting on the CZ gate, parametrised by ε must be sufficiently large to ensure the trace for all product subset inputs in eq. (7.2) are positive.

Focusing only on joint depolarising noise acting on the CZ gate, we will now look at the positivity bounds induced from each of the norms, that is we will scan over all inequalities in eq. (7.2) for a particular subset and superset induced by a norm, and select those requiring the most noise.

Necessary bounds from $\alpha(\mathcal{T})$

Cycling through all possible inequalities of the form:

$$\text{tr}\left[(CUBE(\alpha)^*)^{\otimes 2} CZ_\varepsilon (OCT(\alpha)^{\otimes 2})\right] \geq 0 \tag{7.10}$$

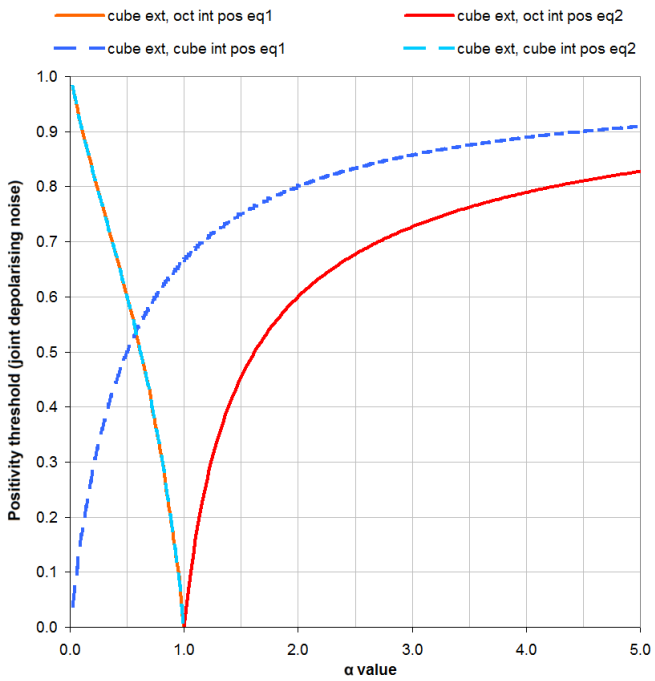


Figure 7.5: Necessary bounds induced by α. The solid lines represent the minimum noise needed to ensure positivity as in eq. (7.10).

---

we find the following two inequalities requiring the most noise:

$$\alpha < 1: \qquad \frac{\alpha^2 - 1}{\alpha^2 - \alpha - 1} \leq \lambda \tag{7.11}$$

$$\alpha \geq 1: \qquad \frac{\alpha^2 - 1}{\alpha^2 + \alpha - 1} \leq \lambda \tag{7.12}$$

Figure (7.5) shows a plot of these two bounds. At $\alpha = 1$ the superset cube contains the Bloch sphere. We can see that at this size no noise is needed to ensure positivity with eq. (7.10).

Necessary bounds from $\gamma(\mathcal{T})$

Cycling through all possible inequalities of the form:

$$\mathrm{tr}\left[(OCT(\gamma)^*)^{\otimes 2} CZ_\varepsilon (CUBE(\frac{\gamma}{3})^{\otimes 2})\right] \geq 0 \tag{7.13}$$

we find the following five inequalities requiring the most noise:

$$\begin{aligned}
\gamma < \frac{3}{2} \qquad & \frac{12 - 8\gamma}{12 + \gamma} \leq \lambda \\
\gamma < \tfrac{1}{2}(\sqrt{13} - 1) \qquad & \frac{-12 + 4\gamma + 4\gamma^2}{-12 - 5\gamma + 4\gamma^2} \leq \lambda \\
\tfrac{1}{2}(\sqrt{13} - 1) \leq \gamma < 2 \qquad & 0 \leq \lambda
\end{aligned}$$

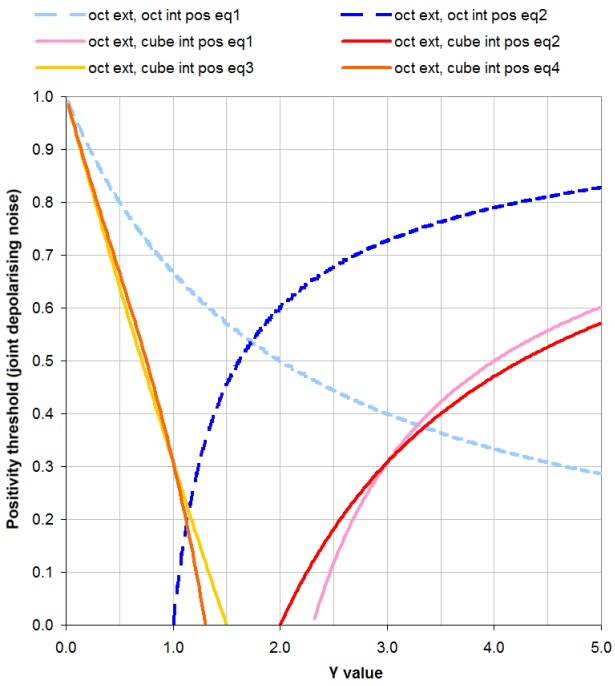


Figure 7.6: Necessary bounds induced by γ. The solid lines represent the minimum noise needed to ensure positivity as in eq. (7.13).

$$\begin{aligned} \gamma \geq 2 \qquad & \frac{4\gamma - 8}{4\gamma + 1} \leq \lambda \\ \gamma \geq \tfrac{1}{2}(1 + \sqrt{13}) \qquad & \frac{-12 - 4\gamma + 4\gamma^2}{-12 + 5\gamma + 4\gamma^2} \leq \lambda \end{aligned} \tag{7.14}$$

Figure (7.6) shows a plot of these bounds. For eq. (7.13), we see that subset inputs of size $\frac{1}{2}(\sqrt{13}-1) \leq \gamma < 2$ require no noise to ensure positivity with respect to $OCT(\gamma)^*$, this includes, at $\gamma = \sqrt{3}$ the Bloch sphere.

Necessary bounds from $\beta(\mathcal{T})$

Cycling through all possible inequalities of the form:

$$\mathrm{tr}\left[(RHOM(\beta)^*)^{\otimes 2} CZ_\varepsilon (CUBOCT(\frac{\beta}{2})^{\otimes 2})\right] \geq 0 \tag{7.15}$$

we find the following five inequalities requiring the most noise:

$$\begin{aligned} \beta < \frac{3}{2} \qquad & \frac{2\beta - 3}{2\beta + 1} \leq \lambda \\ \beta < \tfrac{1}{3}(\sqrt{33} - 1) \qquad & \frac{2\beta^2 + \beta - 4}{2\beta^2 - 3\beta - 4} / \leq \lambda \\ \tfrac{1}{3}(\sqrt{33} - 1) \leq \beta < \tfrac{4}{3} \qquad & 0 \leq \lambda \\ \beta \geq \frac{4}{3} \qquad & \frac{4 - 3\beta}{4 + \beta} \leq \lambda \end{aligned}$$

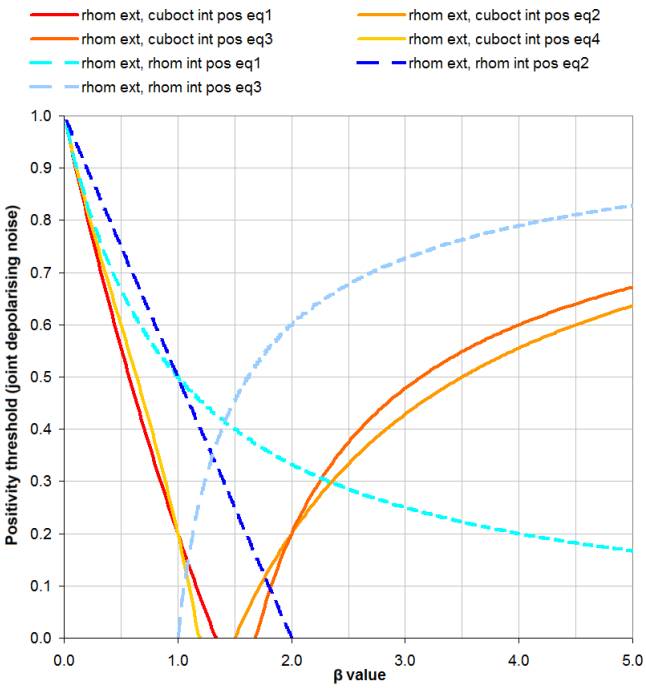


Figure 7.7: Necessary bounds induced by β. The solid lines represent the minimum noise needed to ensure positivity as in eq. (7.15).

$$\beta \geq \tfrac{1}{2}(\sqrt{33}+1) \qquad \frac{2\beta^2-\beta-4}{2\beta^2+3\beta-4} \leq \lambda \tag{7.16}$$

Figure (7.7) shows a plot of these bounds. For eq. (7.15) we see that for subset inputs of size $\frac{1}{3}(\sqrt{33}-1) \leq \beta < \frac{4}{3}$ we require no noise to ensure positivity with respect to $RHOM(\gamma)^*$, this includes, at $\beta = \sqrt{2}$ the Bloch sphere.

7.3.6 Comments on positivity bounds from norms

In order to contrast the positivity bounds plotted in figures (7.5), (7.14) and (7.7), we will relate the sets specified by each norm to a sphere of radius $\delta(\mathcal{T})$, i.e. we will relate the superset set specified to each norm to the largest sphere of size size $\delta(\mathcal{T})$ that the superset contains. This is equivalent to relating the subset specified by each norm to the smallest sphere of size $\delta(\mathcal{T})$ containing the subset. Figure (7.8) shows a plot of the necessary bounds in eq.'s (7.10), (7.13), and (7.15) scaled in terms of $\delta(\mathcal{T})$.

From figure (7.8) for a given $\mathcal{T}$ we would expect the superset/subset positivity for $\alpha(\mathcal{T})$ requiring a larger amount of noise to ensure positivity, than positivity required from considering $\gamma(\mathcal{T})$ or $\beta(\mathcal{T})$.

The most apparent feature is the minima for all norms around the $\delta = 1$ value. At this value the other norms take the values $\alpha = 1$, $\beta = \sqrt{2}$, and $\gamma = \sqrt{3}$. At these values

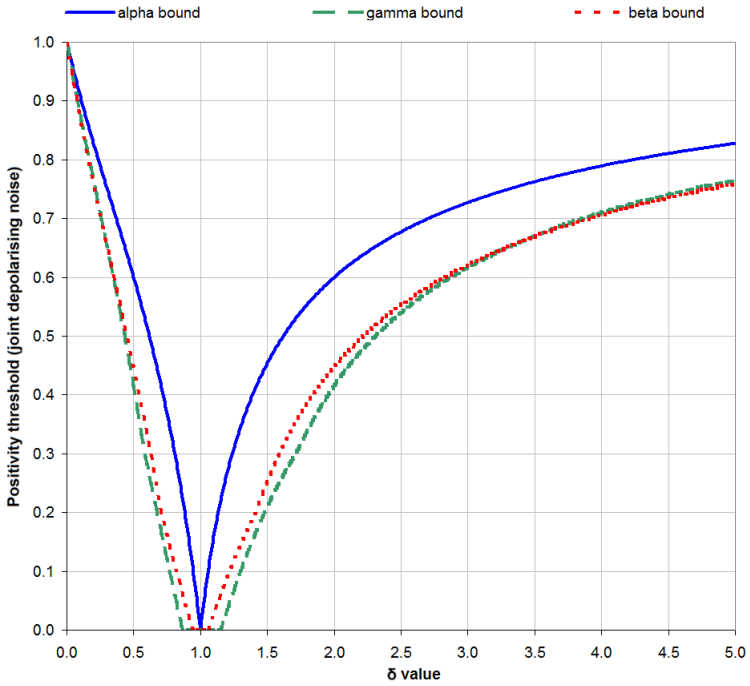


Figure 7.8: Necessary bounds plotted against $\delta(\mathcal{T})$. The supersets induced by each norm each contain a sphere of size $\delta(\mathcal{T})$. The solid line represent the amount of noise required for positivity from considering $\alpha(\mathcal{T})$, the dashed line from $\gamma(\mathcal{T})$, and the dotted line $\beta(\mathcal{T})$.

---

the subsets and duals to the supersets are equal and so we will describe these values as duality points for their respective norms.

We can see why this is a minima as follows: at these duality points the vertices of the superset dual/subsets of $\mathcal{T}$ correspond to operators on the surface of the Bloch sphere. The CZ operation, as a quantum operation automatically satisfies positivity constraints of the form in eq. (7.9) where the superset dual/subset is the Bloch sphere. In other words if the operators in the duals of the shapes and subsets, are at the surface of the Bloch sphere we have a local minima and require no extra noise for positivity.

7.4 Necessary from product structure constraints

As shown in section (4.6) a general two particle operator ρ may be represented using the Pauli table notation as the following array of numbers:

$$\left\{\begin{array}{cccc} 1 & \rho_{01} & \rho_{02} & \rho_{03} \\ \rho_{10} & \rho_{11} & \rho_{12} & \rho_{13} \\ \rho_{20} & \rho_{21} & \rho_{22} & \rho_{23} \\ \rho_{30} & \rho_{31} & \rho_{32} & \rho_{33} \end{array}\right\} \tag{7.17}$$

Suppose our algorithmic state space $\mathcal{T}^{\otimes 2}$ is formed from the convex hull of tensor products operators with $\underline{r}$ vectors of the form $\underline{r}_a := (x_a, y_a, z_a)$, and $\underline{r}_b := (x_b, y_b, z_b)$.

The elements of $\mathcal{T}^{\otimes 2}$ may be represented using a Pauli table as:

$$\left\{\begin{array}{cccc} 1 & x_b & y_b & z_b \\ x_a & x_a x_b & x_a y_b & x_a z_b \\ y_a & y_a x_b & y_a y_b & y_a z_b \\ z_a & z_a x_b & z_a y_b & z_a z_b \end{array}\right\} \tag{7.18}$$

We are interested in determining whether the arbitrary operator ρ is in $\mathcal{T}^{\otimes 2}$. The method outline here will look at forming vectors of certain elements in the Pauli tables in eq.'s (7.17) and (7.18), and then compare the norms of the two vectors.

