Quantum Computation with Gottesman-Kitaev-Preskill Codes: Logical Gates, Measurements, and Analysis Techniques

Mackenzie Shaw

This thesis is presented as required for the conferral of the degree:

Master of Philosophy (Science)

Supervisors:
Prof. Andrew Doherty & Dr. Arne Grimsmo

The University of Sydney
Faculty of Science
School of Physics
Quantum Information Theory Group

2022

Abstract

The Gottesman-Kitaev-Preskill (GKP) error-correcting code uses one or more bosonic modes to encode a finite-dimensional logical space, allowing a low-error logical qubit to be encoded in a small number of resonators. In this thesis, I propose new methods to implement logical gates and measurements with GKP codes and analyse their performance. The logical gate scheme uses the single-qubit Clifford frame to greatly reduce the number of gates needed to implement an algorithm without increasing the hardware requirements. The logical measurement scheme uses one ancilla mode to achieve a 0.1% logical error rate over a measurement time of 630 ns when the measurement efficiency is as low as 75%. Finally, I provide a subsystem decomposition which can be used to analyse GKP codes efficiently even as the Fock space distribution of the codestates goes to infinity.

Acknowledgements

I would like to thank both my supervisors Prof Andrew Doherty and Dr Arne Grimsmo for your help throughout the last 18 months – guiding the research project, teaching me new concepts and ideas, and reading and editing countless drafts of this thesis and the papers within it. I'd like to thank the broader Quantum Theory Group for providing a welcome environment to learn more about quantum physics. And finally, thanks to my friends Evan and Faz for our sometimes ridiculous but always thought-provoking conversations, and to my partner Teresa for your ongoing love and support.

I acknowledge support from the Australian Research Council via the Centre of Excellence in Engineered Quantum Systems (EQUS) project number CE170100009. I am also supported by an Australian Government Research Training Program (RTP) Scholarship.

Statement of Contribution of Student

The bulk of this thesis contains nearly-finished drafts of two papers. Except as noted below, I have carried out all the original research in these papers and am the lead author.

Chapter 1 contains a review of the literature and an overview of the contents of the thesis.

Chapter 2 contains a brief introduction to the background theory required to understand the remainder of the thesis, drawing mostly on material in Nielsen and Chuang's textbook 'Quantum Computation and Quantum Information' [1] and Wiseman and Milburn's textbook 'Quantum Measurement and Control' [2].

Chapters 3 and 4 contain current drafts of two papers which are soon to be submitted to journals for which I am the lead author. Both papers are co-authored by my supervisors Prof Andrew Doherty and Dr Arne Grimsmo. Both papers were written by me and edited by my supervisors. Each chapter contains an introduction (sections 3.1 and 4.1), a theoretical background (sections 3.2 and 4.2) and a conclusion (sections 3.6 and 4.7) which provide an overview of the specific background required to understand each paper, while the remaining sections represent new material. I undertook the research presented in all sections of the papers; however, the ideas underpinning each section of each paper came from various sources. Arne originally proposed the idea for the phase-tracked single-qubit logical Clifford gates in section 3.3 and the measurement schemes in section 3.5, while Andrew proposed the use of quantum trajectories as a method to analyse the measurement schemes in section 3.5. The analysis of the GKP gates and measurements via "bin sizes" and "Zak patches" in sections 3.4 and 3.5 was my idea. In chapter 4, the initial idea of using qubit subsystem decompositions to inspire a GKP subsystem decomposition was proposed by Andrew, while the ideas in the remainder of the paper were developed primarily by me.

Finally, Chapter 5 contains my concluding statements and outlook for future research.

As lead supervisor for the candidature upon which this thesis is based, I can confirm that the authorship attribution statements above are correct:

Prof. Andrew Doherty October 17, 2022

I certify that this thesis contains work carried out by myself except where otherwise acknowledged:

Mackenzie Shaw October 17, 2022

Contents

Chapter 1

Introduction

Quantum computing is an exciting new information processing technology that has the potential to solve a number of problems exponentially faster than our current best classical algorithms [1], with applications ranging from number theory [4] to chemistry [5, 6]. Moreover, developments in quantum computing could enable the realisation of related technologies such as quantum cryptography [7, 8] and quantum astronomy [9]. In a landmark experiment in 2019, a specially-designed algorithm was run on a quantum computer orders of magnitude faster than possible on a classical computer [10]. However, to run *useful* algorithms that offer quantum speed-ups on a physical device, the error rates of the qubits must be at least eight orders of magnitude lower than is possible on current devices [11, 12].[1]

As such, it is widely believed that Quantum Error Correction (QEC) [13] is required to achieve sufficiently low error rates to run useful algorithms. The most widely-studied class of QEC code are *qubit* codes, in which the logical qubits used to run an algorithm are stored redundantly in a larger number of physical qubits, allowing the error rate of each encoded logical qubit to be lower than the error rate of each physical qubit. Recent experiments have been able to realise examples of qubit codes such as the repetition [14] and surface codes [11, 15, 16, 17], but are yet to reach the break-even point of the code – the point at which the logical qubits have longer lifetimes than physical qubits – due to the high levels of noise in the experiment. The goal of these experiments is to reach, and then surpass, the *threshold* of the code – the physical error rate below which one can achieve any target logical error rate given enough physical qubits. Excitingly, a recent experiment [15] showed a slight decrease in the logical error rate when the size of the surface code was increased. However, achieving physical error rates which are one or two orders of magnitude below the threshold of the code can drastically reduce the overhead number of physical qubits required to perform a useful algorithm [18, 19, 20], simplifying the requirements needed to build a powerful quantum computer.

One way to reduce the physical error rate is to encode each "physical" qubit additionally in a *bosonic* QEC code [21, 22, 23, 24, 25], forming a code concatenation

[1] This is estimate is based on a recent superconducting surface-code experiment with a per-cycle error rate $\sim 10^{-2}$, and a target logical error rate to factor a 100-bit number using Shor's algorithm of $\sim 10^{-10}$.

between the bosonic and qubit QEC code. Bosonic codes differ from qubit codes in that the physical state space consists of the formally infinite-dimensional Hilbert space of a quantum harmonic oscillator. As such, bosonic codes can be implemented with a single physical unit such as an electromagnetic resonator, a travelling photon, or an ion trap. Excitingly, the first experiment to surpass the break-even was performed using a bosonic code called a cat code [22, 26]. However, this thesis will focus instead on the Gottesman-Kitaev-Preskill (GKP) code [23], whose codestates have been prepared in superconducting microwave resonators [27, 28] and the motional modes of trapped ions [29, 30] but are yet to achieve break-even.

GKP codes are exciting for two reasons in particular. First, GKP codes have been shown to outperform other bosonic codes against a pure loss noise channel, the dominant noise source in most bosonic systems [31]. Second, GKP state preparation is the only non-Gaussian resource required to achieve universal quantum computation [32], in contrast to cat codes which require non-Gaussian resources to perform Clifford gates. Typically, non-Gaussian resources are harder to implement experimentally than Gaussian resources, so minimising the use of non-Gaussian resources reduces the difficulty of building a quantum computer. We do note that while loss is typically the dominant noise source in bosonic systems, secondary error sources such as dephasing and measurement inefficiencies have the potential to affect GKP codes more significantly than other bosonic codes [33].

Currently, there is still work to be done to design practical protocols to implement universal quantum computing with GKP codes that can guide the next generation of experiments. In particular, the theoretical work done on practical computation with GKP codes has largely focussed on Pauli-eigenstate preparation [34, 30, 35], magic state preparation and injection routines [32, 36], and error-correction schemes [23, 33]. However, work on performing logical gates and measurements in practice has only been conducted in the context of optical photons [37] (to the best of my knowledge). Furthermore, simulations of GKP codes have largely been limited to analysing the random Gaussian displacement noise model, which has been shown to be equivalent to a pure loss noise channel followed by a quantum-limited amplification channel [38], but is unable to exactly capture the behaviour of either the pure loss or dephasing noise channels. Since GKP codes are suspected to be more susceptible to dephasing and measurement inefficiencies, methods for simulating these sources of noise are of particular interest in assessing the feasibility of fault-tolerant quantum computation with GKP codes.

We address each of these issues in the two papers presented in this thesis. In chapter 3, we present a scheme for performing logical Clifford gates and Pauli mea-

surements in GKP codes implemented in a superconducting architecture. Our scheme reduces the overall number of gates required to be performed without increasing the hardware overhead, thus reducing the possible spread of errors through the system. As a part of this proposal, we suggest following each performed logical gate with a modified error-correction scheme that better accounts for errors that are spread due to the logical gates, reducing the error rates of each gate by up to two orders of magnitude. Finally, we complement this with an analysis of logical Pauli measurements in the presence of measurement inefficiencies, and propose a scheme to reduce the effect of the inefficient measurements on the logical read-out error rate. This forms the more practical work contained within this thesis, and we hope it can provide guidance for implementing GKP logical gates and measurements in near-term superconducting experiments.

However, to analyse the schemes introduced in chapter 3, we need an efficient method of simulating GKP codes. Traditional Fock space simulations – which involve truncating the infinite-dimensional Hilbert space of the resonator in the Fock/number basis – are not an ideal choice since the mean and variance of the Fock space distribution of GKP codestates increases rapidly as their quality improves [31]. We combat this issue by introducing a new method of simulation of GKP codes which often becomes computationally *less* intensive as the quality of the codestates improves. Our solution, presented in chapter 4, involves defining a logical *subsystem* decomposition of the Hilbert space $\mathcal{H} = \mathcal{L} \otimes \mathcal{S}$ that we call the *stabiliser subsystem decomposition*. We define the stabiliser subsystem decomposition such that performing the partial trace over the non-logical stabiliser subsystem $\mathcal{S}$ corresponds to an ideal GKP decoding operation. This decomposition is also of intrinsic theoretical interest as it provides a subsystem interpretation of GKP codes that correctly characterises the logical information stored in GKP, building on a previously proposed modular-position subsystem decomposition [3, 37, 39, 40, 41]. Thus, chapter 4 provides a more theoretical approach to GKP codes, and reveals fundamental insights and simulation techniques that may guide future theoretical proposals.