For example, suppose we picked two elements of the top row of the table in eq. (7.17) forming a two element vector out of them as (ρ_{01}, ρ_{02}) . We will compare this vector to an analogous vector formed from the table in eq. (7.18) requiring that for our choice of norm $\|\cdot\|$ the following expression is satisfied:

$$\|(\rho_{01}, \rho_{02})\| \leq \max\{\|(x_b, y_b)\| \mid (x_b, y_b, z_b) \in \mathcal{T}\} \tag{7.19}$$

We can see that eq. (7.19) is solely dependent on $\underline{r}_b$. It is possible to construct inequalities dependent on both $\underline{r}_a$ and $\underline{r}_b$; for example let us consider a vector formed from the elements $(\rho_{11}, \rho_{23}, \rho_{32})$, if we compare it to a vector $(x_a x_b, y_a z_b, z_a y_b)$, for our choice of norm we arrive at the condition:

$$\|(\rho_{11}, \rho_{23}, \rho_{32})\| \leq \max\{\|(x_a x_b, y_a z_b, z_a y_b)\| \mid \underline{r}_a, \underline{r}_b \in \mathcal{T}\} \tag{7.20}$$

If we choose the l_1 norm, then we know from eq. (7.20) that it is necessary that the components satisfy:

$$|\rho_{11}| + |\rho_{23}| + |\rho_{32}| \leq x^2 + 2|yz| = (\delta(\mathcal{T}))^2 - y^2 - z^2 + 2|yz|$$

$$\rightarrow |\rho_{11}| + |\rho_{23}| + |\rho_{32}| \leq (\delta(\mathcal{T}))^2 \tag{7.21}$$

This inequality will be used in next sections. For $\delta = 1$, it requires the partial transpose to have positive expectation values on Bell pairs, in this, it is related to the PPT test applied to quantum states.

7.5 Combining necessary constraints

Here we will show how we can use the delta inequality in eq. (7.21) and positivity requirements in section (7.2) to arrive at stronger necessary bounds on the joint depolarising noise rate to make the CZ gate $\mathcal{T}$-separable.

Let ϱ denote two particle state outputted from a noisy CZ gate affected by joint depolarising noise acting on an operator from $\mathcal{T}^{\otimes 2}$. If $\mathcal{T}$ is contained within the cube

superset specified by $\alpha(\mathcal{S})$ and ϱ is $\mathcal{S}$-separable, then we see from eq. (7.2) and table in eq. (7.8) that ϱ must satisfy the following inequalities:

$$\text{tr}\left[OCT(\frac{1}{\alpha})^{\otimes 2}\varrho\right] \geq 0 \tag{7.22}$$

In particular, we will focus on one of these inequalities - that generated by $\sigma \otimes \eta$, where σ, $\eta \in OCT(\frac{1}{\alpha})$, with σ represented by the Bloch vector $(-\frac{1}{\alpha}, 0, 0)$, and η by $(0, -\frac{1}{\alpha}, 0)$:

$$\text{tr}\left[(\sigma \otimes \eta)\varrho\right] \geq 0 \tag{7.23}$$

The Pauli table for $\sigma \otimes \varsigma$ is:

$$\frac{1}{4}\begin{Bmatrix} 1 & 0 & -\frac{1}{\alpha} & 0 \\ -\frac{1}{\alpha} & 0 & \frac{1}{\alpha^2} & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{Bmatrix} \tag{7.24}$$

It is clear from table (7.24) using eq. (4.32) that the trace becomes

$$\frac{(1-\varepsilon)}{4}\left(\frac{1}{(1-\varepsilon)} - \frac{1}{\alpha}xC - \frac{1}{\alpha}zB + \frac{1}{\alpha^2}(-yA)\right) \geq 0 \tag{7.25}$$

We are free to to choose input states from $\mathcal{S}$ into the CZ operator such that $A = y$, $B = z$, $C = x$. Simplifying (7.25) we can rearrange terms where x, y, and z are picked to achieve $\delta(\mathcal{T})$ - inserting the norm $\delta^2 = x^2 + y^2 + z^2$:

$$\frac{1}{(1-\varepsilon)} - \frac{1}{\alpha}(x^2 + z^2 + \frac{y^2}{\alpha}) \geq 0$$

$$\varepsilon \geq 1 - \frac{\alpha}{(\delta^2 + (\frac{1-\alpha}{\alpha})y^2)} \tag{7.26}$$

For $\alpha \geq 1$ we see that $\left(\frac{1-\alpha}{\alpha}\right)$ becomes negative, and so the largest the right side of inequality (7.26) can only be achieved when y is the smallest value possible, i.e. when $y = 0$. Inequality (7.26) becomes:

$$\varepsilon \geq 1 - \frac{\alpha}{\delta^2} \tag{7.27}$$

For $\alpha \leq 1$, $\left(\frac{1-\alpha}{\alpha}\right)$ remains positive and so we can make y the largest possible value, that is $y = \alpha$:

$$\varepsilon \geq 1 - \frac{\alpha}{(\delta^2 + (1-\alpha)\alpha)} \tag{7.28}$$

Using inequality (7.21) for input state choices represented by Bloch vectors $(0, \alpha, 0)$, and $(0, \alpha, 0)$ gives:

$$(1-\varepsilon)(\alpha^2+2\alpha) \leq \delta^2 \Longrightarrow 1 - \frac{\delta^2}{\alpha(\alpha+2)} \leq \varepsilon \tag{7.29}$$

As inequalities (7.27) and (7.28) are monotonically increasing in δ, and inequality (7.29) is monotonically decreasing for a fixed value of α it is possible to tie these inequalities together to eliminate δ by, for a given value of α, finding the value of δ that maximises the inequalities on ε: We arrive at the following inequalities on ε:

$$\alpha \leq 1: \quad \varepsilon \geq 1 - \frac{(\alpha-1)+\sqrt{\alpha^2+2\alpha+9}}{2(\alpha+2)} \tag{7.30}$$

$$\alpha \geq 1: \quad \varepsilon \geq 1 - \frac{1}{\sqrt{\alpha+2}} \tag{7.31}$$

Note that for $\alpha = 1$ these two constraints attain the value $1 - \frac{1}{\sqrt{3}} \approx 0.423$, not the minima observed from the positivity constraints.

7.6 Sufficiency for sets with $\alpha(\mathcal{T}) = \delta(\mathcal{T})$

The simplest single particle state space polytope we can choose for $\alpha = 1$ is an octahedron formed of Pauli eigenstates. We know from the Gottesman-Knill result that this state space (where $\alpha = 1$) in conjunction with Pauli measurements/preparation and Clifford dynamics is efficiently classically simulatable.

Within this section we will make a specific choice of $\mathcal{T}$ as $OCT(\alpha)$. We will analytically calculate the amount of noise needed to ensure that the noisy CZ gate (under joint-depolarising noise) becomes $OCT(\alpha)$-separable, showing that such noise levels are also sufficient and the necessary bounds determined previously are satisfied.

From the symmetry arguments in section (4.5) we only need to six of the product input states from $OCT(\alpha)^{\otimes 2}$. These input states transform as follows under a CZ gate acted upon by joint depolarising noise:

$$\frac{1}{4}\begin{Bmatrix} 1 & \alpha & 0 & 0 \\ \alpha & \alpha^2 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{Bmatrix} \rightarrow \frac{(1-\varepsilon)}{4}\begin{Bmatrix} \frac{1}{(1-\varepsilon)} & 0 & 0 & 0 \\ 0 & 0 & 0 & \alpha \\ 0 & 0 & \alpha^2 & 0 \\ 0 & \alpha & 0 & 0 \end{Bmatrix} \tag{7.32}$$

$$\frac{1}{4}\begin{Bmatrix} 1 & 0 & \alpha & 0 \\ \alpha & 0 & \alpha^2 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{Bmatrix} \rightarrow \frac{(1-\varepsilon)}{4}\begin{Bmatrix} \frac{1}{(1-\varepsilon)} & 0 & 0 & 0 \\ 0 & 0 & 0 & \alpha \\ 0 & -\alpha^2 & 0 & 0 \\ 0 & 0 & \alpha & 0 \end{Bmatrix} \tag{7.33}$$

$$\frac{1}{4}\begin{Bmatrix} 1 & 0 & \alpha & 0 \\ 0 & 0 & 0 & 0 \\ \alpha & 0 & \alpha^2 & 0 \\ 0 & 0 & 0 & 0 \end{Bmatrix} \rightarrow \frac{(1-\varepsilon)}{4}\begin{Bmatrix} \frac{1}{(1-\varepsilon)} & 0 & 0 & 0 \\ 0 & \alpha^2 & 0 & 0 \\ 0 & 0 & 0 & \alpha \\ 0 & 0 & \alpha & 0 \end{Bmatrix} \tag{7.34}$$

$$\frac{1}{4}\begin{Bmatrix} 1 & 0 & 0 & \alpha \\ \alpha & 0 & 0 & \alpha^2 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{Bmatrix} \rightarrow \frac{(1-\varepsilon)}{4}\begin{Bmatrix} \frac{1}{(1-\varepsilon)} & 0 & 0 & \alpha \\ \alpha^2 & 0 & 0 & \alpha \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{Bmatrix} \tag{7.35}$$

$$\frac{1}{4}\begin{Bmatrix} 1 & 0 & 0 & \alpha \\ 0 & 0 & 0 & 0 \\ \alpha & 0 & 0 & \alpha^2 \\ 0 & 0 & 0 & 0 \end{Bmatrix} \rightarrow \frac{(1-\varepsilon)}{4}\begin{Bmatrix} \frac{1}{(1-\varepsilon)} & 0 & 0 & \alpha \\ 0 & 0 & 0 & 0 \\ \alpha^2 & 0 & 0 & \alpha \\ 0 & 0 & 0 & 0 \end{Bmatrix} \tag{7.36}$$

$$\frac{1}{4}\begin{Bmatrix} 1 & 0 & 0 & \alpha \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ \alpha & 0 & 0 & \alpha^2 \end{Bmatrix} \rightarrow \frac{(1-\varepsilon)}{4}\begin{Bmatrix} \frac{1}{(1-\varepsilon)} & 0 & 0 & \alpha \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ \alpha & 0 & 0 & \alpha^2 \end{Bmatrix} \tag{7.37}$$

We will now work through each of the noisy CZ output states showing that they can be expressed as convex combinations of $OCT(\alpha)$-separable state if for some particular range of values α values the rate of noise ε is above a certain value.

We can however, place the six output states into three groups containing states that require the same amount of noise to ensure CZ $OCT(\alpha)$-separability. The first group contains output states; in eq.'s (7.32), (7.33), and (7.34), the second; (7.35), and (7.36), and the third group only (7.37).

Working through the first set; the output state in eq. (7.32) can be expressed as the following summation of $OCT(\alpha)$-separable states

$$\frac{(1-\varepsilon)}{4}\begin{Bmatrix} \frac{1}{(1-\varepsilon)} & 0 & 0 & 0 \\ 0 & 0 & 0 & \alpha \\ 0 & 0 & \alpha^2 & 0 \\ 0 & \alpha & 0 & 0 \end{Bmatrix} = \frac{1}{4}\begin{Bmatrix} (t+u+v) & 0 & 0 & 0 \\ 0 & 0 & 0 & v\alpha^2 \\ 0 & 0 & u\alpha^2 & 0 \\ 0 & t\alpha^2 & 0 & 0 \end{Bmatrix}$$

$$= \frac{t}{4}\begin{Bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & \alpha^2 & 0 & 0 \end{Bmatrix} + \frac{u}{4}\begin{Bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & \alpha^2 & 0 \\ 0 & 0 & 0 & 0 \end{Bmatrix} + \frac{v}{4}\begin{Bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & \alpha^2 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{Bmatrix} \tag{7.38}$$

It is important here to note that the separable states within the convex sum be (7.38) can be expressed in terms of a summation of product input states.

$$\frac{t}{4}\left\{\begin{array}{cccc} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & \alpha^2 & 0 & 0 \end{array}\right\} = \frac{t}{8}\left\{\begin{array}{cccc} 1 & \alpha & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ \alpha & \alpha^2 & 0 & 0 \end{array}\right\} + \frac{t}{8}\left\{\begin{array}{cccc} 1 & -\alpha & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ -\alpha & \alpha^2 & 0 & 0 \end{array}\right\} \tag{7.39}$$

Comparing table entries in (7.38) we can arrive at the following four expressions:

$$\begin{array}{rcl} t+u+v & = & 1 \\ t & = & \frac{1}{\alpha}(1-\varepsilon) \\ u & = & (1-\varepsilon) \\ v & = & \frac{1}{\alpha}(1-\varepsilon) \end{array} \tag{7.40}$$

Such a convex summation is only possible when we have at least a noise rate of $\varepsilon = 1 - \frac{a}{2+\alpha}$. It is straightforward to show that the other output states in the first group also require a noise rate of $\varepsilon = 1 - \frac{a}{2+\alpha}$ to ensure $OCT(\alpha)$-separability.

In the second group we have for eq. (7.35), the following convex summation of product input states

$$\frac{(1-\varepsilon)}{4}\left\{\begin{array}{cccc} \frac{1}{(1-\varepsilon)} & 0 & 0 & \alpha \\ \alpha^2 & 0 & 0 & \alpha \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{array}\right\} = \frac{p}{4}\mathcal{A} + \frac{q}{4}\mathcal{B} + \frac{r}{4}\mathcal{C} + \frac{s}{4}\mathcal{D} \tag{7.41}$$

with the following product tables:

$$\mathcal{A} = \left\{\begin{array}{cccc} 1 & 0 & 0 & \alpha \\ \alpha & 0 & 0 & \alpha^2 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{array}\right\} \qquad \mathcal{B} = \left\{\begin{array}{cccc} 1 & 0 & 0 & \alpha \\ -\alpha & 0 & 0 & -\alpha^2 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{array}\right\}$$

$$\mathcal{C} = \left\{\begin{array}{cccc} 1 & 0 & 0 & -\alpha \\ \alpha & 0 & 0 & -\alpha^2 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{array}\right\} \qquad \mathcal{D} = \left\{\begin{array}{cccc} 1 & 0 & 0 & -\alpha \\ -\alpha & 0 & 0 & \alpha^2 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{array}\right\}$$

Comparing Pauli table components we arrive at a set of linear eq.'s that can be expressed as the following matrix:

$$\underbrace{\left[\begin{array}{cccc} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & 1 & -1 \\ 1 & -1 & -1 & 1 \end{array}\right]}_{K} \cdot \left(\begin{array}{c} p \\ q \\ r \\ s \end{array}\right) = \left(\begin{array}{c} 1 \\ (1-\varepsilon) \\ \alpha(1-\varepsilon) \\ \frac{(1-\varepsilon)}{\alpha} \end{array}\right) \tag{7.42}$$

As matrix K is non-singular we can invert it to arrive at expressions for the convex coefficients p, q, r, and s:

$$p = \frac{1}{4\alpha}\Big(\alpha + (1-\varepsilon)(\alpha^2+\alpha+1)\Big)$$

$$q = \frac{1}{4\alpha}\Big(\alpha - (1-\varepsilon)(\alpha^2-\alpha+1)\Big)$$

$$r = \frac{1}{4\alpha}\Big(\alpha - (1-\varepsilon)(1+\alpha-\alpha^2)\Big)$$

$$s = \frac{1}{4\alpha}\Big(\alpha - (1-\varepsilon)(\alpha^2+\alpha-1)\Big)$$

As with the first group we require that each of these convex coefficient must be positive. This requirement allows us to calculate the minimum value of ε to ensure all coefficients remain positive.

$$p \geq 0 \rightarrow 1 \geq \varepsilon$$

$$q \geq 0 \rightarrow \varepsilon \geq 1 - \frac{\alpha}{(\alpha^2-\alpha+1)}$$

$$r \geq 0 \rightarrow \begin{cases} 0 < \alpha < 1 & : \quad \varepsilon \geq 1 - \frac{\alpha}{(1+\alpha-\alpha^2)} \\ \alpha \geq 1 & : \quad 1 \geq \varepsilon \end{cases}$$

$$s \geq 0 \rightarrow \begin{cases} 0 < \alpha < 1 & : \quad 1 \geq \varepsilon \\ \alpha \geq 1 & : \quad \varepsilon \geq 1 - \frac{\alpha}{(\alpha^2+\alpha-1)} \end{cases} \tag{7.43}$$

The last two inequalities of (7.43) are key requiring the most amount of noise. It is worth noting that these inequalities are equal to the necessary constraints in eq.'s (7.11), and (7.12).

Finally for the third group we can immediately see that the output state in eq. (7.37) is separable without the need for any noise ε.

Putting together the inequalities from all three sets we have

$$0 < \alpha < \sqrt{3}: \quad \varepsilon \geq 1 - \frac{a}{2+\alpha}$$

$$\alpha \geq \sqrt{3}: \quad \varepsilon \geq 1 - \frac{\alpha}{(\alpha^2+\alpha-1)} \tag{7.44}$$

We have shown for all possible values of α that above noise bounds (7.44) the CZ gate is $OCT(\alpha)$-separable. Figure (7.9) shows a plot of the sufficiency bounds in (7.44) and the necessary bounds determined in previous sections. We see that the bounds (7.44) are both necessary and sufficient - sufficient in the sense that we have shown explicitly that the convex decomposition at these noise rates, and necessary, as for $0 < \alpha < \sqrt{3}$ we achieve a bound from product structure constraints and for $\alpha \geq \sqrt{3}$ we have a bound from Pauli measurement positivity.

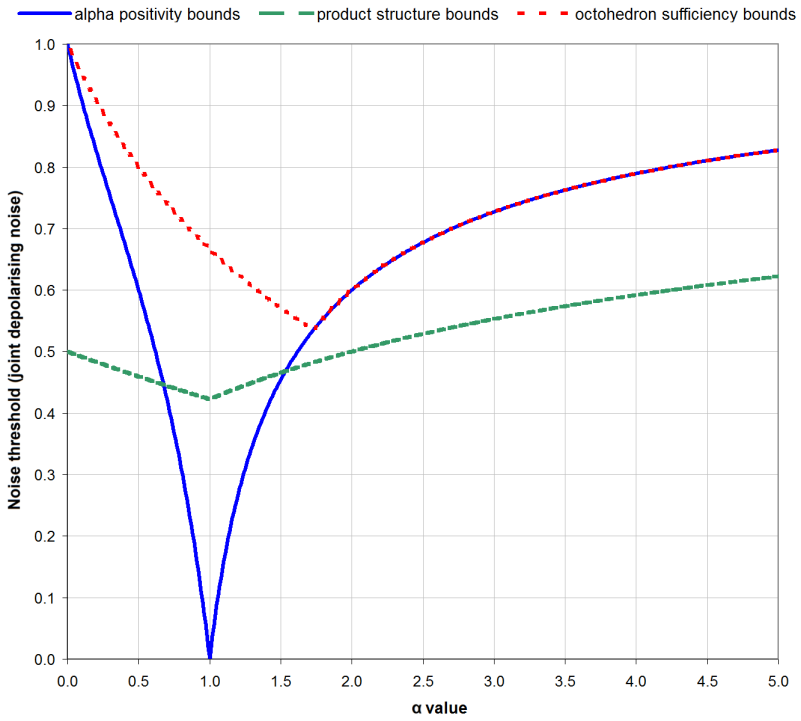


Figure 7.9: Plot of α necessary bounds (7.11) & (7.12) (solid line), product structure necessary bounds (7.30) & (7.31) (dashed line), and octahedron sufficiency inequalities (7.44) (dotted line). The region above the two necessary curves and below the sufficiency curve represents the range of joint depolarising noise rates that for a given α value ensures that the CZ-gate is $OCT(\alpha)$-separable.

While for $0 < \alpha < \sqrt{3}$, the $OCT(\alpha)$ is not the optimal shape it does provide an upper limit to the noise required for output separability - an optimal shape contained within a cube of the size $\alpha = 1$ would require a joint depolarising noise rate of between approximately 0.423 and 0.667.

7.7 Summary of Chapter (7)

In this chapter we attempted to narrow the choices on the optimal $\mathcal{S}$ by constructing necessary bounds from considering norms on the Bloch vectors of operators in $\mathcal{S}$. These norms were used to specify octahedrally symmetric supersets and related subsets, parametrising each shape by the norm length of its vertices. We determined, for each norm, the amount of noise required to ensure trace positivity of operators from the subsets with the duals to the supersets as a function of the norm parameters. We considered four norms, each specifying a different superset shape, and while not using a rescaling parameter, identified shrinking/expanding of the state space with noise on the preparation/measurement. The key inequalities for each norm were identified and

contrasted with each other by relating each superset to a sphere contained within the superset. By matching all the spheres we identified the superset-norm necessary bounds requiring the most amount of noise for positivity, and found it to be those induced from considering the cube superset with an associated octahedron subset.

In the next section we contrasted Pauli table elements of an $\mathcal{S}$-separable state, with elements from an arbitrary two particle state, arguing that the elements form the arbitrary state should satisfy be less than some norm length of the $\mathcal{S}$-separable elements. This allowed us to construct another necessary bound. Drawing on what was learnt with the superset-subset necessary bounds, we constructed a new necessary bound arguing that an arbitrary CZ gate output state must satisfy positivity with respect to a cube superset. We finally combined this, and the previous Pauli table structure constraint to arrive at a tight necessary condition as a function of the cube norm length.

In the final section we constructed $\mathcal{S}$-separability sufficiency bounds for $\mathcal{S}$ taken to an octahedron of Pauli eigenstates rescaled to fit exactly inside a cube superset. When comparing the final necessary bounds and the octahedron sufficiency bound we identified the octahedron to be the optimal shape above a certain amount of rescaling. Under that level of rescaling, the optimal $\mathcal{S}$ remains a unknown.

8 Summary and conclusion

In this thesis we have explored the role of generalised non-quantum notions of entanglement in the efficient classical simulation of noisy quantum devices. We arrived at these new notions by considering a redefined single particle state space based on restrictions to the set of available measurements to the system. These state spaces were sets of quantum, and non-quantum operators that returned positive measurement probabilities with respect to the restricted measurements. The new state spaces were then employed to modify the Harrow and Nielsen algorithm [6] to arrive at new algorithms for classical simulation.

The aim of the research was to come up with new classical algorithms could efficiently classically simulate regimes outside those covered by existing schemes.

Restricted measurements come about in two ways; firstly where the restrictions are a design feature as with some fault tolerant schemes, and secondly where the available measurements are faulty or non-ideal. Within this thesis we chose to focus on magic state architecture based devices because of the developed fault tolerant schemes for them. In particular we considered the case where the single qubit Clifford unitaries were ideal and only the two-qubit CX gate (or equivalently for us the CZ gate) was susceptible to noise.

The restriction to only Pauli measurements, specified a Bloch-cube within which we could choose our new state space $\mathcal{S}$ (subject to restrictions from the single qubit dynamics). We also restricted $\mathcal{S}$ to be invariant under single qubit unitaries and so as the CZ gate was the only "entangling" gate for the possible state spaces, we required it be subject to noise to ensure $\mathcal{S}$-separability so that the Harrow and Nielsen algorithm could be applied. The main technical problem of the thesis was to find the optimal state space $\mathcal{S}$ requiring the least amount of noise to ensure CZ output $\mathcal{S}$-separability, for a range of natural noise models.

We looked at a number of possible of state spaces, starting with the Bloch sphere, and the whole measurement dual state space - the Bloch cube allowing for their modification to account for faulty preparation and measurement. We then looked at the truncated cube, and finally ended by looking at cube, rhombic dodecahedron, octahedron, and sphere supersets to arrive at necessary bounds.

By looking at non-quantum notions of entanglement, we were considering a type of generalised probability theory, where we had a non-quantum state space, but noisy

quantum state dynamics. At particular sufficient noise rates these dynamics were rendered essentially classical. Observations made about the relationship between cube-entanglement, quantum entanglement and non-locality, motivated the research.

In Chapter (5) we presented noise threshold calculation results for the Bloch sphere subject to faulty preparation/measurement, and the Bloch cube with ideal and faulty preparation/measurement, for each of the noise models; local depolarising, joint depolarising, and local dephasing noise.

In general we observed little to no appreciable improvement with the cube state spaces in the ideal case over those obtained using Harrow and Nielsen or application of the Gottesman-Knill theorem for the noise models we considered. In the case where we allowed for fault in the preparation/measurement for Bloch spheres and cubes we did observe some slight improvements. Specifically by adding slight amounts of local depolarising noise immediately prior to measurement, we were able to consider larger versions of the Bloch sphere and cube state space, that required less noise to ensure separable CZ-output. Adding noise to qubit preparation however had the inverse effect, requiring us to add more noise for separability. Interestingly, for Bloch sphere input, a CZ gate undergoing local dephasing noise, required maximum noise to ensure separable output if the Bloch sphere was rescaled in anyway for faulty preparation/measurement.

In Chapter (6) we presented our best candidate for choice for $\mathcal{S}$ - the truncated cube, a cube with its corners sliced-off, the amount of "slicing-off" parametrised by value c. We showed how the separability calculation could be transformed into a linear programming problem, whose numerical results we could uses to make educated guesses to arrive at a symbolic solution for the joint depolarising noise threshold. We calculated this value to be approximately 0.5562, when $c \approx 0.5$ and the truncated cube is contained within a cube of size $\alpha = 1$, and improvement over the $\frac{2}{3}$ achieved with the cubes in the previous chapters. Again we did not consider the other noise models as they would not have transformed into linear programming problem, but instead into optimisation problems with linear objective functions and quadratic constraints.

In the final Chapter (7) we characterised single particle state spaces using norms on the Bloch vectors of the operators in the space. These norms were used to specify subsets and supersets that gave lower bounds on the joint depolarising noise threshold values. We allowed for faulty measurement by allowing these supersets to extend outside the Bloch cube. The optimal state space contained within a cube of size $\alpha \geq \sqrt{3}$ was found to be an octahedron/sphere meeting the cube in the Pauli eigenstate directions. In the case where we considered a cube of size $0 < \alpha < \sqrt{3}$, we could not find an optimal, and instead could only specify a region within which the optimal must lie. The methods we used to arrive at positivity bounds can be extended to the other noise models in future work.

To summarise, in this thesis we have shown, for quantum computers with restricted measurements, we could use a non-quantum notion of entanglement with a Harrow and Nielsen type algorithm to simulate a broader range of noisy quantum computers with limited entanglement. This non-quantum notion of entanglement allowed us to redefine the single particle state space $\mathcal{S}$ used in the algorithm, and we investigated the relationship between $\mathcal{S}$ and the noise threshold required for the algorithm to become applicable. For a particular noise model, when our preparations/measurements were noisy, we were able to identify the optimal state spaces leading to the lowest threshold values. For the noiseless preparations/measurement case we found the truncated cube state space attained a lower threshold value than previous work, but identifying the optimal state space remains an open problem. In addition to the straightforward connection to the fault tolerance noise threshold, the approach taken in this thesis has strong links to studies of generalised probability theories, and would be of interest to those looking at the role of entanglement in the classical simulation of quantum computers with restricted measurements.

Table (8.1) summarises our results for the various choices of $\mathcal{S}$.

Table 8.1: Summary of results for the choices of $\mathcal{S}$

$\mathcal{S}$ choice	Noise type	Rescaling parameter	Noise rate	Source	Comment
Rescaled Sphere	Joint depolarising	≈ 1.73	≈ 0.536	Numerics	
Rescaled Sphere	Local depolarising	≈ 1.16	≈ 0.395	Numerics	
Rescaled Sphere	Joint dephasing	1	0.5	Analytics	See section (5.4.3)
Cube	Joint depolarising	1	≈ 0.667	Analytics	
Cube	Local depolarising	1	≈ 0.586	Analytics	
Cube	Joint dephasing	1	≈ 0.293	Analytics	
Rescaled Cube	Joint depolarising	0.71	≈ 0.586	Numerics	
Rescaled Cube	Local depolarising	0.54	≈ 0.406	Numerics	
Rescaled Cube	Joint dephasing	$\neq 1$	0.5	Analytics	Se section (5.6.3)
Truncated Cube	Joint depolarising	1	≈ 0.556	Symbolic	For truncation $c \approx 0.5$
Rescaled Sphere/Octahedron	Joint depolarising	$\gtrsim 1.73$	$\gtrsim 0.536$	Analytics	Optimal shape for $\alpha \geq \sqrt{3} \approx 1.73$

9 Appendix

9.1 Example: The single bit-flip code

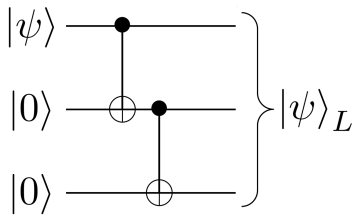


Figure 9.1: Diagram showing quantum circuit used to encode single qubit states $|\psi\rangle$ into the three qubit bit-flip code.

The best example to see the mechanics of a basic quantum code is the incomplete three qubit code for correcting only single qubit bit-flip errors[1]. The Pauli X operator acting on the single qubit basis states has the effect of "flipping" the single qubit logical basis states:

$$X|0\rangle \rightarrow |1\rangle\,, \qquad X|1\rangle \rightarrow |0\rangle \tag{9.1}$$

We may protect against single qubit bit-flip errors using a three qubit code with the following logical basis states:

$$|0\rangle_L = |000\rangle\,, \qquad |1\rangle_L = |111\rangle \tag{9.2}$$

An arbitrary single qubit state $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$ is thus mapped to:

$$\begin{aligned} |\psi\rangle = \alpha|0\rangle + \beta|1\rangle \rightarrow |\psi\rangle_L &= \alpha|0\rangle_L + \beta|1\rangle_L \\ &= \alpha|000\rangle + \beta|111\rangle \end{aligned} \tag{9.3}$$

Figure (9.1) shows a quantum circuit capable to encoding $|\psi\rangle$ to $|\psi\rangle_L$.

A single qubit bit flip error channel acting on the logical basis states in (9.2) has the following Kraus elements:

[1] The three qubit code is an incomplete code as it cannot correct for any arbitrary single qubit quantum error.