Finally, we remark on the contribution of our work towards the overall aim of building a code concatenation of a GKP code with a qubit code, which has been the subject of much theoretical analysis [42, 43, 44, 45, 46]. The overall performance of the concatenated code (and thus the overhead required for a given algorithm) is determined in part by the error rate of the logical gates and measurements proposed in chapter 3. Our schemes to reduce the error rates of each of these components thus bring us closer to building a fault-tolerant quantum computer with GKP codes. In contrast, chapter 4 is best understood in the context of *multi-*

mode GKP codes [23, 47, 48], which are a generalisation of single-mode GKP codes that encompasses concatenated codes as well as other novel code designs. The stabiliser subsystem decomposition can be defined for multi-mode GKP codes, and thus provides a novel simulation method for concatenated GKP codes and other multi-mode GKP codes.

The remainder of this thesis is presented in the following order. We begin in chapter 2 by presenting the technical background information required to understand the publications which make up the body of the thesis. Then, in chapter 3 we present and analyse the schemes for logical gates and measurements for superconducting GKP circuits, with the target audience being a broad audience of both theorists and experimentalists who may wish to utilise our schemes in their experiments. Following this, we provide a more theoretical presentation of the stabiliser subsystem decomposition and how it can be used to simulate GKP codes (chapter 4), with a focus on the mathematical properties of our decomposition and the theoretical insights it provides us. Finally, we provide concluding remarks in chapter 5.

Chapter 2

Background

In this chapter, we provide the background required to understand the main material presented in the thesis in chapters 3 and 4. We begin by introducing the core concepts of quantum information theory in section 2.1, which presents the mathematical formalism of quantum mechanics acting on finite-dimensional, discrete variable systems. This allows us to discuss precisely what is required to construct a *universal* quantum computer in section 2.2. Then, we can turn our attention to continuous variable systems in section 2.3, which describe the quantum mechanical behaviour of infinite-dimensional systems such as quantum harmonic oscillators. This section also provides us with an opportunity to introduce GKP codes at a basic level, leaving the details to the background sections of chapters 3 and 4. In sections 2.4 and 2.5, we introduce the theories of open quantum systems, which is used to describe errors acting on a system, and quantum measurement theory, which is used to describe measurements in the presence of errors. Finally, in section 2.6, we provide a brief discussion of superconducting circuits, which is the physical architecture that we focus on in this thesis, although we do not focus on it here as the details of the mathematical theory are not used in the remainder of the thesis.

2.1 Discrete Variable Systems

Discrete variable systems are the workhorses of quantum computers, and as such we must begin by describing the mathematical formalism of quantum mechanics in the context of quantum information theory (see Ref. [1] for a more detailed introduction). The basic unit of quantum computing is the qubit, described by a complex two-dimensional Hilbert space $\mathcal{H}_{\text{qubit}} \cong \mathbb{C}^2$ spanned by the orthonormal computational states $\{|0\rangle, |1\rangle\}$. States of the qubit are described by *rays* in the Hilbert space – ket vectors $|\psi\rangle = \alpha |0\rangle + \beta |1\rangle \in \mathcal{H}_{\text{qubit}}$ where two kets $|\psi\rangle$ and $|\phi\rangle$ represent the same state if $|\psi\rangle = z |\phi\rangle$ for some $z \in \mathbb{C}$. By convention, we enforce that all kets have unit norm. Operators acting on $\mathcal{H}_{\text{qubit}}$ themselves form a vector

space $\mathcal{L}(\mathcal{H}_{\text{qubit}})$ spanned by the Pauli basis:

$$I = \sigma_0 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \quad Y = \sigma_2 = \begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix}$$
$$X = \sigma_1 = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \quad Z = \sigma_3 = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} \tag{2.1}$$

(where we do not use hats on operators for simplicity). We denote the ± 1-eigenstates of X and Y as $|\pm\rangle$ and $|\pm_Y\rangle$ respectively. Physical operations such as logical gates are described by unitary operators ($U^\dagger U = I$) acting on states, which preserve the norm of the state; for example, time-evolution is described by the unitary operator $U(t) = e^{-iHt}$ (where we have set $\hbar = 1$), where H is the Hamiltonian (energy) operator of the system[1].

We will describe measurements more generally in section 3.5, but here we briefly introduce the concept of a *projective* measurement described by a Hermitian operator O. When a state $|\psi\rangle$ is measured, it is projected into the λ-eigenspace of the operator O (for some eigenvalue λ of O) with probability $\langle\psi|P_\lambda|\psi\rangle$ (where P_λ is the corresponding projection operator), and a measurement outcome λ is returned.

The quantum state of two (or more) qubits is described by a composite Hilbert space which is the tensor product of two (or more) single-qubit Hilbert spaces $\mathcal{H}_{\text{qubit}} \otimes \mathcal{H}_{\text{qubit}}$. The ($n$-qubit) Pauli group is defined by any tensor product of Pauli matrices eq. (2.1) multiplied by any power of the scalar i. Finally, we will also make use of qudits in chapter 4, which are a generalisation of 2-dimensional qubits to a d-dimensional complex Hilbert space spanned by $\{|\mu\rangle\}_{\mu=0,\ldots,d-1}$.

2.2 Universal Quantum Computing

With the mathematical formalism described above, we can now describe how to achieve *universal* quantum computation – i.e. the operations that a quantum computer must be able to perform which allow it to execute any given quantum algorithm. Although there are many ways of achieving universal quantum computing such as measurement-based [49], fusion-based [50] and circuit-based [1] quantum computing, we will be focussing on the Clifford gate + magic state model (which fits within the paradigm of the circuit model) in both chapters 3 and 4. To understand this model, we first introduce the notion of quantum circuits, which describe the operations that are applied to a circuit-based quantum computer in a given algorithm. These are as

[1] Assuming that the Hamiltonian H is time-independent.

follows:

- First, we initialise each qubit in a particular state.
- Next, we perform any number of logical gates on the qubits, where the logical gates can be chosen from some finite set of gates which are implementable on the quantum computer.
- Finally, we perform a measurement of each qubit in a given basis.

The conventional choice of resources required to achieve quantum computation is state preparation in the $|0\rangle$ state, a gate set of Clifford + T, and measurement of the Pauli Z operator. The Clifford group is defined as the group which conjugates Pauli operators to Pauli operators: $APA^\dagger = P'$ for any Clifford operator A and Pauli operators P, P'. The n-qubit Clifford group is generated by the Hadamard (H), phase (S) and controlled-Z (C_Z) gates acting on any qubits:

$$H = \frac{1}{\sqrt{2}}\begin{bmatrix}1 & 1\\ 1 & -1\end{bmatrix} \qquad C_Z = \begin{bmatrix}1 & 0 & 0 & 0\\ 0 & 1 & 0 & 0\\ 0 & 0 & 1 & 0\\ 0 & 0 & 0 & -1\end{bmatrix} \qquad (2.2)$$
$$S = \begin{bmatrix}1 & 0\\ 0 & i\end{bmatrix}$$

If we restricted our gate set to *only* the Clifford group, then any algorithm can be simulated efficiently (in polynomial time) by a classical computer [51]. However, the addition of the $T = \mathrm{diag}(1, e^{i\pi/4})$ gate (or, in general, any non-Clifford gate) results in a universal gate set which can implement universal quantum computing.

Despite the simplicity of this model, T gates are often difficult or impossible to implement in a QEC code without spreading errors [52]. One solution to this problem is to use magic state injection (fig. 2.1), in which a T gate is applied to the target state using an ancilla *magic* state, a Pauli measurement, and Clifford gates (which are performed adaptively depending on the outcome of the measurement). In this way, one can achieve universal quantum computation with a gate set consisting of only the Clifford group, but with state preparation of both the computational $|0\rangle$ state and a magic state such as the +1 Hadamard eigenstate $|H\rangle = \cos(\pi/8)\,|0\rangle + \sin(\pi/8)\,|1\rangle$. Moreover, magic state distillation routines [53] have been designed which take in a large number of noisy magic states and return a smaller number of less noisy magic states, allowing one to ensure that magic states do not jeopardise the fault-tolerance of the quantum computer. However, we note that this procedure does require measurements to be performed in the middle of an algorithm, and gates to be performed adaptively based on the outcomes.

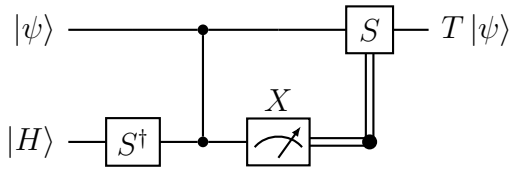

Figure 2.1: A magic state injection circuit in which a T gate is performed without using a physical T gate. This scheme uses one ancilla qubit initialised in the $+1$ Hadamard eigenstate $|H\rangle = \cos(\pi/8)|0\rangle + \sin(\pi/8)|1\rangle$. Two logical gates, $S^\dagger$ and C_Z, are performed, followed by an X measurement of the ancilla qubit. If the measurement outcome of this gate is -1, an S gate is applied to the original qubit, otherwise, nothing is done. After this procedure, the final state on the data qubit is $T|\psi\rangle$.

2.3 Continuous Variable Systems and GKP Codes

Now that we have established the operations required to perform a quantum algorithm, we now turn our attention to continuous variable (CV) systems which describe the physical Hilbert space that encodes bosonic QEC codes. In contrast to discrete variable systems, CV systems such as quantum harmonic oscillators are described by an infinite-dimensional Hilbert space $\mathcal{H}_{\text{CV}}$ spanned by a countably infinite orthonormal Fock basis $\{|n\rangle\}_{n\in\mathbb{N}}$. Operators can be written in terms of ladder operators defined by $a|n\rangle = \sqrt{n}|n-1\rangle$ and $a^\dagger|n\rangle = \sqrt{n+1}|n+1\rangle$ and satisfying $[a, a^\dagger] = 1$, or by position and momentum operators defined by $q = (a + a^\dagger)/\sqrt{2}$ and $p = (a - a^\dagger)/(i\sqrt{2})$ (respectively) and satisfying $[q, p] = i$. Although states can always be described by a superposition of Fock states, we will also frequently make use of the wavefunction of the state, defined by $\psi_q(x) = {}_q\langle x|\psi\rangle$, where $|x\rangle_q$ is the x-eigenstate of the position operator normalised such that $\int_{-\infty}^{\infty} dx\ |x\rangle_{q\,q}\langle x| = I$. Here, the function ψ_q is a member of $L^2(\mathbb{R} \mapsto \mathbb{C})$, the Hilbert space of smooth, square-norm-integrable, complex-valued functions on $\mathbb{R}$.

Now we can provide a brief introduction to GKP codes [23], leaving the details to the background sections of chapters 3 and 4. The simplest example of a GKP code is the single-mode square GKP qubit code, whose ideal codestates are defined in the position basis as:

$$|\bar{0}\rangle = \sum_{s\in\mathbb{Z}} |2s\sqrt{\pi}\rangle_q \qquad |\bar{1}\rangle = \sum_{s\in\mathbb{Z}} |(2s+1)\sqrt{\pi}\rangle_q \tag{2.3}$$

Immediately we can illustrate the types of errors that this code is designed to correct. If an ideal codestate $|\bar{\psi}\rangle = \alpha|\bar{0}\rangle + \beta|\bar{1}\rangle$ is subjected to a small shift in position – described by the operator $e^{-i\delta_x p}$ – we can correct the error as follows.

First, we measure the position of the state modulo $\sqrt{\pi}$ (so as to not reveal the logical information stored in the state). This will give an outcome of $\epsilon_x \equiv \delta_x \pmod{\sqrt{\pi}}$, and thus we can provide a corrective shift in position of $-\epsilon_x$ to return the state back to the ideal codespace. Such a scheme will work whenever $|\delta_x| < \sqrt{\pi}/2$, but larger shifts may induce a logical $\bar{X}$ error which swaps the logical $|\bar{0}\rangle$ and $|\bar{1}\rangle$ codestates.

Mathematically, we can describe the square GKP code using a set of commuting stabiliser generators $S_X = e^{-2i\sqrt{\pi}p}$ and $S_Z = e^{2i\sqrt{\pi}q}$, such that the codespace corresponds to the simultaneous $+1$-eigenspace of both S_X and S_Z. Logical Pauli operators commute with both stabilisers and are given by the operators $\bar{X} = e^{-i\sqrt{\pi}p}$ and $\bar{Z} = e^{i\sqrt{\pi}q}$, with $\bar{Y} = i\bar{X}\bar{Z}$. Moreover, one can take the Fourier transform of the position representation of the codestates in eq. (2.3) to write:

$$|\bar{+}\rangle = \sum_{s\in\mathbb{Z}} |2s\sqrt{\pi}\rangle_p \qquad |\bar{-}\rangle = \sum_{s\in\mathbb{Z}} |(2s+1)\sqrt{\pi}\rangle_p \tag{2.4}$$

in terms of momentum-eigenstates. Repeating the argument described above for momentum eigenstates shows that the square GKP code can correct arbitrary shifts of up to $\sqrt{\pi}/2$ in position *and* momentum. In general, realistic noise sources such as loss and dephasing can always be written as a sum of shifts in position and momentum. GKP codes will be effective against any noise sources that have small support on large shifts in position and momentum.

Unfortunately, the codestates in eqs. (2.3) and (2.4) are not physical as they have infinite norm. As such, we must in practice aim to prepare *approximate* GKP codestates that are normalisable and thus can be created in a physical system. While there are many possible definitions of approximate GKP codestates (many of which are related by simple transformations [54]), in both papers we focus exclusively on codestates defined by the envelope operator $e^{-\Delta^2 a^\dagger a}$ (first appearing in [55]), a non-unitary operator which reduces the norm of the state. In terms of the wavefunction of the state:

$$|\bar{\mu}_\Delta\rangle \propto e^{-\Delta^2 a^\dagger a}|\bar{\mu}\rangle \propto \sum_{s\in\mathbb{Z}} e^{-\frac{1}{2}\tanh(\Delta^2)[(2s+\mu)\sqrt{\pi}]^2} \int_{\mathbb{R}} dx\; e^{-\frac{1}{2}\coth(\Delta^2)[x-\mathrm{sech}(\Delta^2)(2s+\mu)\sqrt{\pi}]^2} |x\rangle_q \tag{2.5}$$

where $\mu = 0, 1$ and the constant of proportionality in $|\bar{\mu}_\Delta\rangle$ is to ensure the state is normalised. The envelope operator can itself be viewed as a source of noise consisting of a superposition of shifts in position and momentum, where the support of the envelope operator on large shifts is suppressed exponentially as the size of the shifts increases. Although the majority of the errors introduced by the envelope operator are correctable, there is a chance that a larger shift is applied to the system, causing

a logical error – an effect that is analysed in detail in both chapters 3 and 4.

2.4 Open Quantum Systems

Although kets can describe any pure quantum state, we must also consider states which consist of mixtures of quantum states, such as those that arise due to stochastic noise acting on a quantum system. To do this, we describe states as density operators $\rho \in \mathcal{L}(\mathcal{H})$. A general density operator can be written:

$$\rho = \sum_i p_i \, |\psi_i\rangle\langle\psi_i| \tag{2.6}$$

where $\{p_i\}$ forms a probability distribution and the states $|\psi_i\rangle$ form an orthonormal basis of $\mathcal{H}$. Equivalently, the set of density operators is equal to the subset of operators in $\mathcal{L}(\mathcal{H})$ that are Hermitian, positive, and have unit trace. The inner product of two density matrices ρ and τ is given by the Hilbert-Schmidt inner product $\mathrm{tr}(\rho\tau^\dagger)$, and the expectation value of an operator O is given by $\mathrm{tr}(O\rho)$. Density matrices often arise from considering entanglement between two systems $\mathcal{H}_A \otimes \mathcal{H}_B$. If one does not have access to the second Hilbert space $\mathcal{H}_B$, then an effective description of the state in $\mathcal{H}_A$ is given by the partial trace of the overall density operator: $\rho_A = \mathrm{tr}_B(\rho_{AB})$.

Physical operations on density operators are described by quantum channels $\mathcal{E}$, which must be completely positive and trace preserving (CPTP) in order to map density operators to other density operators. Unitary operators act on density matrices via the equation $U\rho U^\dagger = \mathcal{J}[U]\rho$, where we introduce the notation $\mathcal{J}$ for simplicity. However, general CPTP quantum channels need not be equivalent to unitary operators, and instead can be written in three equivalent ways:

- as a Kraus decomposition $\mathcal{E}(\rho) = \sum_i E_i \rho E_i^\dagger$, where $\sum_i E_i^\dagger E_i = I$,[2]
- as the partial trace of a unitary operator acting between the system and its environment: $\mathcal{E}(\rho) = \mathrm{tr}_{\mathrm{env}}\big(U(\rho \otimes |0\rangle_{\mathrm{env}}\langle 0|)U^\dagger\big)$, or
- as a superoperator $E = \sum_i E_i \otimes E_i^*$ acting on vectorised density operators $|\rho\rangle = \sum_{m,n} \rho_{mn} \, |m\rangle|n\rangle \in \mathcal{H} \otimes \mathcal{H}^*$, where $\rho_{mn} = \langle m|\rho|n\rangle$ and * denotes complex conjugation.

The time evolution of (time-independent) open quantum systems can always be

[2]Note that the Kraus decomposition is not unique.

written as a master equation in Lindbladian form:

$$\frac{d\rho}{dt} = -i[H,\rho] + \sum_i \mathcal{D}[c_i]\rho, \qquad \mathcal{D}[c_i]\rho = c_i\rho c_i^\dagger - \frac{1}{2}\{c_i^\dagger c_i, \rho\} \tag{2.7}$$

where H is the Hamiltonian of the system, $\{c_i\}$ are the Lindblad operators, and $\{A, B\} = AB + BA$ is the anti-commutator between A and B. In general, one can derive the quantum channel corresponding to evolution under a master equation for time t by performing a series of time-ordered integrals – see Ref. [2] for more details.

The most common examples of quantum channels considered in this thesis are pure loss, defined by time evolution under a Lindbladian $d\rho/dt = \kappa\mathcal{D}[a]\rho$, and white-noise dephasing given by $d\rho/dt = \kappa_\phi\mathcal{D}[a^\dagger a]\rho$. Both of these channels are dealt with in more detail in chapter 4.

2.5 Quantum Measurement Theory

Next, we turn our attention to a more detailed description of quantum measurement theory, which is used extensively in chapter 3 to describe logical measurement schemes on GKP codes (see [2] for a more detailed introduction). In section 2.1 we introduced the notion of a projective measurement, in which a state is projected into the λ-eigenspace of some Hermitian operator O via the projection operator P_λ, with probability $\langle\psi|P_\lambda|\psi\rangle$ and a measurement outcome of λ is recorded. However, many practical measurement schemes cannot be described by a projective measurement of a Hermitian operator, and instead must be described as follows.

First, we specify a set of *effect operators* M_λ for each possible measurement outcome λ. These operators determine the state of the system after a measurement via the equation:

$$\rho_\lambda = \frac{M_\lambda \rho M_\lambda^\dagger}{\mathrm{tr}(M_\lambda \rho M_\lambda^\dagger)} \tag{2.8}$$

where ρ is the pre-measurement state of the system, and ρ_λ is the normalised post-measurement state conditioned on the outcome λ. The probability of observing an outcome λ is thus given by $\mathrm{tr}(E_\lambda\rho)$ where $E_\lambda = M_\lambda^\dagger M_\lambda$. We call the set of operators $\{E_\lambda\}$ the POVM (positive operator-valued measure) of the measurement. Indeed, if one is interested only in the measurement statistics and not the post-measurement state of the system, one only needs to consider the POVM and not the effect operators themselves. In order for the POVM to define a valid probability distribution for every input state, we require that $\sum_\lambda E_\lambda = \sum_\lambda M_\lambda^\dagger M_\lambda = I$. We can also specify the *unconditional* post-measurement state of the system, which is averaged over the

measurement outcomes (i.e. if we "forget" what measurement outcome was observed):

$$\rho_f = \sum_\lambda \mathrm{tr}(E_\lambda \rho)\rho_\lambda = \sum_\lambda M_\lambda \rho M_\lambda^\dagger \tag{2.9}$$

This equation can be viewed as a Kraus decomposition of a quantum channel acting on ρ, where each Kraus operator corresponds to an effect operator of the measurement.