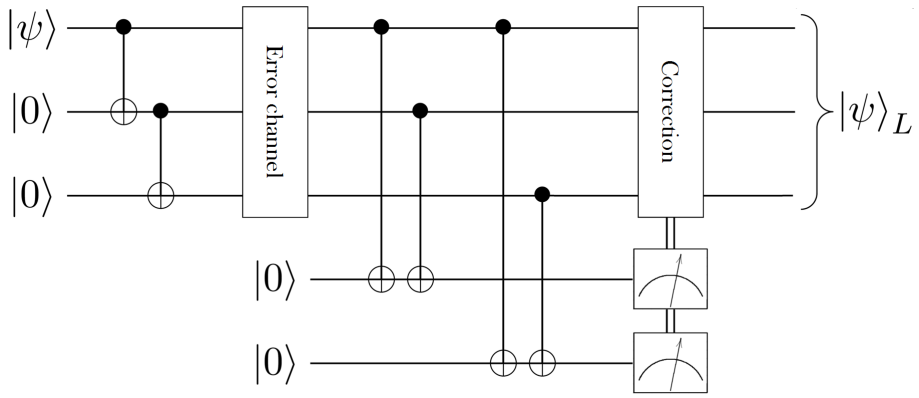


Figure 9.2: Diagram showing quantum circuit encoding into the three qubit bit-flip code, exposure a bit-flip error channel and recovery operation. Here the recovery operation is formed of two stages; The first introduces ancilla states for use with syndrome measurements to detect whether errors have occurred, they then feed classical information to select the appropriate correction operation to be enacted.

$$\{\mathbb{1}\otimes\mathbb{1}\otimes\mathbb{1},\, X\otimes\mathbb{1}\otimes\mathbb{1},\, \mathbb{1}\otimes X\otimes\mathbb{1},\, \mathbb{1}\otimes\mathbb{1}\otimes X\} \tag{9.4}$$

corresponding to no error, and a lone bit-flip on the first, second, and third physical qubits respectively.

The recovery operation here will be broken down into two stages; firstly a *syndrome measurements* to ascertain what error have occurred and finally a restorative operation undoing the effects of the error channel.

The syndrome measurement involves introducing ancilla qubits and measuring them to discern the nature of the error that may occurred. As with eq. (2.16), syndrome measures should in no way learn anything about $|\psi\rangle_L$. Figure (9.2) shows a quantum circuit diagram depicting the three qubit error correction scheme.

For the bit-flip code, the addition of two ancilla is sufficient to discern the location of a flip error on one of the qubits forming the encoded state. Various control-not operations are used to couple the ancilla to the corrupted encoded state. Immediately prior to measurement the state of the corrupted encoded state is:

Location of error	State prior to measurement
No error	$\alpha\,\|000\rangle\,\|00\rangle + \beta\,\|111\rangle\,\|00\rangle$
Qubit 1	$\alpha\,\|100\rangle\,\|11\rangle + \beta\,\|011\rangle\,\|11\rangle$
Qubit 2	$\alpha\,\|010\rangle\,\|10\rangle + \beta\,\|101\rangle\,\|10\rangle$
Qubit 3	$\alpha\,\|001\rangle\,\|01\rangle + \beta\,\|110\rangle\,\|01\rangle$

(9.5)

Projective Pauli X measurements on the ancillas yield the bit pairs; 00, 11, 10, 01 corresponding to no error, and bit-flip on the first, second, and third qubit of the encoded state. Crucially these projective measurements have no bearing on the superpositions of the corrupt encoded state. With the error identified, application of a Pauli X operator to the effected qubit with undo the effects of noise on the original state.

9.2 Example: The 3 qubit stabiliser group

The three qubit stabiliser set $\mathcal{S} = \{\mathbb{1}^{\otimes 3},\ Z\otimes Z\otimes \mathbb{1},\ \mathbb{1}\otimes Z\otimes Z,\ Z\otimes \mathbb{1}\otimes Z\}$ has generator elements $Z\otimes Z\otimes \mathbb{1}$, and $Z\otimes \mathbb{1}\otimes Z$. We may therefore write $\mathcal{S} = \langle Z\otimes Z\otimes \mathbb{1},\ Z\otimes \mathbb{1}\otimes Z\rangle$.

The table below lists the states stabilised by each non-identity element of $\mathcal{S}$:

Operator	State Stabilized
$Z\otimes Z\otimes \mathbb{1}$	$\|000\rangle, \|001\rangle, \|110\rangle, \|111\rangle$
$\mathbb{1}\otimes Z\otimes Z$	$\|000\rangle, \|100\rangle, \|011\rangle, \|111\rangle$
$Z\otimes \mathbb{1}\otimes Z$	$\|000\rangle, \|010\rangle, \|011\rangle, \|111\rangle$

(9.6)

From (9.6) we see that the only vectors stabilized by all of the elements of S are $|000\rangle, |111\rangle$, and thus the associated subspace V_S is spanned by $|000\rangle, |111\rangle$.

9.3 Measurement in the stabiliser formalism

It is straightforward to calculate the effects of Pauli group element measurements within the stabiliser formalism. Let us consider an element M of the Pauli group $\mathcal{G}_n$. Without loss of generality M can be assumed to not have a factor of $\pm i$. We can therefore take M to be a Hermitian with $M^2 = \mathbb{1}^{\otimes n}$ and the following projectors:

$$P_+ = \frac{1}{2}(\mathbb{1}^{\otimes n} + M) \qquad P_- = \frac{1}{2}(\mathbb{1}^{\otimes n} - M) \tag{9.7}$$

associated with the eigenvalues $+1$, and -1 respectively.

Suppose we have a state $|\psi\rangle$ stabilised by the set $\mathcal{S} = \langle S_1, S_2, ..., S_r\rangle$. The action of a projective measurement of M on $|\psi\rangle$ has the two possible effects depending on whether or not M commutes with the elements of $\mathcal{S}$:

1. M commutes with all elements $S \in \mathcal{S}$.
2. M anti-commutes with some or all generators of $\mathcal{S}$.

In the first case, as M commutes with the elements of $\mathcal{S}$ and has an eigenvalue $+1$, it follows that either M or $-M$ must be elements of $\mathcal{S}$. M (and similarly $-M$) therefore leaves $\mathcal{S}$ invariant and $|\psi\rangle$ unmodified. The probability of measuring M in the $+1$ eigenstate $|\psi\rangle$ is hence unity.

For the second case, let us assume without loss of generality that M anti-commutes with S_1, and commutes with all other $S_i \in \mathcal{S}$, where $i \neq 1$. As $S_1 |\psi\rangle = |\psi\rangle$ the measurement probabilities for M are transformed as follows:

$$p(+1) = \mathrm{tr}\left[P_+ |\psi\rangle\right] = \mathrm{tr}\left[P_+ S_1 |\psi\rangle\right] = \mathrm{tr}\left[S_1 P_- |\psi\rangle\right] = p(-1) \tag{9.8}$$

As $p(+1) = p(-1)$ it follows that $p(+1) = p(-1) = \frac{1}{2}$. If we obtain the eigenvalue $+1$ for a M measurement then stabiliser set specifying the new state becomes $\langle M, S_2, ..., S_r\rangle$, for -1 we have $\langle -M, S_2, ..., S_r\rangle$.

As we are checking through all the $2r$ commutation relations, each taking $\mathcal{O}(r)$ time to perform, the act of performing a Pauli group measurement takes $\mathcal{O}(r^2)$ time.

9.4 Entanglement breaking operations

Entanglement breaking operations were used in [42] where extensions of separable machines were considered. They are characterised as follows; suppose we have a composite quantum system composed of sub-systems denoted A and B. A two system trace preserving map $\mathcal{E}_{AB}$ is said to be *entanglement breaking* if the operation has the following CJ state representation:

$$\rho(\mathcal{E}_{AB}) = \sum_j p_j \rho_j^{AB} \otimes \rho_j^{R_A R_B} \tag{9.9}$$

In other words the CJ state representing the operation is separable across the $AB : R_A R_B$ cut.

9.5 Proof for lemma (4.4.4)

Proof. The extremal "pure" cube product states represented by the corners of the Bloch cube take the following form in table notation:

$$\frac{1}{4}\begin{Bmatrix} 1 & x_2 & y_2 & z_2 \\ x_1 & x_1x_2 & x_1y_2 & x_1z_2 \\ y_1 & y_1x_2 & y_1y_2 & x_1z_2 \\ z_1 & z_1x_2 & z_1y_2 & z_1z_2 \end{Bmatrix} \tag{9.10}$$

where the coefficients x_k, y_k, z_k $(k = 1, 2)$ take the values of ± 1. The maximally quantum entangled Bell state $|\psi_+\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$, can through explicit calculation by expressing the basis state dyads (i.e. $|00\rangle\langle 00|$, $|01\rangle\langle 00|$, etc.) as Pauli tensor products, be shown to correspond to the decomposition table:

$$\frac{1}{4}\begin{Bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 \end{Bmatrix} \tag{9.11}$$

Within this table representation it is easy to see that the Bell state $|\psi_+\rangle$ is cube-separable as it can be formed by a uniform average of corner cube extrema states satisfying $x_1 = x_2$, $y_1 = -y_2$. The remaining Bell quantum states may also be shown to be cube-separable as they differ from each other by only local Pauli unitaries acting on each side. □

9.6 CZ gate leads to negative probabilities

The following example shows that the noiseless CZ gate can take products of operators from the Bloch cube to two particle operators that return negative probabilities under Pauli measurements:

Example 9.6.1. ***Negative probabilities from cube input.***

Let us consider the CZ output state in eq. (4.25), the probability of measuring $X \otimes X$ and obtaining $+1$ on one particle and -1 on the other is:

$$\frac{1}{4}(1 + xC - yB - zA) \tag{9.12}$$

If we choose to input a product of pure cube states with $x = A = 1$, $y = B = 1$, $z = 1$, and $C = -1$, the probability of getting the pair of outcomes is $-\frac{1}{2}$.

We must therefore add noise to the CZ gate not only to ensure separability, but also at very least to ensure measurement positivity. Here we will expand on this example and prove the following:

Theorem 9.6.2. ***Negative probabilities from input outside $\mathcal{Q}$.***

Let $\mathcal{M}$ be the set of Pauli X, Y, Z measurements, and our choice of $\mathcal{S}$ satisfy $\mathcal{Q} \subset \mathcal{S} \subseteq \mathcal{M}^*$, where $\mathcal{Q}$ is the Bloch sphere.
The noiseless CZ gate acts to take all input product operators $\varrho_A \otimes \varrho_B \in \mathcal{S} \otimes \mathcal{S}$, where ϱ_A,$\varrho_B \notin \mathcal{Q}$, to operators that return negative Pauli measurement probabilities.

Proof. Let us consider eq. (9.12) again; for consistency we require that this probability be positive. As $\mathcal{S} \supset \mathcal{Q}$ we are free to fix the vector $V = (C, -B, -A)$ to be any unit vector taken from $\mathcal{Q}$, i.e. A, B, C can be components of any unit Bloch vector. Setting the vector $v = (x, y, z)$, we can express eq. (9.12) as

$$1 + v \cdot V \geq 0 \tag{9.13}$$

This can only be positive for all choices of V, if v corresponds to a Bloch vectors in $\mathcal{Q}$. Therefore for separable input, the CZ gate can only output states that yield positive measurement probabilities if these input states are in the Bloch sphere. □

9.7 Matrices from section (6.6.2)

The $\boldsymbol{B}_{num,1}$ matrix for the product input state ϱ_1 where $c = 0.7$:

$$\begin{bmatrix}
0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\
0.7 & 1 & -1 & 0.7 & -0.7 & -1 & 1 & 1 & 1 & 0.7 & 1 & -0.7 & -0.7 & 1 & 1 & 1 \\
1 & 1 & 0.7 & -1 & -1 & 1 & -0.7 & -0.7 & 1 & 1 & 1 & -1 & 1 & -1 & -0.7 & 1 \\
1 & 0.7 & 1 & 1 & -1 & 0.7 & -1 & -1 & 0.7 & -1 & -0.7 & 1 & -1 & 0.7 & -1 & -0.7 \\
0.7 & 0.7 & 1 & 1 & 1 & -1 & 1 & 1 & -0.7 & -0.7 & -1 & 1 & -1 & 1 & 1 & 1 \\
1 & 0.7 & -1 & 0.7 & -0.7 & 1 & 1 & 1 & -0.7 & -0.49 & -1 & -0.7 & 0.7 & 1 & 1 & 1 \\
-0.7 & 0.7 & 0.7 & -1 & -1 & -1 & -0.7 & -0.7 & -0.7 & -0.7 & -1 & -1 & -1 & -1 & -0.7 & 1 \\
0.7 & 0.49 & 1 & 1 & -1 & -0.7 & -1 & -1 & -0.49 & 0.7 & 0.7 & 1 & 1 & 0.7 & -1 & -0.7 \\
1 & 1 & 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 & -1 & 1 & 1 & -0.7 & -0.7 & -1 \\
-0.7 & 1 & -1 & 0.7 & -0.7 & -1 & -1 & -1 & -1 & -0.7 & -1 & -0.7 & -0.7 & -0.7 & -0.7 & -1 \\
0.49 & 1 & 0.7 & -1 & -1 & 1 & 0.7 & 0.7 & -1 & -1 & -1 & -1 & 1 & 0.7 & 0.49 & -1 \\
1 & 0.7 & 1 & 1 & -1 & 0.7 & 1 & 1 & -0.7 & 1 & 0.7 & 1 & -1 & -0.49 & 0.7 & 0.7 \\
1 & 1 & 0.7 & 0.7 & 0.7 & 0.7 & 0.7 & 0.7 & 1 & 1 & 0.7 & -0.7 & -0.7 & -1 & -1 & -0.7 \\
0.7 & 1 & -0.7 & 0.49 & -0.49 & -0.7 & 0.7 & 0.7 & 1 & 0.7 & 0.7 & 0.49 & 0.49 & -1 & -1 & -0.7 \\
1 & 1 & 0.49 & -0.7 & -0.7 & 0.7 & -0.49 & -0.49 & 1 & 1 & 0.7 & 0.7 & -0.7 & 1 & 0.7 & -0.7 \\
1 & 0.7 & 0.7 & 0.7 & -0.7 & 0.49 & -0.7 & -0.7 & 0.7 & -1 & -0.49 & -0.7 & 0.7 & -0.7 & 1 & 0.49
\end{bmatrix} \tag{9.14}$$

The constructed $\boldsymbol{D}_{s,1}$ matrix for the product input state ϱ_1 where $c = 0.7$:

$$\begin{bmatrix}
-2 & -1+c & -1-c & -2 & 0 & 0 & 0 & -1+c & 0 & -1-c & -1-c & 0 & 0 & 0 \\
-1+c & -2 & -2 & 0 & 0 & -1-c & 0 & 0 & 0 & -2 & 0 & -2 & -1-c & 0 \\
1-c & 1-c & -1-c & 0 & 0 & -1-c & 0 & -1-c & -2c & 1-c & -1-c & 0 & -1-c & -2c \\
1-c & 1-c & 1-c & -1-c & 1-c & 1-c & -2c & -2c & -1-c & 1-c & -1-c & 1-c & 1-c & 1-c \\
-1-c & 0 & -2c & 1-c & 1-c & 1-c & -2c & -c-c^2 & -1-c & -2c & 0 & 1-c & 1-c & 1-c \\
0 & -1-c & -1-c & -1-c & 1-c & -2c & -2c & -2c & -1-c & -1-c & -1-c & -1-c & -2c & 1-c \\
1-c^2 & 1-c^2 & -1-c^2 & -c-c^2 & c-c^2 & -1-c^2 & -2c^2 & c-c^2 & c-c^2 & 1-c^2 & 1-c^2 & c-c^2 & -1-c^2 & -c-c^2 \\
0 & 0 & 0 & 0 & -1-c & -2 & -2 & -2 & -2 & 0 & 0 & -1-c & -1-c & -2 \\
-2 & -1+c & -1-c & -2 & -1-c & -2 & -2 & -1-c & -2 & -1-c & -1-c & -1-c & -1-c & -2 \\
-1+c & -2 & -2 & 0 & -1-c & -1+c & -2 & -2 & -2 & -2 & 0 & -1+c & -1+c^2 & -2 \\
1-c & 1-c & -1-c & 0 & -c-c^2 & 1-c & -2c & 1-c & 0 & 1-c & -1-c & -c-c^2 & 0 & 0 \\
-1+c & -1+c & -1+c & -1+c & 0 & -1+c & 0 & 0 & -1+c & -1-c & -1-c & -2 & -2 & -1-c \\
-1-c & -1+c^2 & -1-c^2 & -1-c & 0 & -1+c & 0 & -1+c & -1+c & -1+c^2 & -1+c^2 & -2 & -2 & -1-c \\
-1+c^2 & -1-c & -1-c & -1+c & 0 & -1-c^2 & 0 & 0 & -1+c & -1+c & -1-c & 0 & -1+c & -1-c \\
0 & 0 & -2c & -c+c^2 & 0 & -2c & 0 & -1-c & -c-c^2 & -2c & 0 & -2c & 1-c & -c+c^2
\end{bmatrix} \tag{9.15}$$

9.8 Obtaining symbolic coefficients

For the product input ϱ_1; the corresponding $\boldsymbol{B}_{s,1}$ matrix for $c = 0.5$ is:

$$\begin{bmatrix}
0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\
c & 1 & 1 & c & -c & 1 & 1 & c & -1 & c & -1 & -1 & 1 & 1 & 1 & 1 \\
1 & 1 & c & -1 & 1 & 1 & 1 & -1 & -1 & -1 & 1 & 1 & c & c & -c & 1 \\
1 & c & 1 & 1 & 1 & c & -c & 1 & c & -1 & c & -c & -1 & -1 & -1 & -c \\
c & c & 1 & 1 & -c & 1 & -1 & 1 & 1 & 1 & -c & -1 & c & 1 & 1 & -1 \\
1 & c & 1 & c & c^2 & 1 & -1 & c & -1 & c & c & 1 & c & 1 & 1 & -1 \\
-c & c & c & -1 & -c & 1 & -1 & -1 & -1 & -1 & -c & -1 & c^2 & c & -c & -1 \\
c & c^2 & 1 & 1 & -c & c & c & 1 & c & -1 & -c^2 & c & -c & -1 & -1 & c \\
1 & 1 & c & 1 & 1 & -c & -1 & 1 & 1 & 1 & 1 & 1 & -1 & -1 & -1 & -c \\
-c & 1 & c & c & -c & -c & -1 & c & -1 & c & -1 & -1 & -1 & -1 & -1 & -c \\
c^2 & 1 & c^2 & -1 & 1 & -c & -1 & -1 & -1 & -1 & 1 & 1 & -c & -c & c & -c \\
1 & c & c & 1 & 1 & -c^2 & c & 1 & c & -1 & c & -c & 1 & 1 & 1 & c^2 \\
1 & 1 & 1 & c & 1 & 1 & c & -c & -c & -c & -1 & -c & -1 & -c & -c & -1 \\
c & 1 & 1 & c^2 & -c & 1 & c & -c^2 & c & -c^2 & 1 & c & -1 & -c & -c & -1 \\
1 & 1 & c & -c & 1 & 1 & c & c & c & c & -1 & -c & -c & -c^2 & c^2 & -1 \\
1 & c & 1 & c & 1 & c & -c^2 & -c & -c^2 & c & -c & c^2 & 1 & c & c & c
\end{bmatrix} \tag{9.16}$$

The symbolic basic variable vector is:

$$\begin{pmatrix}
\frac{2c}{1+2c} \\
\frac{-2+28c-114c^2+123c^3-65c^4+136c^5+172c^6+110c^7+100c^8-12c^9+30c^{10}-c^{11}+7c^{12}}{2(4+3c+61c^2+12c^3-178c^4+156c^5+46c^7-10c^8-79c^9-13c^{10}-10c^{11}+8c^{12})} \\
-\frac{2c\left(-9+8c-39c^2+54c^3+46c^4+102c^5+14c^6+70c^7-13c^8+18c^9+c^{10}+4c^{11}\right)}{(-1+c)(1+c)(1+2c)(-4+c-70c^2+60c^3-22c^4-14c^5-30c^6-44c^7-6c^8-3c^9+4c^{10})} \\
-\frac{-9+12c-40c^2+124c^3-14c^4+124c^5+28c^6+100c^7+31c^8+24c^9+4c^{10}}{2(1+2c)(-4+c-70c^2+60c^3-22c^4-14c^5-30c^6-44c^7-6c^8-3c^9+4c^{10})} \\
-\frac{4-3c+56c^2+11c^3-104c^4+6c^5+4c^6+70c^7+28c^8+37c^9+12c^{10}+7c^{11}}{(1+c)(1+2c)(-4+c-70c^2+60c^3-22c^4-14c^5-30c^6-44c^7-6c^8-3c^9+4c^{10})} \\
-\frac{-1-35c+28c^2+56c^3+248c^4+18c^5+324c^6+24c^7+139c^8-55c^9+24c^{10}-8c^{11}+6c^{12}}{2(1+c)(1+2c)(-4+c-70c^2+60c^3-22c^4-14c^5-30c^6-44c^7-6c^8-3c^9+4c^{10})} \\
\frac{-4+15c-162c^2+17c^3+110c^4+654c^5+28c^6+90c^7-280c^8-45c^9+38c^{10}+37c^{11}+14c^{12}}{4(1+c)(1+2c)(-4+c-70c^2+60c^3-22c^4-14c^5-30c^6-44c^7-6c^8-3c^9+4c^{10})} \\
-\frac{26-36c+179c^2-185c^3-397c^4-281c^5+860c^6+226c^7+604c^8-26c^9+7c^{10}+39c^{11}+7c^{12}+7c^{13}-6c^{14}}{4(1+c)(1+2c)(-4+c-70c^2+60c^3-22c^4-14c^5-30c^6-44c^7-6c^8-3c^9+4c^{10})} \\
\frac{12-59c+213c^2-295c^3-95c^4+22c^5+162c^6-270c^7+94c^8-367c^9+105c^{10}-43c^{11}+21c^{12}-12c^{13}}{4(1+c)(1+2c)(-4+c-70c^2+60c^3-22c^4-14c^5-30c^6-44c^7-6c^8-3c^9+4c^{10})} \\
\frac{-8+44c-203c^2+398c^3-423c^4+190c^5+362c^6+32c^7+418c^8-116c^9+97c^{10}-30c^{11}+13c^{12}-6c^{13}}{4(1+2c)(-4+c-70c^2+60c^3-22c^4-14c^5-30c^6-44c^7-6c^8-3c^9+4c^{10})} \\
\frac{c\left(7+7c-69c^2-48c^3+142c^4+46c^5+126c^6-8c^7+43c^8+3c^9+7c^{10}\right)}{2(1+2c)(-4+c-70c^2+60c^3-22c^4-14c^5-30c^6-44c^7-6c^8-3c^9+4c^{10})} \\
-\frac{4+4c+101c^2-249c^3+20c^4+220c^5+238c^6+186c^7+32c^8-24c^9-11c^{10}-9c^{11}}{4(1+2c)(-4+c-70c^2+60c^3-22c^4-14c^5-30c^6-44c^7-6c^8-3c^9+4c^{10})} \\
0 \\
-\frac{4+30c+17c^2-38c^3-349c^4+82c^5-8c^6+304c^7+54c^8+302c^9+45c^{10}+86c^{11}-13c^{12}+2c^{13}-6c^{14}}{4c(1+c)(1+2c)(-4+c-70c^2+60c^3-22c^4-14c^5-30c^6-44c^7-6c^8-3c^9+4c^{10})} \\
\frac{4+30c-17c^2+12c^3-397c^4+202c^5-56c^6+164c^7-162c^8-98c^9-157c^{10}-64c^{11}+11c^{12}+10c^{13}+6c^{14}}{4c(1+c)(1+2c)(-4+c-70c^2+60c^3-22c^4-14c^5-30c^6-44c^7-6c^8-3c^9+4c^{10})} \\
\frac{3+61c^2-66c^3-116c^4-256c^5+158c^6+84c^7+183c^8-27c^{10}-18c^{11}-6c^{12}}{2(1+c)(1+2c)(-4+c-70c^2+60c^3-22c^4-14c^5-30c^6-44c^7-6c^8-3c^9+4c^{10})}
\end{pmatrix} \tag{9.17}$$

For ϱ_2; the corresponding $\boldsymbol{B}_{s,\,2}$ for $c = 0.5$ is:

$$\begin{bmatrix}
0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\
1 & c & -1 & 1 & 1 & -1 & 1 & 1 & 1 & -c & -1 & -1 & -1 & 1 & 1 & 1 \\
c & 1 & -c & -c & 1 & 1 & 1 & c & 1 & -1 & -c & 1 & c & c & -1 & -c \\
1 & 1 & 1 & 1 & -c & -c & c & -1 & c & 1 & -1 & c & 1 & -1 & -c & -1 \\
c & c & c & 1 & -1 & -1 & c & -c & -1 & 1 & 1 & -c & -1 & 1 & 1 & -1 \\
c & c^2 & -c & 1 & -1 & 1 & c & -c & -1 & -c & -1 & c & 1 & 1 & 1 & -1 \\
-1 & c & -c^2 & -c & -1 & -1 & c & -c^2 & -1 & -1 & -c & -c & -c & c & -1 & c \\
c & c & c & 1 & c & c & c^2 & c & -c & 1 & -1 & -c^2 & -1 & -1 & -c & 1 \\
1 & 1 & 1 & c & 1 & 1 & -1 & -1 & -c & 1 & 1 & 1 & 1 & -1 & -1 & -c \\
-c^2 & c & -1 & c & 1 & -1 & -1 & -1 & -c & -c & -1 & -1 & -1 & -1 & -1 & -c \\
c & 1 & -c & -c^2 & 1 & 1 & -1 & -c & -c & -1 & -c & 1 & c & -c & 1 & c^2 \\
1 & 1 & 1 & c & -c & -c & -c & 1 & -c^2 & 1 & -1 & c & 1 & 1 & c & c \\
1 & 1 & 1 & 1 & c & c & 1 & 1 & 1 & -c & -c & -1 & -c & -c & -c & -1 \\
1 & c & -1 & 1 & c & -c & 1 & 1 & 1 & c^2 & c & 1 & c & -c & -c & -1 \\
c & 1 & -c & -c & c & c & 1 & c & 1 & c & c^2 & -1 & -c^2 & -c^2 & c & c \\
1 & 1 & 1 & 1 & -c^2 & -c^2 & c & -1 & c & -c & c & -c & -c & c & c^2 & 1
\end{bmatrix} \tag{9.18}$$

The symbolic basic variable vector is:

$$\begin{pmatrix} \frac{1+c^2}{2+c^2} \\ \frac{-24+178c+72c^2+609c^3+366c^4-3c^5+216c^6-402c^7-126c^8+220c^9-176c^{10}+125c^{11}-54c^{12}+37c^{13}-16c^{14}+4c^{15}-2c^{16}}{2(1+c)^2(2+c^2)(36+57c+166c^2-46c^3+156c^4-190c^5+56c^6+42c^7-40c^8+51c^9-46c^{10}+20c^{11}-8c^{12}+2c^{13})} \\ \frac{48-276c+532c^2+59c^3+302c^4+587c^5-122c^6+1002c^7+374c^8-606c^9+336c^{10}-225c^{11}+58c^{12}-25c^{13}+6c^{14}-4c^{15}+2c^{16}}{4(1+c)^2(2+c^2)(36+57c+166c^2-46c^3+156c^4-190c^5+56c^6+42c^7-40c^8+51c^9-46c^{10}+20c^{11}-8c^{12}+2c^{13})} \\ \frac{60+127c-47c^2+51c^3-197c^4+129c^5-148c^6+c^7+55c^8-100c^9+118c^{10}-72c^{11}+30c^{12}-8c^{13}+c^{14}}{2(2+c^2)(36+57c+166c^2-46c^3+156c^4-190c^5+56c^6+42c^7-40c^8+51c^9-46c^{10}+20c^{11}-8c^{12}+2c^{13})} \\ \frac{c\left(88+524c-423c^2+341c^3+115c^4-317c^5+924c^6-148c^7-82c^8+178c^9-271c^{10}+157c^{11}-89c^{12}+31c^{13}-6c^{14}+2c^{15}\right)}{4(1+c)(2+c^2)(36+57c+166c^2-46c^3+156c^4-190c^5+56c^6+42c^7-40c^8+51c^9-46c^{10}+20c^{11}-8c^{12}+2c^{13})} \\ \frac{-48+184c-316c^2+549c^3+304c^4-432c^5+233c^6-634c^7-108c^8+372c^9-222c^{10}+189c^{11}-92c^{12}+28c^{13}-7c^{14}}{4(1+c)(2+c^2)(36+57c+166c^2-46c^3+156c^4-190c^5+56c^6+42c^7-40c^8+51c^9-46c^{10}+20c^{11}-8c^{12}+2c^{13})} \\ \frac{-120+98c+328c^2-274c^3+645c^4-806c^5+746c^6+84c^7-332c^8+332c^9-282c^{10}+148c^{11}-81c^{12}+32c^{13}-8c^{14}+2c^{15}}{4(2+c^2)(36+57c+166c^2-46c^3+156c^4-190c^5+56c^6+42c^7-40c^8+51c^9-46c^{10}+20c^{11}-8c^{12}+2c^{13})} \\ -\frac{(-1+c)^2\left(24-170c+74c^2-23c^3+307c^4+107c^5+59c^6-57c^7-27c^8-47c^9+7c^{10}-4c^{11}+4c^{12}+2c^{13}\right)}{4(1+c)(2+c^2)(36+57c+166c^2-46c^3+156c^4-190c^5+56c^6+42c^7-40c^8+51c^9-46c^{10}+20c^{11}-8c^{12}+2c^{13})} \\ \frac{48-54c+46c^2+9c^3+93c^4+18c^5-158c^6+5c^7-45c^8+c^9+59c^{10}-34c^{11}+20c^{12}-9c^{13}+c^{14}}{(1+c)(2+c^2)(36+57c+166c^2-46c^3+156c^4-190c^5+56c^6+42c^7-40c^8+51c^9-46c^{10}+20c^{11}-8c^{12}+2c^{13})} \\ \frac{12+19c+10c^2-2c^3+94c^4-88c^5-85c^6+65c^7-89c^8+112c^9-65c^{10}+21c^{11}-5c^{12}+c^{13}}{2(2+c^2)(36+57c+166c^2-46c^3+156c^4-190c^5+56c^6+42c^7-40c^8+51c^9-46c^{10}+20c^{11}-8c^{12}+2c^{13})} \\ -\frac{c\left(-208+318c-155c^2+162c^3+201c^4-564c^5+384c^6+40c^7-346c^8+266c^9-135c^{10}+38c^{11}+c^{12}-4c^{13}+2c^{14}\right)}{4(1+c)(2+c^2)(36+57c+166c^2-46c^3+156c^4-190c^5+56c^6+42c^7-40c^8+51c^9-46c^{10}+20c^{11}-8c^{12}+2c^{13})} \\ \frac{24-218c+390c^2-164c^3+567c^4+727c^5-212c^6+686c^7-60c^8-272c^9+194c^{10}-188c^{11}+99c^{12}-53c^{13}+20c^{14}-6c^{15}+2c^{16}}{4(1+c)(2+c^2)(36+57c+166c^2-46c^3+156c^4-190c^5+56c^6+42c^7-40c^8+51c^9-46c^{10}+20c^{11}-8c^{12}+2c^{13})} \\ \frac{60+67c+178c^2-223c^3-130c^4-251c^5-332c^6+135c^7-144c^8+165c^9-94c^{10}+87c^{11}-42c^{12}+19c^{13}-8c^{14}+c^{15}}{4(1+c)(2+c^2)(36+57c+166c^2-46c^3+156c^4-190c^5+56c^6+42c^7-40c^8+51c^9-46c^{10}+20c^{11}-8c^{12}+2c^{13})} \\ \frac{(-1+c)^2\left(84+33c+394c^2+371c^3+480c^4+351c^5+358c^6-170c^7-58c^8-25c^9-40c^{10}+15c^{11}-2c^{12}+c^{13}\right)}{4(1+c)(2+c^2)(36+57c+166c^2-46c^3+156c^4-190c^5+56c^6+42c^7-40c^8+51c^9-46c^{10}+20c^{11}-8c^{12}+2c^{13})} \\ -\frac{24+10c-332c^2+6c^3-559c^4+626c^5-692c^6+426c^7+174c^8-410c^9+368c^{10}-238c^{11}+113c^{12}-34c^{13}+8c^{14}-2c^{15}}{4(2+c^2)(36+57c+166c^2-46c^3+156c^4-190c^5+56c^6+42c^7-40c^8+51c^9-46c^{10}+20c^{11}-8c^{12}+2c^{13})} \\ \frac{c\left(44-125c+180c^2-197c^3+5c^4+188c^5-255c^6+222c^7-22c^8-117c^9+138c^{10}-89c^{11}+37c^{12}-10c^{13}+c^{14}\right)}{2(1+c)(2+c^2)(36+57c+166c^2-46c^3+156c^4-190c^5+56c^6+42c^7-40c^8+51c^9-46c^{10}+20c^{11}-8c^{12}+2c^{13})} \end{pmatrix} \tag{9.19}$$