One simple example of a measurement that can be described by this is a photon number measurement of a cavity, which is defined by the effect operators $M_n = |0\rangle\langle n|$. Note that the final state of the system after any measurement is the vacuum state $|0\rangle$, since any photons that have been detected have leaked out of the cavity and are destroyed by the detector, and as such, this measurement cannot be described by a projective measurement of any Hermitian operator.

We will also frequently refer to homodyne detection, which is a measurement of a quadrature operator $q' = e^{i\theta a^\dagger a} q e^{-i\theta a^\dagger a} = q\cos\theta + p\sin\theta$, which has POVM elements $E_x = |x\rangle_{q'q'}\langle x|$ for $x \in \mathbb{R}$. Finally, we will consider *inefficient* homodyne detection, with POVM elements:

$$E_x = \sqrt{\frac{\eta}{\pi(1-\eta)}} \int_{\mathbb{R}} dx' \, \exp\left(-\frac{\eta}{1-\eta}(x-x')^2\right) |x'\rangle_{q'q'}\langle x'| \tag{2.10}$$

One can understand this as applying some Gaussian uncertainty to an ideal homodyne measurement outcome, with $\eta = 1$ representing an ideal measurement and $\eta \in (0, 1)$ representing a non-ideal measurement.

2.6 Superconducting Circuits

Finally, we give a brief background to superconducting circuits, which is the main experimental platform we consider in chapter 3. Superconducting quantum systems are made up of mesoscopic electrical circuits (often with length scales on the order of μm to mm) cooled to their ground state in order to observe the quantum mechanical behaviour of the circuit. In this regime, the physics of the system is described by circuit quantum electrodynamics (circuit QED) [56, 57, 58], of which we provide a brief overview below.

In order to derive the quantum Hamiltonian of a superconducting system using circuit QED, one must start with a lumped element representation of the circuit. The lumped element representation approximates the geometry of the circuit as being comprised of distinct elements, such as capacitors and inductors, connected by superconducting wires. Once this is done, the quantum Hamiltonian of the system

can be derived. Key to this description is the identification of conjugate variables ϕ and n that represent the flux and charge of a node respectively. These variables obey the canonical commutation relation $[\phi, n] = i$, and can also be transformed into ladder operators $[a, a^\dagger] = 1$.

A simple example of a superconducting quantum circuit is the LC oscillator, which consists of an in-series capacitor and inductor. The Hamiltonian of this system is identical to that of a quantum harmonic oscillator:

$$H = \hbar\omega a^\dagger a, \qquad \omega = (LC)^{-1/2} \tag{2.11}$$

where L and C are the inductance and capacitance of the inductor and capacitor respectively. GKP codestates can be encoded in the *rotating frame* of a quantum harmonic oscillator by defining the time-dependent variables $q(t) = e^{-i\omega t a^\dagger a}\phi e^{i\omega t a^\dagger a}$ and $p(t) = e^{-i\omega t a^\dagger a} n e^{i\omega t a^\dagger a}$. The Hamiltonian of the system in this rotating frame is $H = 0$, meaning that states described in terms of the rotating quadratures $q(t), p(t)$ do not evolve in time. This allows GKP codestates to be prepared and stabilised by simply performing all operations in the rotating frame of the LC oscillator.

The key piece of technology that has enabled the realisation of superconducting quantum devices is the Josephson junction [59], which consists of two superconducting regions separated by a thin insulating gap such that Cooper pairs can tunnel from one side of the gap to the other. This element in effect behaves as a non-linear inductor, and when combined in parallel with a capacitor gives rise to a system described by the Hamiltonian:

$$H = 4E_C n^2 - E_J \cos(\phi) \tag{2.12}$$

where $E_C = e^2/2C$, C is the total capacitance of the circuit, and E_J is a parameter which represents the behaviour of the Josephson junction. This Hamiltonian, in the regime $E_J \gg E_C$, describes transmon qubits that form the basis of many of the current leading experimental quantum computing efforts [10, 11]. In particular, the non-linearity of the transmon allows one to construct a qubit out of the lowest two energy levels of the system without significant leakage into the higher energy levels. However, for the purposes of GKP codes, the non-linearity of the transmon is instead used to engineer the Hamiltonians required to perform logical gates (as described in chapter 3). In particular, the leading non-linear term of the transmon Hamiltonian eq. (2.12) is a quartic term in $a, a^\dagger$ that can be used to perform *four-wave mixing* (described in chapter 3). In contrast, one can use a SNAIL [60], which consists of four Josephson junctions in a loop, to create an element with an effective

contribution to the Hamiltonian proportional to $\sin(\phi)$. As the leading order non-linearity is third order, this allows *three-wave mixing*, which we will find has a number of benefits over its four-wave mixing counterpart.

Chapter 3

Practical Logical Clifford Gates and Pauli Measurements in Superconducting Gottesman-Kitaev-Preskill Qubits

Abstract:

The Gottesman-Kitaev-Preskill code is one of the most exciting routes to fault-tolerant quantum computing since Gaussian resources and GKP Pauli-eigenstate preparation are sufficient to achieve universal quantum computing. Encouraged by the preparation of GKP codewords in recent experiments, in this work we provide a practical proposal for logical Clifford gates and state read-out in GKP codes implemented in superconducting circuits. We present a method of performing logical Clifford circuits without physically implementing any single-qubit gates, reducing the potential for them to spread errors in the system. In superconducting circuits, all the required two-qubit gates can be implemented with a single piece of hardware. We analyse the error-spreading properties of logical Clifford gates, and describe how a modification in the decoder following the implementation of each gate can (in the ideal case, exactly) counteract the errors spread by the gates. Finally, we consider the problem of homodyne measurement inefficiencies in the context of logical state read-out, and present a scheme that can implement a measurement with a 0.1% logical error rate in 630 ns with a measurement efficiency of just 75%. We hope that these proposals will guide the next generation of GKP experiments in superconducting devices.

3.1 Introduction

In order to achieve the ambitious goal of constructing a large-scale fault-tolerant quantum computer, quantum error correction (QEC) is required to achieve the low error rates needed to run useful algorithms. Bosonic QEC codes [21, 22, 23, 24] are a promising approach to QEC because they encode logical information in the formally

infinite dimensional Hilbert space of a quantum harmonic oscillator, allowing for robust logical qubits to be constructed from a single physical device. Moreover, one could then use these bosonic codes as the physical "qubits" of a traditional QEC code such as the surface code [44, 45, 46], using the enhanced error tolerance of the bosonic code to reduce the overhead required to reach a given overall logical error rate.

Actively pursued examples of bosonic codes include the cat code [22], binomial code [24] and Gottesman-Kitaev-Preskill (GKP) code [23], with the first experiment to reach the break-even point of a QEC code utilising cat codes [26]. However, GKP codes are particularly interesting since universal quantum computation can be achieved using only Gaussian resources combined with a supply of either GKP Pauli-eigenstates [32] or GKP Hadamard-eigenstates [36]. Such GKP Pauli-eigenstates have been produced in both superconducting microwave cavities [27, 28] and the motional states of trapped ions [29, 30], but are still below the quality required to surpass the threshold of the GKP-surface code [28, 46]. Theoretical work into practical logical gates [37, 61, 39] and error-correction schemes [33] has also recently come into focus. However, to provide a realistic experimental proposal for a given platform, more work is required to develop platform-specific schemes that are fault-tolerant to errors, convenient to implement experimentally, and avoid the leading sources of error in a given platform.

In this work, we present a proposal to perform logical Clifford gates and state read-out in circuit QED that is hardware-efficient and reduces the impact of logical errors and measurement inefficiency. We propose a scheme that removes the need to physically perform single-qubit Clifford gates, reducing the number of physical gates (and hence the spread of errors) in a given algorithm. Furthermore, homodyne detection in microwave circuits is severely limited in practice, with state-of-the-art experiments achieving efficiencies below 90% [62, 63]. As such, we also propose a scheme to improve the effective efficiency of logical read-out by coupling each high-Q GKP mode to a low-Q read-out ancilla. We believe that these schemes provide a promising route to performing logical gates and state read-out in near-term GKP experiments in circuit QED.

We make use of a number of recently-developed techniques to analyse the performance of our scheme. We analyse the spread of logical errors by logical Clifford gates using the stabiliser subsystem decomposition (chapter 4), which is an improvement of the modular-position subsystem decomposition that has been used in previous analyses [3, 37, 39]. In particular, it is designed such that the partial trace over the stabiliser subsystem corresponds exactly to an ideal decoding map, thus allowing us

to use the stabiliser subsystem decomposition to calculate gate fidelities and other quantities of interest. Using this, we show that the spreading of errors due to the application of any logical Clifford gate can be exactly counteracted by a modification of the decoder following the gate, under the assumptions of ideal gate execution and error correction. To analyse our measurement scheme, we use the theory of quantum trajectories, and follow closely the methods of Warszawski *et al.* [64] to obtain an exact POVM representation of the measurement scheme. This allows us to present realistic experimental parameters that can be used to execute high fidelity fast logical read-out of GKP states.

This work focuses on only two of the steps – logical Clifford gates and state read-out – required to perform a fault-tolerant quantum algorithm using GKP codes. Therefore our scheme needs to be combined with other work done on Pauli-eigenstate preparation [34, 35, 27, 28, 30], error correction schemes [23, 33], and implementing non-Clifford gates either directly [39, 48] or via magic state preparation [32]. Moreover, in the context of a fault-tolerant algorithm one needs either to concatenate a GKP code with a qubit code [42, 43, 44, 45, 46] or use a "genuine" multi-mode GKP code [23, 47, 48].

The remainder of this paper is organised as follows. In section 3.2, we provide an overview of GKP codes and the notation we will use throughout the manuscript. In section 3.3, we describe how to remove single-qubit gates from a quantum circuit, and introduce the *generalised controlled gates* which must be performed instead, providing circuits that can implement these gates in a circuit QED experiment. We move on to quantifying the quality of logical gates in section 3.4, in which we explain how to minimise the spread of logical errors using a modified decoding scheme. Finally, we analyse the effect of measurement inefficiencies on logical state read-out in section 3.5, and present our proposal to use an additional ancilla mode to perform homodyne detection with an enhanced efficiency. We provide concluding remarks in section 3.6.