We can alternatively guess at the coefficients for ϱ_2 provided the CZ gate is undergoing noise at a rate of $\varepsilon = \frac{1+c^2}{2+c^2}$, where $c = 0.5$. The noisy CZ output state for ϱ_2 in this instance can be given as

$$\frac{1}{4}\left(\frac{1}{2+c^2}\right)\begin{Bmatrix} 2+c^2 & 1 & c & 1 \\ c & c & -1 & c \\ 1 & -c^2 & c & 1 \\ 1 & 1 & c & 1 \end{Bmatrix} \tag{9.20}$$

This output state can be expressed as a convex combination of the separable states:

$$\frac{1}{4}\left(\frac{1}{2+c^2}\right)\begin{Bmatrix} 1 & 1 & c & 1 \\ c & c & c^2 & c \\ 1 & 1 & c & 1 \\ 1 & 1 & c & 1 \end{Bmatrix} + \frac{1}{4}\left(\frac{1+c^2}{2+c^2}\right)\begin{Bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{Bmatrix} \tag{9.21}$$

For ϱ_3; the corresponding $\boldsymbol{B}_{s,3}$ for $c = 0.5$ is:

$$\begin{bmatrix}
0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\
1 & 1 & 1 & 1 & c & 1 & -1 & 1 & 1 & 1 & -1 & -1 & -1 & 1 & c & -1 \\
1 & 1 & -c & c & 1 & 1 & 1 & -1 & -c & 1 & -1 & 1 & 1 & -1 & -1 & 1 \\
c & -c & 1 & 1 & 1 & -c & c & -c & 1 & -c & c & -c & c & c & -1 & -c \\
c^2 & c & 1 & 1 & -1 & -1 & -1 & 1 & 1 & -c & 1 & -c & -1 & 1 & 1 & -1 \\
1 & c & 1 & 1 & -c & -1 & 1 & 1 & 1 & -c & -1 & c & 1 & 1 & c & 1 \\
-1 & c & -c & c & -1 & -1 & -1 & -1 & -c & -c & -1 & -c & -1 & -1 & -1 & -1 \\
c & -c^2 & 1 & 1 & -1 & c & -c & -c & 1 & c^2 & c & c^2 & -c & c & -1 & c \\
c & 1 & c & 1 & c & c & 1 & -c & -1 & -1 & c & 1 & 1 & -c & -c & -c \\
-c & 1 & c & 1 & c^2 & c & -1 & -c & -1 & -1 & -c & -1 & -1 & -c & -c^2 & c \\
c & 1 & -c^2 & c & c & c & 1 & c & c & -1 & -c & 1 & 1 & c & c & -c \\
1 & -c & c & 1 & c & -c^2 & c & c^2 & -1 & c & c^2 & -c & c & -c^2 & c & c^2 \\
1 & 1 & 1 & c & 1 & 1 & c & 1 & c & 1 & -1 & -1 & -c & -1 & -1 & -1 \\
1 & 1 & 1 & c & c & 1 & -c & 1 & c & 1 & 1 & 1 & c & -1 & -c & 1 \\
1 & 1 & -c & c^2 & 1 & 1 & c & -1 & -c^2 & 1 & 1 & -1 & -c & 1 & 1 & -1 \\
c & -c & 1 & c & 1 & -c & c^2 & -c & c & -c & -c & c & -c^2 & -c & 1 & c
\end{bmatrix} \tag{9.22}$$

The symbolic basic variable vector is:

$$\begin{pmatrix}
\frac{1+c^2}{2+c^2} \\
-\frac{(-1+c)^2\left(-1+3c^2+23c^3+24c^4+62c^5+12c^6+20c^7+17c^8+42c^9-23c^{10}+13c^{11}\right)}{4(1+c)(1+c^2)(2+c^2)(-2-5c+3c^2-15c^3+9c^4-27c^5+23c^6-25c^7+13c^8-8c^9+2c^{10})} \\
\frac{1+c-7c^2-27c^3+32c^4-20c^5+86c^6-80c^7+97c^8-37c^9+17c^{10}+3c^{11}-2c^{12}}{2c(1+c)(2+c^2)(-2-5c+3c^2-15c^3+9c^4-27c^5+23c^6-25c^7+13c^8-8c^9+2c^{10})} \\
-\frac{1+2c-3c^2+19c^4+6c^5+3c^6+52c^7-40c^8+36c^9-12c^{10}}{2c(2+c^2)(-2-5c+3c^2-15c^3+9c^4-27c^5+23c^6-25c^7+13c^8-8c^9+2c^{10})} \\
-\frac{1-3c+2c^2+4c^3+2c^4-19c^5-16c^6+23c^7-35c^8+10c^9-2c^{10}+c^{11}}{(-1+c)(2+c^2)(-2-5c+3c^2-15c^3+9c^4-27c^5+23c^6-25c^7+13c^8-8c^9+2c^{10})} \\
\frac{5-3c+c^2-9c^3+64c^4-82c^5+4c^6-258c^7+19c^8-115c^9+5c^{10}-209c^{11}+134c^{12}-88c^{13}+22c^{14}-4c^{15}+2c^{16}}{4(-1+c)(1+c)(1+c^2)(2+c^2)(-2-5c+3c^2-15c^3+9c^4-27c^5+23c^6-25c^7+13c^8-8c^9+2c^{10})} \\
-\frac{-1+4c-3c^2+5c^3+4c^4+33c^5+3c^6-21c^7+62c^8-43c^9+24c^{10}-8c^{11}+7c^{12}-2c^{13}}{4c^2(2+c^2)(-2-5c+3c^2-15c^3+9c^4-27c^5+23c^6-25c^7+13c^8-8c^9+2c^{10})} \\
\frac{1+3c-28c^2+49c^3-90c^4+186c^5-206c^6+322c^7-221c^8+139c^9-30c^{10}+5c^{11}-2c^{12}}{4(-1+c)(2+c^2)(-2-5c+3c^2-15c^3+9c^4-27c^5+23c^6-25c^7+13c^8-8c^9+2c^{10})} \\
\frac{1-10c^2+12c^3+26c^4+2c^5-32c^6-98c^7+65c^8-94c^9+18c^{10}-18c^{11}-4c^{12}+4c^{13}}{2(-1+c)c(1+c)(2+c^2)(-2-5c+3c^2-15c^3+9c^4-27c^5+23c^6-25c^7+13c^8-8c^9+2c^{10})} \\
-\frac{3+18c-16c^2+97c^3-101c^4+233c^5-168c^6+362c^7-235c^8+148c^9-92c^{10}+37c^{11}-35c^{12}+c^{13}+4c^{14}}{4(1+c)(1+c^2)(2+c^2)(-2-5c+3c^2-15c^3+9c^4-27c^5+23c^6-25c^7+13c^8-8c^9+2c^{10})} \\
\frac{c\left(-5+26c-50c^2+40c^3-56c^4+58c^5-110c^6+78c^7-59c^8+20c^9-8c^{10}+2c^{11}\right)}{4(2+c^2)(-2-5c+3c^2-15c^3+9c^4-27c^5+23c^6-25c^7+13c^8-8c^9+2c^{10})} \\
-\frac{1+c-6c^2+4c^3+3c^4-2c^5-46c^6-30c^7+47c^8-117c^9+58c^{10}-46c^{11}+5c^{12}-2c^{13}+2c^{14}}{2(-1+c)c(1+c)(2+c^2)(-2-5c+3c^2-15c^3+9c^4-27c^5+23c^6-25c^7+13c^8-8c^9+2c^{10})} \\
\frac{1-3c+7c^2-4c^3-3c^4+19c^5-52c^6+26c^7-169c^8+177c^9-193c^{10}+90c^{11}-37c^{12}+15c^{13}-2c^{14}}{4(-1+c)c^2(2+c^2)(-2-5c+3c^2-15c^3+9c^4-27c^5+23c^6-25c^7+13c^8-8c^9+2c^{10})} \\
\frac{-3+31c^2-45c^3-45c^4+44c^5+124c^6+38c^7-79c^8+238c^9-115c^{10}+101c^{11}-33c^{12}+6c^{13}-8c^{14}+2c^{15}}{4(-1+c)c(1+c)(2+c^2)(-2-5c+3c^2-15c^3+9c^4-27c^5+23c^6-25c^7+13c^8-8c^9+2c^{10})} \\
\frac{1+c-11c^2+29c^3-7c^4-8c^5-38c^6-16c^7+31c^8-141c^9+81c^{10}-61c^{11}+7c^{12}+4c^{13}}{2(-1+c)c(1+c)(2+c^2)(-2-5c+3c^2-15c^3+9c^4-27c^5+23c^6-25c^7+13c^8-8c^9+2c^{10})} \\
\frac{1-6c+6c^2+4c^3+9c^4-19c^5-c^6+37c^7-112c^8+71c^9-61c^{10}+7c^{11}-2c^{12}+2c^{13}}{4c(2+c^2)(-2-5c+3c^2-15c^3+9c^4-27c^5+23c^6-25c^7+13c^8-8c^9+2c^{10})}
\end{pmatrix} \tag{9.23}$$

For ϱ_4; the corresponding $\boldsymbol{B}_{s,\,4}$ for $c = 0.5$ is:

$$\begin{bmatrix}
0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\
1 & 1 & 1 & -c & c & 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 & -c & 1 & 1 \\
c & c & -c & 1 & 1 & 1 & 1 & -1 & 1 & 1 & -c & c & -1 & 1 & -1 & -1 \\
1 & 1 & 1 & 1 & 1 & c & -c & c & -c & -c & -1 & 1 & -c & -1 & -c & -c \\
1 & 1 & 1 & 1 & -c & -1 & -1 & 1 & -1 & c & c & 1 & 1 & -c & 1 & 1 \\
c^2 & 1 & 1 & -c & -c^2 & -1 & -1 & 1 & -1 & -c & -c & -1 & -1 & c^2 & 1 & 1 \\
-c & c & -c & 1 & -c & -1 & -1 & -1 & -1 & c & -c^2 & c & -1 & -c & -1 & -1 \\
1 & 1 & 1 & 1 & -c & -c & c & c & c & -c^2 & -c & 1 & -c & c & -c & -c \\
c & c & c & 1 & 1 & c & c & -1 & -1 & 1 & 1 & 1 & 1 & 1 & -c & -1 \\
-c & c & c & -c & c & c & c & -1 & -1 & -1 & -1 & -1 & -1 & -c & -c & -1 \\
1 & c^2 & -c^2 & 1 & 1 & c & c & 1 & -1 & 1 & -c & c & -1 & 1 & c & 1 \\
c & c & c & 1 & 1 & c^2 & -c^2 & -c & c & -c & -1 & 1 & -c & -1 & c^2 & c \\
1 & 1 & 1 & c & 1 & 1 & 1 & c & c & -1 & -1 & -c & -c & -1 & -1 & -c \\
1 & 1 & 1 & -c^2 & c & 1 & 1 & c & c & 1 & 1 & c & c & c & -1 & -c \\
c & c & -c & c & 1 & 1 & 1 & -c & c & -1 & c & -c^2 & c & -1 & 1 & c \\
1 & 1 & 1 & c & 1 & c & -c & c^2 & -c^2 & c & 1 & -c & c^2 & 1 & c & c^2
\end{bmatrix} \tag{9.24}$$

The symbolic basic variable vector is:

$$\begin{pmatrix}
\frac{2c}{1+2c} \\
\frac{-35+77c+127c^2-389c^3-78c^4+216c^5+112c^6+144c^7+63c^8+11c^9+c^{10}+5c^{11}+2c^{12}}{4(1+c)(1+2c)(-10+49c-90c^2+40c^3-6c^4-18c^5+10c^6-8c^7+c^9)} \\
\frac{15-48c+15c^2+44c^3-74c^4+66c^5+18c^6+2c^7+27c^8-2c^9-c^{10}+2c^{11}}{4(1+2c)(-10+49c-90c^2+40c^3-6c^4-18c^5+10c^6-8c^7+c^9)} \\
\frac{c\left(16-95c+93c^2-7c^3+61c^4-17c^5+19c^6-9c^7+3c^8\right)}{2(1+2c)(-10+49c-90c^2+40c^3-6c^4-18c^5+10c^6-8c^7+c^9)} \\
\frac{-10+69c-184c^2+168c^3-62c^4+100c^5+86c^6-10c^7+40c^8-9c^9+2c^{10}+2c^{11}}{2(-1+c)(1+2c)(-10+49c-90c^2+40c^3-6c^4-18c^5+10c^6-8c^7+c^9)} \\
-\frac{-10+48c-85c^2+38c^3-123c^4+250c^5-6c^6+156c^7+68c^8+20c^9+27c^{10}-2c^{11}+c^{12}+2c^{13}}{4(-1+c)c(1+2c)(-10+49c-90c^2+40c^3-6c^4-18c^5+10c^6-8c^7+c^9)} \\
\frac{5-34c+97c^2-95c^3-22c^4+48c^5-35c^6-26c^7+18c^8-22c^9+2c^{10}+c^{11}-c^{12}}{2c(1+2c)(-10+49c-90c^2+40c^3-6c^4-18c^5+10c^6-8c^7+c^9)} \\
-\frac{-9+70c-176c^2+90c^3+182c^4-10c^5+96c^6+6c^7+3c^8+4c^9}{4(1+2c)(-10+49c-90c^2+40c^3-6c^4-18c^5+10c^6-8c^7+c^9)} \\
-\frac{-20+117c-227c^2+54c^3+156c^4+64c^5+50c^6+74c^7-24c^8+11c^9+c^{10}}{4(1+2c)(-10+49c-90c^2+40c^3-6c^4-18c^5+10c^6-8c^7+c^9)} \\
\frac{5-19c+37c^2-31c^3-7c^4-11c^5-9c^6+3c^7-2c^8+2c^9}{(1+2c)(-10+49c-90c^2+40c^3-6c^4-18c^5+10c^6-8c^7+c^9)} \\
0 \\
-\frac{20-89c+123c^2-32c^3-58c^4+66c^5-28c^6+48c^7+6c^8+7c^9+c^{10}}{2(1+c)(1+2c)(-10+49c-90c^2+40c^3-6c^4-18c^5+10c^6-8c^7+c^9)} \\
-\frac{(-1+c)\left(20-116c+145c^2+44c^3-100c^4-8c^5-66c^6+4c^7+12c^9+c^{10}\right)}{4(1+c)(1+2c)(-10+49c-90c^2+40c^3-6c^4-18c^5+10c^6-8c^7+c^9)} \\
-\frac{10-37c+47c^2+29c^3-107c^4+39c^5-45c^6+3c^7+c^8-2c^9-2c^{10}}{2(1+2c)(-10+49c-90c^2+40c^3-6c^4-18c^5+10c^6-8c^7+c^9)} \\
\frac{-10+33c-52c^2+71c^3-182c^4+280c^5+2c^6+72c^7+48c^8-9c^9+2c^{10}+c^{11}}{4(-1+c)(1+2c)(-10+49c-90c^2+40c^3-6c^4-18c^5+10c^6-8c^7+c^9)} \\
-\frac{-29+141c-269c^2+93c^3+154c^4+6c^5+186c^6+102c^7+67c^8+45c^9+19c^{10}-3c^{11}}{4(-1+c)(1+c)(1+2c)(-10+49c-90c^2+40c^3-6c^4-18c^5+10c^6-8c^7+c^9)}
\end{pmatrix} \tag{9.25}$$

For ϱ_5; the corresponding $\boldsymbol{B}_{s,\,5}$ for $c = 0.5$ is:

$$\begin{bmatrix}
0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\
1 & -c & 1 & c & -1 & -c & 1 & 1 & 1 & 1 & -1 & -1 & -1 & 1 & 1 & 1 \\
1 & 1 & 1 & 1 & 1 & 1 & c & 1 & -1 & 1 & -1 & 1 & c & -1 & -1 & -1 \\
c & 1 & -c & 1 & -c & 1 & 1 & -c & c & -c & c & c & -1 & -c & c & -c \\
c & c & 1 & 1 & -c & -1 & c & c & 1 & -1 & c & -c & -c & 1 & 1 & -c \\
c & -c^2 & 1 & c & c & c & c & c & 1 & -1 & -c & c & c & 1 & 1 & -c \\
-c & c & 1 & 1 & -c & -1 & c^2 & c & -1 & -1 & -c & -c & -c^2 & -1 & -1 & c \\
1 & c & -c & 1 & c^2 & -1 & c & -c^2 & c & c & c^2 & -c^2 & c & -c & c & c^2 \\
c^2 & 1 & c & 1 & 1 & 1 & -1 & -1 & -1 & -c & 1 & 1 & 1 & -c & -1 & -1 \\
-1 & -c & c & c & -1 & -c & -1 & -1 & -1 & -c & -1 & -1 & -1 & -c & -1 & -1 \\
1 & 1 & c & 1 & 1 & 1 & -c & -1 & 1 & -c & -1 & 1 & c & c & 1 & 1 \\
c & 1 & -c^2 & 1 & -c & 1 & -1 & c & -c & c^2 & c & c & -1 & c^2 & -c & c \\
1 & 1 & 1 & c & 1 & c & 1 & 1 & c & 1 & -1 & -1 & -1 & -1 & -c & -1 \\
1 & -c & 1 & c^2 & -1 & -c^2 & 1 & 1 & c & 1 & 1 & 1 & 1 & -1 & -c & -1 \\
1 & 1 & 1 & c & 1 & c & c & 1 & -c & 1 & 1 & -1 & -c & 1 & c & 1 \\
c & 1 & -c & c & -c & c & 1 & -c & c^2 & -c & -c & -c & 1 & c & -c^2 & c
\end{bmatrix} \tag{9.26}$$

The symbolic basic variable vector is:

$$\begin{pmatrix}
\frac{1+c^2}{2+c^2} \\
\frac{1+c-7c^2-27c^3+32c^4-20c^5+86c^6-80c^7+97c^8-37c^9+17c^{10}+3c^{11}-2c^{12}}{2c(1+c)(2+c^2)(-2-5c+3c^2-15c^3+9c^4-27c^5+23c^6-25c^7+13c^8-8c^9+2c^{10})} \\
-\frac{(-1+c)^2\left(-1+3c^2+23c^3+24c^4+62c^5+12c^6+20c^7+17c^8+42c^9-23c^{10}+13c^{11}\right)}{4(1+c)(1+c^2)(2+c^2)(-2-5c+3c^2-15c^3+9c^4-27c^5+23c^6-25c^7+13c^8-8c^9+2c^{10})} \\
-\frac{1+2c-3c^2+19c^4+6c^5+3c^6+52c^7-40c^8+36c^9-12c^{10}}{2c(2+c^2)(-2-5c+3c^2-15c^3+9c^4-27c^5+23c^6-25c^7+13c^8-8c^9+2c^{10})} \\
\frac{1+3c-28c^2+49c^3-90c^4+186c^5-206c^6+322c^7-221c^8+139c^9-30c^{10}+5c^{11}-2c^{12}}{4(-1+c)(2+c^2)(-2-5c+3c^2-15c^3+9c^4-27c^5+23c^6-25c^7+13c^8-8c^9+2c^{10})} \\
\frac{1-10c^2+12c^3+26c^4+2c^5-32c^6-98c^7+65c^8-94c^9+18c^{10}-18c^{11}-4c^{12}+4c^{13}}{2(-1+c)c(1+c)(2+c^2)(-2-5c+3c^2-15c^3+9c^4-27c^5+23c^6-25c^7+13c^8-8c^9+2c^{10})} \\
-\frac{1-3c+2c^2+4c^3+2c^4-19c^5-16c^6+23c^7-35c^8+10c^9-2c^{10}+c^{11}}{(-1+c)(2+c^2)(-2-5c+3c^2-15c^3+9c^4-27c^5+23c^6-25c^7+13c^8-8c^9+2c^{10})} \\
\frac{5-3c+c^2-9c^3+64c^4-82c^5+4c^6-258c^7+19c^8-115c^9+5c^{10}-209c^{11}+134c^{12}-88c^{13}+22c^{14}-4c^{15}+2c^{16}}{4(-1+c)(1+c)(1+c^2)(2+c^2)(-2-5c+3c^2-15c^3+9c^4-27c^5+23c^6-25c^7+13c^8-8c^9+2c^{10})} \\
-\frac{-1+4c-3c^2+5c^3+4c^4+33c^5+3c^6-21c^7+62c^8-43c^9+24c^{10}-8c^{11}+7c^{12}-2c^{13}}{4c^2(2+c^2)(-2-5c+3c^2-15c^3+9c^4-27c^5+23c^6-25c^7+13c^8-8c^9+2c^{10})} \\
-\frac{3+18c-16c^2+97c^3-101c^4+233c^5-168c^6+362c^7-235c^8+148c^9-92c^{10}+37c^{11}-35c^{12}+c^{13}+4c^{14}}{4(1+c)(1+c^2)(2+c^2)(-2-5c+3c^2-15c^3+9c^4-27c^5+23c^6-25c^7+13c^8-8c^9+2c^{10})} \\
\frac{c\left(-5+26c-50c^2+40c^3-56c^4+58c^5-110c^6+78c^7-59c^8+20c^9-8c^{10}+2c^{11}\right)}{4(2+c^2)(-2-5c+3c^2-15c^3+9c^4-27c^5+23c^6-25c^7+13c^8-8c^9+2c^{10})} \\
\frac{-3+31c^2-45c^3-45c^4+44c^5+124c^6+38c^7-79c^8+238c^9-115c^{10}+101c^{11}-33c^{12}+6c^{13}-8c^{14}+2c^{15}}{4(-1+c)c(1+c)(2+c^2)(-2-5c+3c^2-15c^3+9c^4-27c^5+23c^6-25c^7+13c^8-8c^9+2c^{10})} \\
\frac{1+c-11c^2+29c^3-7c^4-8c^5-38c^6-16c^7+31c^8-141c^9+81c^{10}-61c^{11}+7c^{12}+4c^{13}}{2(-1+c)c(1+c)(2+c^2)(-2-5c+3c^2-15c^3+9c^4-27c^5+23c^6-25c^7+13c^8-8c^9+2c^{10})} \\
-\frac{1+c-6c^2+4c^3+3c^4-2c^5-46c^6-30c^7+47c^8-117c^9+58c^{10}-46c^{11}+5c^{12}-2c^{13}+2c^{14}}{2(-1+c)c(1+c)(2+c^2)(-2-5c+3c^2-15c^3+9c^4-27c^5+23c^6-25c^7+13c^8-8c^9+2c^{10})} \\
\frac{1-3c+7c^2-4c^3-3c^4+19c^5-52c^6+26c^7-169c^8+177c^9-193c^{10}+90c^{11}-37c^{12}+15c^{13}-2c^{14}}{4(-1+c)c^2(2+c^2)(-2-5c+3c^2-15c^3+9c^4-27c^5+23c^6-25c^7+13c^8-8c^9+2c^{10})} \\
\frac{1-6c+6c^2+4c^3+9c^4-19c^5-c^6+37c^7-112c^8+71c^9-61c^{10}+7c^{11}-2c^{12}+2c^{13}}{4c(2+c^2)(-2-5c+3c^2-15c^3+9c^4-27c^5+23c^6-25c^7+13c^8-8c^9+2c^{10})}
\end{pmatrix} \tag{9.27}$$

For ϱ_6; the corresponding $\boldsymbol{B}_{s,6}$ for $c = 0.5$ is:

$$\begin{bmatrix}
0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\
c & c & 1 & -1 & 1 & 1 & 1 & c & 1 & c & -1 & -1 & -c & -1 & 1 & -1 \\
c & 1 & c & 1 & -1 & c & 1 & 1 & 1 & 1 & -c & -1 & -1 & 1 & -1 & -c \\
c & 1 & 1 & c & c & -1 & -c & -1 & -c & -1 & 1 & c & 1 & -c & -c & -1 \\
c & c & 1 & -1 & 1 & -c & -c & -1 & -1 & -1 & c & 1 & 1 & -1 & 1 & -1 \\
1 & c^2 & 1 & 1 & 1 & -c & -c & -c & -1 & -c & -c & -1 & -c & 1 & 1 & 1 \\
-1 & c & c & -1 & -1 & -c^2 & -c & -1 & -1 & -1 & -c^2 & -1 & -1 & -1 & -1 & c \\
1 & c & 1 & -c & c & c & c^2 & 1 & c & 1 & c & c & 1 & c & -c & 1 \\
c & 1 & c & 1 & -1 & -1 & -1 & -c & -c & -1 & 1 & c & 1 & 1 & -1 & -c \\
-1 & c & c & -1 & -1 & -1 & -1 & -c^2 & -c & -c & -1 & -c & -c & -1 & -1 & c \\
1 & 1 & c^2 & 1 & 1 & -c & -1 & -c & -c & -1 & -c & -c & -1 & 1 & 1 & c^2 \\
1 & 1 & c & c & -c & 1 & c & c & c^2 & 1 & 1 & c^2 & 1 & -c & c & c \\
c & 1 & 1 & c & c & 1 & 1 & 1 & 1 & c & -1 & -1 & -c & -c & -c & -1 \\
1 & c & 1 & -c & c & 1 & 1 & c & 1 & c^2 & 1 & 1 & c^2 & c & -c & 1 \\
1 & 1 & c & c & -c & c & 1 & 1 & 1 & c & c & 1 & c & -c & c & c \\
c^2 & 1 & 1 & c^2 & c^2 & -1 & -c & -1 & -c & -c & -1 & -c & -c & c^2 & c^2 & 1
\end{bmatrix} \tag{9.28}$$

The symbolic basic variable vector is:

$$\begin{pmatrix}
\frac{4\left(3+2c+c^2\right)}{20+16c+9c^2+2c^3+c^4} \\
\frac{4\left(1+c+c^2\right)}{(1+c)(20+16c+9c^2+2c^3+c^4)} \\
\frac{c\left(-16-5c-6c^2+2c^3+c^5\right)}{(-2+c)(1+c)(20+16c+9c^2+2c^3+c^4)} \\
\frac{-12+27c+21c^2+12c^3+c^5-c^6}{4(1+c)(20+16c+9c^2+2c^3+c^4)} \\
\frac{-8-14c+c^2-6c^3+2c^4+c^6}{4(-40-12c-2c^2+5c^3+c^5)} \\
0 \\
\frac{2(-1+c)c^2}{-40-12c-2c^2+5c^3+c^5} \\
-\frac{2\left(4-2c+c^2\right)}{-40-12c-2c^2+5c^3+c^5} \\
0 \\
0 \\
-\frac{2\left(4-2c+c^2\right)}{-40-12c-2c^2+5c^3+c^5} \\
0 \\
\frac{2(-1+c)c^2}{-40-12c-2c^2+5c^3+c^5} \\
-\frac{8+54c+c^2-9c^3-8c^4+2c^5-c^6+c^7}{4(1+c)(-40-12c-2c^2+5c^3+c^5)} \\
-\frac{40+2c+11c^2-11c^3+12c^4-6c^5+c^6-c^7}{4(1+c)(-40-12c-2c^2+5c^3+c^5)} \\
\frac{-8+12c+5c^2+6c^3-2c^4-c^6}{(1+c)(-40-12c-2c^2+5c^3+c^5)}
\end{pmatrix} \tag{9.29}$$

Bibliography

[1] M. A. Nielsen, I. L. Chuang, Quantum Computation and Quantum Information, 10th Edition, Cambridge University Press, New York, NY, USA, 2011.