3.2 GKP Codes

We now present a brief overview of the properties of GKP codes [23] and the notation that we will use in the remainder of this manuscript. GKP codes are a class of bosonic stabiliser codes in which the codespace is the simultaneous +1-eigenspace of operators acting on a continuous variable (CV) Hilbert space. The CV system can be described by ladder operators $[a, a^\dagger] = 1$ or quadrature operators $[q, p] = i$. We denote the number states as $|n\rangle$, and position/momentum eigenstates as $|x\rangle_q$

and $|y\rangle_p$. We also introduce the translation operator $T(\mathbf{v}) \equiv e^{-iv_1p+iv_2q}$ for $\mathbf{v} \in \mathbb{R}^2$, which obeys $T(\mathbf{v})T(\mathbf{w}) = e^{-i(v_1w_2-v_2w_1)}T(\mathbf{w})T(\mathbf{v})$. This definition of the translation operator also ensures that $T(\lambda\hat{\mathbf{i}})\,|x\rangle_q = |x+\lambda\rangle_q$ and $T(\lambda\hat{\mathbf{j}})\,|y\rangle_p = |y+\lambda\rangle_p$ (where $\hat{\mathbf{i}}, \hat{\mathbf{j}}$ form the standard orthonormal basis of $\mathbb{R}^2$).

To define a (single-mode) GKP code, we begin with two vectors $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ that satisfy $\alpha_1\beta_2 - \beta_1\alpha_2 = \pi$. The stabiliser generators are then given by $S_X = T(2\boldsymbol{\alpha})$ and $S_Z = T(2\boldsymbol{\beta})$, which (together with their inverses) generate the stabiliser group. The logical Pauli operators are given by $\bar{X} = T(\boldsymbol{\alpha})$, $\bar{Y} = T(\boldsymbol{\alpha}+\boldsymbol{\beta})$ and $\bar{Z} = T(\boldsymbol{\beta})$, where we use bars to indicate logical operators and states. We define the GKP lattice $\mathcal{L}_{\boldsymbol{\alpha},\boldsymbol{\beta}} = \{m\boldsymbol{\alpha} + n\boldsymbol{\beta} \mid m, n \in \mathbb{Z}\}$, and the corresponding Voronoi cell $\mathcal{V}_{\boldsymbol{\alpha},\boldsymbol{\beta}} = \big\{\mathbf{v} \in \mathbb{R}^2 \;\big|\; |\mathbf{v}| < |\mathbf{v}-\mathbf{l}|\ \forall \mathbf{l} \in \mathcal{L}_{\boldsymbol{\alpha},\boldsymbol{\beta}}, \mathbf{l} \neq \mathbf{0}\big\}$ which contains the set of points closer to the origin than any other point in $\mathcal{L}_{\boldsymbol{\alpha},\boldsymbol{\beta}}$.

The simplest example of a GKP code is the square GKP code, given by $\boldsymbol{\alpha}_{\text{sq}} = \sqrt{\pi}\hat{\mathbf{i}}$ and $\boldsymbol{\beta}_{\text{sq}} = \sqrt{\pi}\hat{\mathbf{j}}$. In this case, $\bar{X}_{\text{sq}} = e^{-i\sqrt{\pi}p}$ and $\bar{Z}_{\text{sq}} = e^{i\sqrt{\pi}q}$. General GKP codes can be conveniently described by introducing the canonically transformed quadrature operators $Q = \frac{1}{\sqrt{\pi}}(\beta_2 q - \beta_1 p)$, $P = \frac{1}{\sqrt{\pi}}(\alpha_1 p - \alpha_2 q)$, such that $\bar{X} = e^{-i\sqrt{\pi}P}$ and $\bar{Z} = e^{i\sqrt{\pi}Q}$. As such, most of the properties of square GKP codes can be mapped to general GKP codes simply by replacing $q, p \mapsto Q, P$. Of particular interest is the hexagonal GKP code, given by $\boldsymbol{\alpha}_{\text{hex}} = \sqrt{\frac{\sqrt{3}\pi}{2}}\hat{\mathbf{i}} - \sqrt{\frac{\pi}{2\sqrt{3}}}\hat{\mathbf{j}}$ and $\boldsymbol{\beta}_{\text{hex}} = \sqrt{\frac{2\pi}{\sqrt{3}}}\hat{\mathbf{j}}$, which has been shown to have the lowest logical error rate out of all single-mode GKP geometries under a pure loss noise model [31]. Note that we have chosen a rotated definition of the hexagonal code compared to literature, such that $\beta_1 = 0$ and $Q = \beta_2 q/\sqrt{\pi}$.

In order to aid our discussion of logical Pauli operators in sections 3.3 and 3.4, we introduce the notation $s_1 = -P,\ s_2 = Q - P,\ s_3 = Q$. Using this, we can write $\bar{\sigma}_i = e^{i\sqrt{\pi}s_i}$, where $\sigma_{1,2,3} = X, Y, Z$ respectively. Each quadrature s_i can be written in polar coordinates given by $s_i = r_i(q\cos\theta_i + p\sin\theta_i)$.

Logical Clifford operators, which map logical Pauli operators to logical Pauli operators, are given by unitary Gaussian operators $\bar{A}$ acting on the CV space. Concretely, the logical Clifford group is generated by the operators $\bar{H} = e^{i\frac{\pi}{4}(Q^2+P^2)}$, $\bar{S} = e^{iQ^2/2}$ and $\bar{C}_Z = e^{iQ\otimes Q}$. Equivalently, one can describe an arbitrary n-mode Gaussian operator U with a symplectic matrix $\Sigma_U \in \text{Sp}(2n, \mathbb{R})$ acting on the vector of quadrature operators $\boldsymbol{\xi} = [q_1, \ldots, q_n, p_1, \ldots, p_n]^T$, such that $U^\dagger \boldsymbol{\xi} U = \Sigma_U \boldsymbol{\xi}$ (where U and $U^\dagger$ act component-wise on $\boldsymbol{\xi}$). For example, a single-mode rotation operator $R(\theta) = e^{i\theta a^\dagger a}$ can be described as a symplectic operator $\Sigma_{R(\theta)} = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix}$. In

the square GKP code, we have:

$$\Sigma_{\bar{H}_{\mathrm{sq}}} = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix}, \quad \Sigma_{\bar{S}_{\mathrm{sq}}} = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix}, \quad \Sigma_{\bar{C}_{Z,\mathrm{sq}}} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 \end{bmatrix} \tag{3.1}$$

In general GKP codes, these symplectic matrices can be found by conjugating $\Sigma_{\bar{A}}$ with the change of basis matrix $\boldsymbol{\Xi} = M\boldsymbol{\xi}$ (where $\boldsymbol{\Xi} = [Q_1, \ldots, Q_n, P_1, \ldots, P_n]^T$). This symplectic description of Clifford gates will be used in our discussion of error-protected logical gates in section 3.4.

The ideal codestates of the GKP code are given by:

$$|\bar{\mu}\rangle = \sum_{s\in\mathbb{Z}} e^{i(2s+\mu)^2\alpha_1\alpha_2/2} \left|(2s+\mu)\alpha_1\right\rangle_q \tag{3.2}$$

for $\mu = 0, 1$. The above equation holds for general GKP codes, with the only restriction that $\boldsymbol{\alpha}, \boldsymbol{\beta}$ are rotated such that $\beta_1 = 0$.

However, these ideal codestates are non-normalisable and hence cannot be realised in any physical system. To construct normalisable codestates, we apply the non-unitary envelope operator $e^{-\Delta^2 a^\dagger a}$ to each of the ideal codestates. The envelope operator can be written in the position basis as:

$${}_q\langle x'| e^{-\Delta^2 a^\dagger a} |x\rangle_q \propto \exp\left(-\frac{1}{2}\coth\left(\Delta^2\right)\left(x'^2 + x^2\right) + \operatorname{csch}\left(\Delta^2\right)x'x\right) \tag{3.3}$$

with constant of proportionality $\sqrt{\frac{\coth(\Delta^2)+1}{2\pi}}$. From this, we define approximate GKP codestates as normalised states $|\bar{\mu}_\Delta\rangle \propto e^{-\Delta^2 a^\dagger a}|\bar{\mu}\rangle$. This approximation, however, introduces errors into the system as these states are not exact $+1$ eigenstates of the stabilisers. The parameter Δ characterises the quality of the approximate GKP codestates, where the limit $\Delta \to 0$ approaches the ideal codestates. We will also commonly quote the average photon number of the GKP states $\bar{n} = \frac{1}{2}\big(\langle\bar{0}_\Delta|a^\dagger a|\bar{0}_\Delta\rangle + \langle\bar{1}_\Delta|a^\dagger a|\bar{1}_\Delta\rangle\big) \approx \frac{1}{2\Delta^2} - \frac{1}{2}$, and the GKP squeezing parameter $\Delta_{\mathrm{dB}} = -10\log_{10}(\Delta^2)$, both of which tend to infinity as $\Delta \to 0$.

Such codestates have been prepared experimentally in superconducting resonators with an experimentally-determined squeezing of $\Delta_{\mathrm{dB}} = 9.1$ [28]. GKP-surface code studies have shown that the surface code threshold can be reached using codestates with $\Delta_{\mathrm{dB}} = 9.9$ assuming that the dominant source of noise is due solely to the approximate GKP codestates. However, in the presence of circuit noise, a larger squeezing is required to get under the surface code threshold. As such, we use

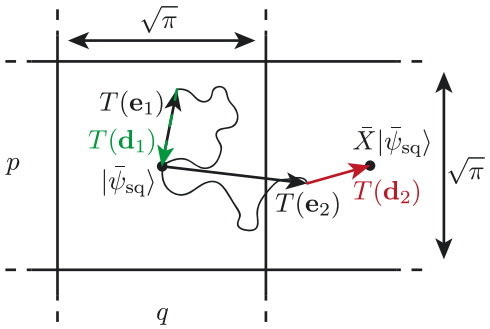

Figure 3.1: The ideal error-correction procedure for the square GKP code, performed over the Voronoi cell of the square GKP lattice. In this case, the Voronoi cell is a square centred at the origin with side-length $\sqrt{\pi}$. In green, a random walk of translation errors results in an overall translation error $T(\mathbf{e}_1)$ acting on the ideal codestate. Since both components of $T(\mathbf{e}_1)$ are less than $\sqrt{\pi}/2$, the shortest translation that returns the state to the codespace is $T(\mathbf{d}_1)$. In red, the translation error $T(\mathbf{e}_2)$ translates the position of $|\bar{0}\rangle$ by more than $\sqrt{\pi}/2$. This means that the smallest translation that returns the state to the codespace is $T(\mathbf{d}_2)$, which has the net result of applying a logical $\bar{X}$ on the state.