[2] S. Bravyi, A. Kitaev, Universal quantum computation with ideal clifford gates and noisy ancillas, Phys. Rev. A 71 (2005) 022316. doi:10.1103/PhysRevA.71.022316. URL http://link.aps.org/doi/10.1103/PhysRevA.71.022316

[3] D. Gottesman, Stabilizer codes and quantum error correction, Ph.D. thesis, California Institute of Technology (1997).

[4] J. Preskill, Fault tolerant quantum computation, in: Introduction to Quantum Computation, World Scientific, 1998. arXiv:quant-ph/9712048.
URL http://www.worldscientific.com/worldscibooks/10.1142/3724

[5] M. Plenio, S. Virmani, Upper bounds on fault tolerance thresholds of noisy clifford-based quantum computers, New Journal of Physics 12 (3) (2010) 033012. arXiv:0810.4340, doi:10.1088/1367-2630/12/3/033012.

[6] A. Harrow, M. Nielsen, Robustness of quantum gates in the presence of noise, Phys. Rev. A 68 (1) (2003) 012308. arXiv:quant-ph/0301108, doi:10.1103/PhysRevA.68.012308.

[7] N. Ratanje, S. Virmani, Generalized state spaces and nonlocality in fault-tolerant quantum-computing schemes, Phys. Rev. A 83 (2011) 032309. doi:10.1103/PhysRevA.83.032309.
URL http://link.aps.org/doi/10.1103/PhysRevA.83.032309

[8] N. Ratanje, S. Virmani, Exploiting non-quantum entanglement to widen applicability of limited-entanglement classical simulations of quantum systems (2012), arXiv:1201.0613.

[9] J. Audretsch, Entangled Systems: New Directions in Quantum Physics, Wiley-VCH Verlag GmbH, 2008. doi:10.1002/9783527619153.
URL http://dx.doi.org/10.1002/9783527619153

[10] J. Watrous, Theory of quantum information (2011).
URL https://cs.uwaterloo.ca/ watrous/TQI/TQI.pdf

[11] G. Kimura, A. Kossakowski, The bloch-vector space for n-level systems: the spherical-coordinate point of view, Open Systems & Information Dynamics 12 (3)

(2005) 207–229. doi:10.1007/s11080-005-0919-y.
URL http://dx.doi.org/10.1007/s11080-005-0919-y

[12] M. Horodecki, P. Horodecki, R. Horodecki, Separability of mixed states: necessary and sufficient conditions, Physics Letters A 223 (1996) 1–8. arXiv:quant-ph/9605038, doi:10.1016/S0375-9601(96)00706-2.

[13] A. Jamiołkowski, Linear transformations which preserve trace and positive semi-definiteness of operators, Reports on Mathematical Physics 3 (1972) 275–278. doi:10.1016/0034-4877(72)90011-0.

[14] M.-D. Choi, Completely positive linear maps on complex matrices, Linear Algebra and its Applications 10 (3) (1975) 285 – 290. doi:http://dx.doi.org/10.1016/0024-3795(75)90075-0.
URL http://www.sciencedirect.com/science/article/pii/0024379575900750

[15] B. Schumacher, Quantum coding, Phys. Rev. A 51 (1995) 2738–2747. doi:10.1103/PhysRevA.51.2738.
URL http://link.aps.org/doi/10.1103/PhysRevA.51.2738

[16] W. F. Stinespring, Positive functions on c^*-algebras, Proc. Amer. Math. Soc. 6(2) (1955) 211–216.

[17] M. Keyl, Fundamentals of quantum information theory, Physics Reports 369 (5) (2002) 431 – 548. doi:http://dx.doi.org/10.1016/S0370-1573(02)00266-1.
URL http://www.sciencedirect.com/science/article/pii/S0370157302002661

[18] D. Divincenzo, Two-bit gates are universal for quantum computation, Phys. Rev. A 51 (1995) 1015–1022. arXiv:cond-mat/9407022, doi:10.1103/PhysRevA.51.1015.

[19] P. O. Boykin, T. Mor, M. Pulver, V. Roychowdhury, F. Vatan, On universal and fault-tolerant quantum computing: A novel basis and a new constructive proof of universality for shor's basis, in: Proceedings of the 40th Annual Symposium on Foundations of Computer Science, FOCS '99, IEEE Computer Society, Washington, DC, USA, 1999, pp. 486–. arXiv:quant-ph/9906054.
URL http://dl.acm.org/citation.cfm?id=795665.796528

[20] C. Bennett, D. Divincenzo, J. Smolin, W. Wootters, Mixed-state entanglement and quantum error correction, Phys. Rev. A 54 (1996) 3824–3851. arXiv:quant-ph/9604024, doi:10.1103/PhysRevA.54.3824.

[21] E. Knill, R. Laflamme, L. Viola, Theory of quantum error correction for general noise, Physical Review Letters 84 (2000) 2525–2528. arXiv:quant-ph/9604034, doi:10.1103/PhysRevLett.84.2525.

[22] A. Ekert, C. Macchiavello, Quantum error correction for communication, Phys. Rev. Lett. 77 (1996) 2585–2588. doi:10.1103/PhysRevLett.77.2585.
URL http://link.aps.org/doi/10.1103/PhysRevLett.77.2585

[23] J. Preskill, Lecture notes for physics 229: Quantum information and computation, California Institute of Technology (1998) 16.
URL http://theory.caltech.edu/ preskill/ph219/

[24] D. Gottesman, The heisenberg representation of quantum computers, arXiv:quant-ph/9807006.

[25] M. Hein, W. Dür, J. Eisert, R. Raussendorf, M. Van den Nest, H. Briegel, Entanglement in graph states and its applications, Proceedings of the International School of Physics 'Enrico Fermi' on 'Quantum Computers, Algorithms and Chaos' 162 (2006) 115–218. arXiv:quant-ph/0602096, doi:10.3254/978-1-61499-018-5-115.

[26] Y. Shi, Both toffoli and controlled-not need little help to do universal quantum computation, arXiv:quant-ph/0205115arXiv:quant-ph/0205115.

[27] P. W. Shor, Fault-tolerant quantum computation, in: Foundations of Computer Science, 1996. Proceedings., 37th Annual Symposium on, 1996, pp. 56–65. doi:10.1109/SFCS.1996.548464.

[28] A. Kitaev, Quantum computations: algorithms and error correction, Russian Mathematical Surveys 52 (1997) 1191–1249. doi:10.1070/RM1997v052n06ABEH002155.

[29] D. Gottesman, Theory of fault-tolerant quantum computation, Phys. Rev. A 57 (1998) 127–137. arXiv:quant-ph/9702029, doi:10.1103/PhysRevA.57.127.

[30] D. Aharonov, M. Ben-Or, Fault-tolerant quantum computation with constant error rate, arXiv:quant-ph/9906129arXiv:quant-ph/9906129.

[31] A. W. Cross, D. P. Divincenzo, B. M. Terhal, A comparative code study for quantum fault tolerance, Quantum Info. Comput. 9 (7) (2009) 541–572.
URL http://dl.acm.org/citation.cfm?id=2011814.2011815

[32] P. Aliferis, D. Gottesman, J. Preskill, Quantum accuracy threshold for concatenated distance-3 codes, Quantum Info. Comput. 6 (2) (2006) 97–165.
URL http://dl.acm.org/citation.cfm?id=2011665.2011666

[33] B. W. Reichardt, Fault-tolerance threshold for a distance-three quantum code, in: M. Bugliesi, B. Preneel, V. Sassone, I. Wegener (Eds.), Automata, Languages and Programming, Vol. 4051 of Lecture Notes in Computer Science, Springer Berlin Heidelberg, 2006, pp. 50–61.

[34] P. Aliferis, A. W. Cross, Subsystem fault tolerance with the bacon-shor code, Phys. Rev. Lett. 98 (2007) 220502. doi:10.1103/PhysRevLett.98.220502.
URL http://link.aps.org/doi/10.1103/PhysRevLett.98.220502

[35] E. Knill, Fault-tolerant postselected quantum computation: Threshold analysis (2004), arXiv:quant-ph/0404104.

[36] B. W. Reichardt, Improved ancilla preparation scheme increases fault-tolerant threshold (2004), arXiv:quant-ph/0406025.

[37] P. Aliferis, D. Gottesman, J. Preskill, Accuracy threshold for postselected quantum computation, Quantum Info. Comput. 8 (3) (2008) 181–244. arXiv:quant-ph/0703264.
URL http://dl.acm.org/citation.cfm?id=2011763.2011764

[38] B. W. Reichardt, Error-detection-based quantum fault tolerance against discrete pauli noise, Ph.D. thesis, University of California, Berkeley (2006).

[39] C. Wang, J. Preskill, Confinement higgs transition in a disordered gauge theory and the accuracy threshold for quantum memory, Annals Phys. 303 (2003) 31–58. arXiv:quant-ph/0207088, doi:10.1016/S0003-4916(02)00019-2.

[40] A. G. Fowler, A. C. Whiteside, L. C. L. Hollenberg, Towards practical classical processing for the surface code, Phys. Rev. Lett. 108 (2012) 180501. doi:10.1103/PhysRevLett.108.180501.
URL http://link.aps.org/doi/10.1103/PhysRevLett.108.180501

[41] J. R. Wootton, D. Loss, High threshold error correction for the surface code, Phys. Rev. Lett. 109 (2012) 160503. doi:10.1103/PhysRevLett.109.160503.
URL http://link.aps.org/doi/10.1103/PhysRevLett.109.160503

[42] S. Virmani, S. Huelga, M. Plenio, Classical simulability, entanglement breaking, and quantum computation thresholds, Phys. Rev. A 71 (4) (2005) 042328. arXiv:quant-ph/0408076, doi:10.1103/PhysRevA.71.042328.

[43] H. Buhrman, R. Cleve, M. Laurent, N. Linden, A. Schrijver, F. Unger, New limits on fault-tolerant quantum computation, in: Foundations of Computer Science, 2006. FOCS '06. 47th Annual IEEE Symposium on, 2006, pp. 411–419. doi:10.1109/FOCS.2006.50.

[44] R. Jozsa, A. Miyake, Matchgates and classical simulation of quantum circuits, Proceedings of the Royal Society of London Series A 464 (2008) 3089–3106. arXiv:0804.4050, doi:10.1098/rspa.2008.0189.

[45] R. Jozsa, N. Linden, On the role of entanglement in quantum-computational speed-up, Proceedings of the Royal Society of London Series A 459 (2003) 2011–2032. arXiv:quant-ph/0201143, doi:10.1098/rspa.2002.1097.

[46] M. Van den Nest, Classical simulation of quantum computation, the gottesman-knill theorem, and slightly beyond, Quant. Inf. Comp. 10 (3-4) (2010) pp–0258.

[47] R. Jozsa, M. Van den Nest, Classical simulation complexity of extended clifford circuits, Quantum Information & Computation 14 (7-8) (2014) 633–648.
URL http://www.rintonpress.com/xxqic14/qic-14-78/0633-0648.pdf

[48] L. G. Valiant, Quantum circuits that can be simulated classically in polynomial time, SIAM Journal on Computing 31 (4) (2002) 1229–1254. arXiv:http://dx.doi.org/10.1137/S0097539700377025, doi:10.1137/S0097539700377025.
URL http://dx.doi.org/10.1137/S0097539700377025

[49] R. Jozsa, On the simulation of quantum circuits, arXiv:quant-ph/0603163arXiv:quant-ph/0603163.

[50] G. Vidal, Efficient classical simulation of slightly entangled quantum computations, Physical Review Letters 91 (14) (2003) 147902. arXiv:quant-ph/0301063, doi:10.1103/PhysRevLett.91.147902.

[51] I. Markov, Y. Shi, Simulating quantum computation by contracting tensor networks, SIAM Journal on Computing 38 (3) (2008) 963–981. arXiv:http://dx.doi.org/10.1137/050644756, doi:10.1137/050644756.
URL http://dx.doi.org/10.1137/050644756

[52] R. Jozsa, B. Kraus, A. Miyake, J. Watrous, Matchgate and space-bounded quantum computations are equivalent, Proceedings of the Royal Society of London Series A 466 (2009) 809–830. arXiv:0908.1467, doi:10.1098/rspa.2009.0433.

[53] V. Vedral, M. B. Plenio, M. A. Rippin, P. L. Knight, Quantifying entanglement, Phys. Rev. Lett. 78 (1997) 2275–2279. doi:10.1103/PhysRevLett.78.2275.
URL http://link.aps.org/doi/10.1103/PhysRevLett.78.2275

[54] H. Barnum, M. Nielsen, B. Schumacher, Information transmission through a noisy quantum channel, Phys. Rev. A 57 (1998) 4153–4175. arXiv:quant-ph/9702049, doi:10.1103/PhysRevA.57.4153.

[55] J. I. Cirac, W. Dür, B. Kraus, M. Lewenstein, Entangling operations and their implementation using a small amount of entanglement, Phys. Rev. Lett. 86 (2001) 544–547. doi:10.1103/PhysRevLett.86.544.
URL http://link.aps.org/doi/10.1103/PhysRevLett.86.544

[56] J. Barrett, Information processing in generalized probabilistic theories, Phys. Rev. A 75 (3) (2007) 032304. doi:10.1103/PhysRevA.75.032304.

[57] L. Hardy, Quantum theory from five reasonable axioms (2001), arXiv:quant-ph/0101012.

[58] R. P. Feynman, Simulating physics with computers, International Journal of Theoretical Physics 21 (6) (1982) 467–488. doi:10.1007/BF02650179.
URL http://dx.doi.org/10.1007/BF02650179

[59] J. S. Bell, On the einstein-podolsky-rosen paradox. physics 1, 195–200 (1964). reprinted in john stuart bell, speakable and unspeakable in quantum mechanics (2004).

[60] R. Werner, Quantum states with einstein-podolsky-rosen correlations admitting a hidden-variable model, Phys. Rev. A 40 (1989) 4277–4281. doi:10.1103/PhysRevA.40.4277.

[61] M. Horodecki, P. W. Shor, M. B. Ruskai, Entanglement breaking channels, Reviews in Mathematical Physics 15 (06) (2003) 629–641. arXiv:http://www.worldscientific.com/doi/pdf/10.1142/S0129055X03001709, doi:10.1142/S0129055X03001709.
URL http://www.worldscientific.com/doi/abs/10.1142/S0129055X03001709

[62] A. Peres, Separability criterion for density matrices, Phys. Rev. Lett. 77 (1996) 1413–1415. doi:10.1103/PhysRevLett.77.1413.
URL http://link.aps.org/doi/10.1103/PhysRevLett.77.1413

[63] M. Horodecki, P. Horodecki, R. Horodecki, Separability of mixed quantum states: Linear contractions and permutation criteria, Open Systems & Information Dynamics 13 (1) (2006) 103–111. doi:10.1007/s11080-006-7271-8.
URL http://dx.doi.org/10.1007/s11080-006-7271-8

[64] S. Boyd, L. Vandenberghe, Convex Optimization, Cambridge University Press, New York, NY, USA, 2004.

[65] C. Griffin, Linear programming: Penn state math 484 lecture notes (2009).
URL http://www.personal.psu.edu/cxg286/Math484.html