$\Delta_{\text{dB}} = 12$ as a rough target squeezing for practical quantum computing with GKP codes. We will also focus on implementations of GKP qubits in the rotating frame of a microwave resonator with Hamiltonian $H = \hbar\omega a^\dagger a$.

Since the codestates $|\bar{\mu}_\Delta\rangle$ are not orthogonal, we define the orthonormalised GKP codestates $|\bar{\mu}_{\Delta,\text{o}}\rangle$, which form an orthonormal basis of the subspace spanned by $\{|\bar{\mu}_\Delta\rangle\}_{\mu=0,1}$. We will use both orthonormalised and non-orthonormalised GKP codestates at different points throughout this manuscript depending on the application we are using them for. Note that the difference between $|\bar{\mu}_\Delta\rangle$ and $|\bar{\mu}_{\Delta,\text{o}}\rangle$ is negligible for any practical values of Δ.

Next, we discuss error correction for the GKP code, beginning with what we refer to as ideal error correction, which consists of the following steps. First, we measure both stabilisers $S_X = e^{-2i(\alpha_1 p - \alpha_2 q)}$ and $S_Z = e^{2i(\beta_2 q - \beta_1 p)}$, with measurement outcomes M_X, M_Z. We only consider the case where these stabilisers can be measured ideally; it is in this sense that we use the word "ideal" in the phrase ideal error-correction. We assign each pair of measurement outcomes with a quasi-position and momentum $\mathbf{k} = [k_q, k_p]^T$ such that $M_X = e^{-2i(\alpha_1 k_p - \alpha_2 k_q)}$ and $M_Z = e^{2i(\beta_2 k_q - \beta_1 k_p)}$ respectively. Finally, a translation $T(-\mathbf{k})$ is applied that returns the state to the ideal codespace of the code.

However, the periodicity of the complex exponential means that the map from measurement outcomes $(M_X, M_Z) \mapsto \mathbf{k} \in \mathbb{R}^2$ is not well-defined; in particular, the vectors $\mathbf{k}$, $\mathbf{k}+\boldsymbol{\alpha}$ and $\mathbf{k}+\boldsymbol{\beta}$ all correspond to the same pair of measurement outcomes. As such, one must specify a correction patch $\mathcal{P}$ that uniquely specifies a vector $\mathbf{k}$ for

each pair of measurement outcomes. In general, the best choice of patch depends on the noise model being applied to the GKP code. If we consider a noise model defined by a random walk of translation errors, we can write the overall error (up to an irrelevant overall phase) as $T(\mathbf{e})$, where we can sample $\mathbf{e}$ from a mean-zero Gaussian distribution. In this case, the best patch to decode over is the Voronoi cell $\mathcal{V}_{\boldsymbol{\alpha},\boldsymbol{\beta}}$ of the GKP lattice, see fig. 3.1. This choice ensures that the translation $T(-\mathbf{k})$ is the shortest translation that returns the state to the codespace, thereby minimising the chance of a logical error. However, we will consider error correction over different patches in our discussion of error-protected logical gates in section 3.4. The above description also remains valid in the case of multi-mode error correction by increasing the number of stabilisers and the dimension of $\mathbf{k}$.

3.3 Phase-tracked single-qubit Clifford gates

In this section we outline how to perform arbitrary Clifford circuits in GKP codes without physically implementing any single-qubit Clifford gates, thus reducing the spread of errors in the computation. To do this, the single-qubit Clifford gates can be tracked in software, and absorbed into the two-qubit Clifford gates in the circuit. We call the resulting two-qubit gates that one must perform *generalised controlled gates*, and the process is sometimes referred to as the "Clifford frame" [65]. Moreover, all the generalised controlled gates can be implemented using a single piece of hardware, with each gate differentiated by the phase of a local oscillator. This is advantageous since it reduces the number of physical gates that must be implemented, thus reducing the spread of errors in the circuit (as discussed in section 3.4).

We now step through precisely how the single-qubit Clifford gates need to be tracked in order to implement a general quantum computation. We start with a universal quantum computing circuit comprising of state preparation in the $|0\rangle$ or $|T\rangle$ states, Pauli Z-measurements, and adaptive Hadamard, phase and controlled-Z gates. We do not consider the preparation of GKP magic states in this manuscript as they have been discussed elsewhere [32]. One can rewrite such a circuit instead consisting of state preparation in the $|0\rangle$ or $|T\rangle$ states, Pauli X, Y and Z-measurements, and generalised controlled gates (which must be performed adaptively). For Pauli matrices σ_i, σ_j $(i, j \in \{1, 2, 3\})$, we define the σ_i-controlled-σ_j gate as:

$$C_{\sigma_i\sigma_j} = I \otimes I - \frac{1}{2}(I - \sigma_i) \otimes (I - \sigma_j) \tag{3.4}$$

These gates can be interpreted as applying a σ_j gate to the target (second) qubit

if the control (first) qubit is a -1 eigenstate of σ_i, and otherwise doing nothing. Note that the Z-controlled-Z gate is what is commonly referred to simply as the controlled-Z gate, and the Z-controlled-X gate is a controlled-NOT gate.

In order to rewrite a Clifford circuit in terms of generalised controlled gates, we use the following fact. Given a generalised controlled gate $C_{\sigma_i\sigma_j}$ and a single-qubit Clifford gate A, we have that:

$$C_{\sigma_i\sigma_j}(A \otimes I) = (A \otimes B)C_{\sigma_{i'}\sigma_j} \tag{3.5}$$

where $\sigma_{i'}$ is given by calculating $A^\dagger \sigma_i A = (-1)^a \sigma_{i'}$ (for $a = 0, 1$), and where $B = I$ if $a = 0$ and $B = \sigma_j$ if $a = 1$. This can be used to commute the Hadamard and phase gates past the controlled-Z gates. Subsequently, the remaining single-qubit Clifford gates can be commuted past the Z-measurements, leaving X, Y and Z Pauli measurements (which are discussed in Sec. 3.5) and removing the single-qubit Clifford gates entirely from the circuit.

To implement a generalised controlled gate $C_{\sigma_i\sigma_j}$ between two GKP modes a and b with a gate time T, we must engineer the Hamiltonian:

$$H_{\sigma_i\sigma_j} = -\frac{\hbar}{T} s_i \otimes s_j = -\frac{\hbar r_i r_j}{2T}(e^{-i(\theta_i+\theta_j)}ab + e^{-i(\theta_i-\theta_j)}ab^\dagger + \text{h.c.}) \tag{3.6}$$

where $s_1 = -P$, $s_2 = Q - P$, $s_3 = Q$ are the quadratures introduced in section 3.2 and r_i, θ_i are their polar coordinates. We can interpret this Hamiltonian as quadrature-quadrature coupling; or, equivalently, as a phase-coherent superposition of beamsplitter and two-mode squeezing interactions. We remark that for the square GKP code, $r_2 = \sqrt{2} > r_1 = r_3 = 1$, and as such, the Hamiltonian strength or the gate time must be increased for generalised controlled gates involving Y; while in the hexagonal GKP code $r_1 = r_2 = r_3$.

To implement the Hamiltonian eq. (3.6) we can utilise either four-wave or three-wave mixing between GKP modes, as depicted in fig. 3.2. These schemes require a transmon or a SNAIL [60] (respectively) capacitively coupled to each of the microwave resonators housing the GKP modes. Moreover, two drive tones are required in both schemes with frequencies $(\omega_a \pm \omega_b)/2$ or $\omega_a \pm \omega_b$ (respectively), where ω_a and ω_b are the microwave resonant frequencies. Finally, these tones must be applied with a relative phase that determines which generalised controlled gate is implemented.

In the following paragraphs we give an intuitive and non-rigorous description of how to construct the Hamiltonian that results from such a four- or three-wave mixing circuit, and refer the reader to Ref. [66] for a more thorough derivation. Four-wave mixing can be understood as adding to the system Hamiltonian any

non-rotating terms consisting of the product of any four of the operators: $a, a^\dagger, b, b^\dagger$ and drive terms $V_d e^{\pm i(\omega_d t+\phi_d)}$ (where V_d, ω_d and ϕ_d are the strength, frequency and phase of any of the microwave drive tones). Using this heuristic, one can see that Kerr and cross-Kerr terms will be added to the Hamiltonian even in the absence of microwave drives since the terms $(a^\dagger)^2a^2$, $(b^\dagger)^2b^2$ and $a^\dagger ab^\dagger b$ are always non-rotating. However, adding a drive term with frequency $\omega_1 = (\omega_a + \omega_b)/2$ ensures that the term $V_1^2 abe^{2i(\omega_1 t+\phi_1)}$ (and its Hermitian conjugate) are non-rotating, thus providing a two-mode squeezing interaction with relative phase $2\phi_1$. As the GKP codewords are encoded in the rotating frame of the resonator, this term acts as $V_1^2 abe^{2i\phi_1} + \text{h.c.}$ in the rotating frame, which forms half of the interaction required in eq. (3.6). Following similar logic, one can see that the beamsplitter terms in eq. (3.6) can be engineered with a second drive with frequency $\omega_2 = (\omega_a - \omega_b)/2$. The phase difference between each of the applied microwave tones and the oscillator determines the phase on the beamsplitter and two-mode squeezing terms, thereby specifying which generalised controlled gate is implemented. However, the addition of the two drives also adds the AC Stark shift terms such as $V_1^2 a^\dagger a$, which alter the resonant frequency of each cavity depending on the drive strength.

To avoid the unwanted Kerr, cross-Kerr and AC Stark shift terms, we propose using three-wave mixing, which can be implemented in circuit QED by replacing the transmon with a SNAIL. Intuitively, three-wave mixing differs from four-wave mixing by only allowing the addition of non-rotating terms containing products of *three* operators each. This avoids the introduction of Kerr, cross-Kerr and AC stark shift terms, while still allowing simultaneous beamsplitter and two-mode squeezing interactions provided the drive frequencies are $\omega_1 = \omega_a + \omega_b$ and $\omega_2 = \omega_a - \omega_b$. This would allow one to perform generalised controlled gates, and hence all Clifford circuits, between microwave resonators encoding GKP states, while avoiding Kerr, cross-Kerr and AC Stark shift terms.

The net effect of implementing circuits in this way is that we have removed the need to explicitly perform single-qubit Clifford gates, instead accounting for them by altering the phase of a local oscillator in the three- or four-wave mixing circuit. We envisage that the generalised controlled gate that must be implemented (or, equivalently, the phases of the local oscillators in each gate) can be tracked in software parallel to the quantum computation as the measurement results from magic state injection gadgets are obtained.

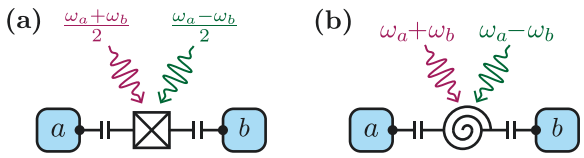

Figure 3.2: Schematic diagrams of (a) four-wave mixing, and (b) three-wave mixing between two GKP modes a and b with resonant frequencies ω_a and ω_b. By phase coherently driving the non-linear coupling element with two microwave drives whose frequencies are given in eq. (3.6), one can engineer the Hamiltonian required to perform each generalised controlled gate by altering the relative phase between the drives and the gate time. Supplemented with state preparation in the $|0\rangle$ or $|T\rangle$ states and measurement in any logical Pauli basis, this is sufficient to achieve universal quantum computing.

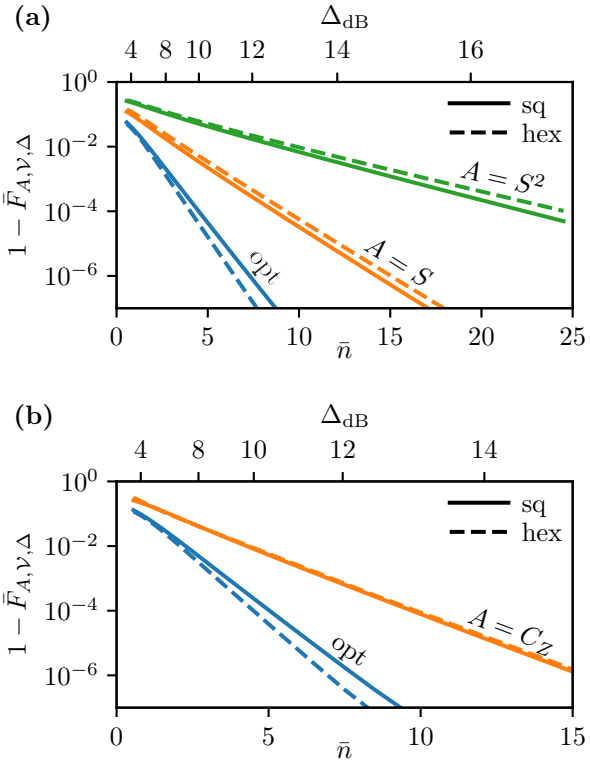

Figure 3.3: Average Gate Infidelity $1-\bar{F}_{A,\mathcal{V}_{\boldsymbol{\alpha},\boldsymbol{\beta}},\Delta}$ of (a) single-qubit, and (b) two-qubit logical Clifford gates $A=I,S,S^2,C_Z$ applied to the square and hexagonal GKP codes, decoded over the Voronoi cell $\mathcal{V}_{\boldsymbol{\alpha},\boldsymbol{\beta}}$ of their respective lattices. The average gate infidelity of the S,S^2 and C_Z gates can be reduced to be the same as that of the identity gate by instead performing ideal error correction over a modified patch $\mathcal{P}=\Sigma_{\bar{A}}\mathcal{V}_{\boldsymbol{\alpha},\boldsymbol{\beta}}$ immediately following the application of the gate, where $\Sigma_{\bar{A}}$ is a symplectic matrix with unitary representation equal to $\bar{A}$. The plots labelled opt represent the average gate fidelity of the identity gate $\bar{F}_{I,\mathcal{V},\Delta}$ and that of Clifford gates with a modified error correction patch $\bar{F}_{\bar{A},\Sigma_{\bar{A}}\mathcal{V},\Delta}$. Note that the square and hexagonal C_Z gates have similar average gate infidelities and are overlapping in (b).

3.4 Error-resistant Clifford gates

We now move our attention to analysing the quality of logical Clifford gates where the only source of noise is from the approximation of the GKP codewords themselves. In particular, we present a modification of the error-correction patch immediately following the application of a Clifford gate that can exactly counteract the spreading of errors due to the gate, thus reducing the average gate infidelity by up to two orders of magnitude. To explain this result, we will begin by introducing the metric we will use to quantify the quality of logical gates. We analyse both single-qubit and two-qubit Clifford gates in this setting, and find that in general, Clifford gates will propagate errors already present in the system unless the error correction patch is modified. Our proposed modification is a generalisation of the error-protected two-qubit gates described in Ref. [46]. It is beyond the scope of this paper to consider additional sources of noise such as photon loss, Kerr non-linearities, mistimed gates, or non-ideal error correction following the gate. However, in the presence of these additional noise sources, we argue that removing the physical implementation of single-qubit gates from the computation will reduce the error rate of the computation, justifying the use of our scheme in section 3.3.

We define the metric as follows. Given a logical gate U (which acts on n qubits) with implementation on GKP codestates given by $\bar{U}$ (which acts on n approximate GKP codestates $|\bar{\psi}_\Delta\rangle$), we define a quantum channel:

$$\mathcal{E}_{U,\mathcal{P},\Delta} = \mathcal{J}[R^\dagger] \circ \mathcal{C}_\mathcal{P} \circ \mathcal{J}[\bar{U} R_\Delta] \tag{3.7}$$

where $\mathcal{J}[O]\rho = O\rho O^\dagger$. This can be understood as follows. First, $R_\Delta = \left(|\bar{0}_{\Delta,\mathrm{o}}\rangle\langle 0| + |\bar{1}_{\Delta,\mathrm{o}}\rangle\langle 1|\right)^{\otimes n}$ is the encoding operator that maps n-qubit density operators to n-mode density operators that encode the information in (orthogonalised) approximate GKP codestates. Once the state is encoded, we apply the logical operator $\bar{U}$ to the state. Before decoding, a round of ideal error correction over a patch $\mathcal{P}$ is performed which returns the state to the ideal codespace, represented by the map $\mathcal{C}_\mathcal{P}$. Finally, the logical information in the state is decoded by the map $R^\dagger = \left(|0\rangle\langle\bar{0}| + |1\rangle\langle\bar{1}|\right)^{\otimes n}$. With $\mathcal{E}_{U,\mathcal{P},\Delta}$, we can calculate the average gate fidelity $\bar{F}_{U,\mathcal{P},\Delta}$ of the logical gate U from the entanglement fidelity of the map $\mathcal{J}[U^\dagger] \circ \mathcal{E}_{U,\mathcal{P},\Delta}$ [67].

In order to compute $\mathcal{E}_{U,\mathcal{P},\Delta}$ we utilise the GKP stabiliser subsystem decomposition that we have recently developed (chapter 4), which has the property that taking the partial trace over a stabiliser subsystem corresponds to the ideal decoding map $\mathcal{J}[R^\dagger] \circ \mathcal{C}_{\mathcal{V}_{\boldsymbol{\alpha},\boldsymbol{\beta}}}$. The mathematical details of the decomposition are presented in chapter 4, so we will instead focus on the main results of our analysis here. One

advantage to using this technique is that numerical calculation of the gate fidelity requires fewer computational resources as $\Delta \to 0$, in contrast to traditional Fock space simulations which require higher truncation dimensions as $\Delta \to 0$. We note that the analysis presented here differs quantitatively from previous work based on the modular-position subsystem decomposition [3, 37]; the relationship between the GKP stabiliser and modular-position subsystem decompositions is explored in chapter 4.

Before presenting our numerical results, we derive in section 3.A an estimate for $1 - \bar{F}_{I,\mathcal{P},\Delta}$. We denote d as twice the length of the shortest vector on the boundary of $\mathcal{P}$ (the "distance" of the patch), and a as the number of vectors with this property divided by two (representing the number of equally likely logical errors). Then, the estimate is given by:

$$1 - \bar{F}_{I,\mathcal{P},\Delta} \approx ca\frac{4\Delta}{d\sqrt{\pi}}e^{-d^2/(4\Delta^2)} \tag{3.8}$$

where $c = 1/3$ for single-qubit gates and $c = 2/5$ for two-qubit gates. Here, the only source of errors are those due to the use of approximate GKP codestates, and the average gate infidelity of the identity gate is not equal to 0. For a given logical gate A, the average gate infidelity will be larger than or equal to that of the identity gate, depending on how $\bar{A}$ propagates the errors due to Δ. We show in chapter 4 that

$$\bar{F}_{A,\mathcal{P},\Delta} = \bar{F}_{I,\Sigma_{\bar{A}}^{-1}\mathcal{P},\Delta} \tag{3.9}$$

for any logical Clifford gate A represented by its symplectic matrix $\Sigma_{\bar{A}}$. If error correction is performed over the Voronoi cell $\mathcal{P} = \mathcal{V}_{\boldsymbol{\alpha},\boldsymbol{\beta}}$, then typically the modified patch $\Sigma_{\bar{A}}^{-1}\mathcal{P}$ has a shorter distance than $\mathcal{P}$. Note, however, that any gates implemented by a rotation – notably the Hadamard gate $\bar{H}_{\mathrm{sq}} = e^{i\frac{\pi}{2}a^\dagger a}$ in the square GKP code, and the permutation gate $\bar{H}_{\mathrm{hex}}\bar{S}_{\mathrm{hex}}^\dagger = e^{i\frac{\pi}{3}a^\dagger a}$ in the hexagonal GKP code – achieve the same average gate fidelity as the identity gate since they leave the Voronoi cells of the codes invariant.

In table 3.1, we present the distance and degeneracy of the Voronoi cells of the square and hexagonal GKP codes, as well as their modification due to the Clifford gates $\bar{S}, \bar{S}^2, \bar{C}_Z$. In fig. 3.3, we present numerically calculated average gate infidelities $1 - F_{U,\mathcal{V}_{\boldsymbol{\alpha},\boldsymbol{\beta}},\Delta}$ of ideal single-qubit and two-qubit Clifford gates for the square and hexagonal GKP code, decoded over a patch corresponding to the Voronoi cell of the square/hexagonal lattice. In particular, the average gate infidelity of the identity gate is roughly two orders of magnitude lower than that of the phase gate for $\Delta_{\mathrm{dB}} = 12$ in both the square and hexagonal codes. In the hexagonal GKP code, each generalised controlled gate $C_{\sigma_i\sigma_j}$ has an average gate infidelity identical to that of C_Z, while in

Code	Square		Hexagonal	
Gate	a	$d/\sqrt{\pi}$	a	$d/\sqrt{\pi}$
$\bar{I}$	2	1	3	$\sqrt{2/\sqrt{3}} \approx 1.07$
$\bar{S}$	1	$1/\sqrt{2} \approx 0.707$	2	$\sqrt{2\sqrt{3}/7} \approx 0.703$
$\bar{S}^2$	1	$1/\sqrt{5} \approx 0.447$	1	$\sqrt{2\sqrt{3}/19} \approx 0.427$
$\bar{I} \otimes \bar{I}$	4	1	6	$\sqrt{2/\sqrt{3}} \approx 1.07$
$\bar{C}_Z$	2	$1/\sqrt{2} \approx 0.707$	2	$\sqrt{2\sqrt{3}/7} \approx 0.703$

Table 3.1: Summary of the distances d and degeneracies a of the patches $\Sigma_{\bar{A}}^{-1}\mathcal{V}_{\boldsymbol{\alpha},\boldsymbol{\beta}}$ corresponding to the logical Clifford gates $\bar{A} = \bar{I}, \bar{S}, \bar{S}^2, \bar{I}\otimes\bar{I}, \bar{C}_Z$ for the square and hexagonal GKP codes decoded over their Voronoi cells $\mathcal{V}_{\boldsymbol{\alpha},\boldsymbol{\beta}}$. The average gate fidelity of each gate is estimated by eq. (3.9), but can be improved to the average gate fidelity of the identity gate by modifying the patch over which error correction is performed to $\mathcal{P} = \Sigma_{\bar{A}}\mathcal{V}_{\boldsymbol{\alpha},\boldsymbol{\beta}}$.

the square GKP code the average gate infidelity is larger in gates with i or $j = 2$.

To counteract the effects of the spreading of errors, one can instead use a modified error correction patch $\mathcal{P}$ that, in the case of ideal error correction, exactly counteracts the spreading of errors due to the logical Clifford gates. Given the symplectic matrix $\Sigma_{\bar{A}}$ that represents the logical Clifford gate $\bar{A}$, the modified error correction patch is given by $\Sigma_{\bar{A}}\mathcal{V}_{\boldsymbol{\alpha},\boldsymbol{\beta}} = \{\Sigma_{\bar{A}}\mathbf{v} \mid \mathbf{v} \in \mathcal{V}_{\boldsymbol{\alpha},\boldsymbol{\beta}}\}$ (see fig. 3.4). From eq. (3.9), one can see that such a modification exactly counteracts the spreading of errors due to the logical gate such that $\bar{F}_{A,\Sigma_{\bar{A}}\mathcal{V},\Delta} = \bar{F}_{I,\mathcal{V},\Delta}$. Our results show that to achieve a 10^{-4} error rate in a C_Z gate (even in the absence of gate noise), one would need a squeezing of $\Delta_{\mathrm{dB}} \approx 10.5$ (for square GKP) with our simplification, compared to a squeezing of $\Delta_{\mathrm{dB}} \approx 13$ without it. In the context of a full computation, we envisage that the round of error correction immediately following the implementation of a logical gate is performed over the corresponding modified patch, while all other rounds are performed over the Voronoi cell.

This idea is a generalisation of error-protected two-qubit gates, which have already been applied in GKP-surface code simulations [46]. Since this eliminates the errors caused by the spreading of errors by the gate, the new leading sources of error will be from the non-ideal implementation of logical gates, and from performing approximate error correction over the modified patch, both of which are left to future research. However, we expect that performing approximate error correction over the modified patch will result in larger errors than over the Voronoi cell of the lattice. Both of

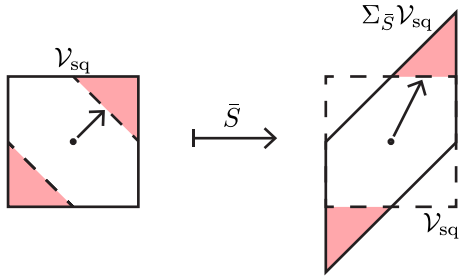

Figure 3.4: Diagram of the transformation of the square GKP Voronoi cell $\mathcal{V}_{\text{sq}} \mapsto \Sigma_{\bar{S}_{\text{sq}}}\mathcal{V}_{\text{sq}}$ under a logical phase gate $\bar{S}_{\text{sq}}$ represented by the symplectic matrix $\Sigma_{\bar{S}_{\text{sq}}}$ (eq. (3.1)). If error correction is subsequently performed over $\mathcal{V}_{\text{sq}}$, then pre-gate translation errors lying in the red shaded regions are mapped to logical errors by $\bar{S}$. The shortest such translation is represented by the arrow. Alternatively, if error correction is performed over the modified patch $\Sigma_{\bar{S}_{\text{sq}}}\mathcal{V}_{\text{sq}}$, the pre-image of this patch is $\mathcal{V}_{\text{sq}}$, maximising the distance of the patch.

these considerations make it advantageous to remove single-qubit Clifford gates from a quantum computation, justifying our scheme in section 3.3.

3.5 Logical Pauli Measurements

In order to implement Clifford circuits using only generalised controlled gates, it is necessary to also perform single-qubit logical measurements of all three Pauli operators X, Y, Z. To perform such a read-out at the end of a computation, the simplest proposal is to use homodyne detection on the GKP mode. However, the efficiency of homodyne detection in the microwave regime is low, with typical values ranging between 0.5 and 0.75. As such we will analyse the effect of such inefficient homodyne detection on the logical readout failure rate of GKP codes.

Moreover, we propose two schemes that can improve the effective efficiency of logical readout, one scheme based on single-mode squeezing, and the second based on quadrature-quadrature coupling to a separate read-out mode (which could be implemented with the circuits shown in fig. 3.2). Both of these schemes can be written as an effective homodyne read-out of the GKP mode with an effective efficiency η_{eff} depending on the parameters used in the scheme. We show how to derive η_{eff} in each of these schemes using the theory of quantum trajectories, and propose experimental parameters that would allow a high-efficiency fast measurement of the GKP mode.

In order to measure a logical Pauli operator $\bar{\sigma}_i = e^{i\sqrt{\pi}s_i}$, we can measure the rotated quadrature $q' = s_i/r_i = q\cos\theta_i + p\sin\theta_i$, round the result to the nearest multiple of $b = \sqrt{\pi}/r_i$ (which we call the *bin size* of the measurement), and interpret the result

as $+1$ if it is an even multiple of b and as -1 if it is an odd multiple. In the remainder of this section, we will (without loss of generality) only consider $\bar{Z}$ measurements of GKP codes rotated such that $\beta_1 = 0$, as all other Pauli measurements are equivalent up to a re-scaling of the bin size and rotation of the measurement quadrature. This convention sets the bin size $b = \sqrt{\pi}/r_3 = \alpha_1$ and measurement quadrature $q' = q$.

Interestingly, applying the above $\bar{Z}$ measurement to approximate GKP codewords $|\bar{\mu}_\Delta\rangle$ is not equivalent to maximum likelihood decoding of the measurement outcome, in which a measurement outcome $q = x$ is interpreted as a logical $+1$ outcome iff $\left|{}_q\langle x|\bar{0}_\Delta\rangle\right|^2 > \left|{}_q\langle x|\bar{1}_\Delta\rangle\right|^2$. Such a maximum likelihood decoder cannot be exactly represented using a constant bin size as described above; however, it is well approximated by setting the bin size to $b = \cosh(\Delta^2)\alpha_1$, where the correction $\cosh(\Delta^2) \to 1$ as $\Delta \to 0$. This correction can be understood qualitatively by noting that the height of the central $q = 0$ peak of ${}_q\langle x|\bar{0}_\Delta\rangle$ is larger than the height of the innermost peaks of ${}_q\langle x|\bar{1}_\Delta\rangle$, which occur at $q = \pm\,\mathrm{sech}(\Delta^2)\alpha_1$. As a result, the two wavefunctions intersect at a value of q greater than $\mathrm{sech}(\Delta^2)\alpha_1/2$. We note however that the differences between using a bin size of $b = \alpha_1$, $b = \cosh(\Delta^2)\alpha_1$, and maximum likelihood decoding, are negligible for any values of Δ small enough to be useful in practice. Here, we will use a constant bin size $b = \cosh(\Delta^2)\alpha_1$ in all subsequent results.

Now, we consider inefficient homodyne detection of the position quadrature q, which can be described by the POVM elements:

$$W_\eta(X) = \mathcal{N}\int dx\ \ |x\rangle_q\langle x| \exp\Big(-\frac{\eta}{1-\eta}(x-X)^2\Big) \tag{3.10}$$

where $X \in \mathbb{R}$ is the recorded outcome of the measurement and $\mathcal{N}$ is a normalisation constant. This POVM already accounts for the rescaling of the measurement outcome due to loss occurring from the inefficient measurement. We define the measurement error of the logical readout as $M_{\text{error}} = \big(P(1|0) + P(0|1)\big)/2$, where $P(1|0)$ is the probability of recording a -1 measurement outcome given the initial state was $|\bar{0}_\Delta\rangle$. This can be calculated by evaluating the integral:

$$P(1|0) = \sum_{t\in\mathbb{Z}} \int_{(2t+\frac{1}{2})b}^{(2t+\frac{3}{2})b} dX\ \ \mathrm{tr}\left(W_\eta(X)\,|\bar{0}_\Delta\rangle\langle\bar{0}_\Delta|\,\right) \tag{3.11}$$

After substituting the expression for ${}_q\langle x|\bar{0}_\Delta\rangle$ (which may be obtained from eqs. (3.2) and (3.3)) and evaluating the integrals over X and x, this gives an exact infinite series for the error probability $P(1|0)$, which can be evaluated numerically by truncating the infinite series (see section 3.B). The values obtained using this semi-analytical method

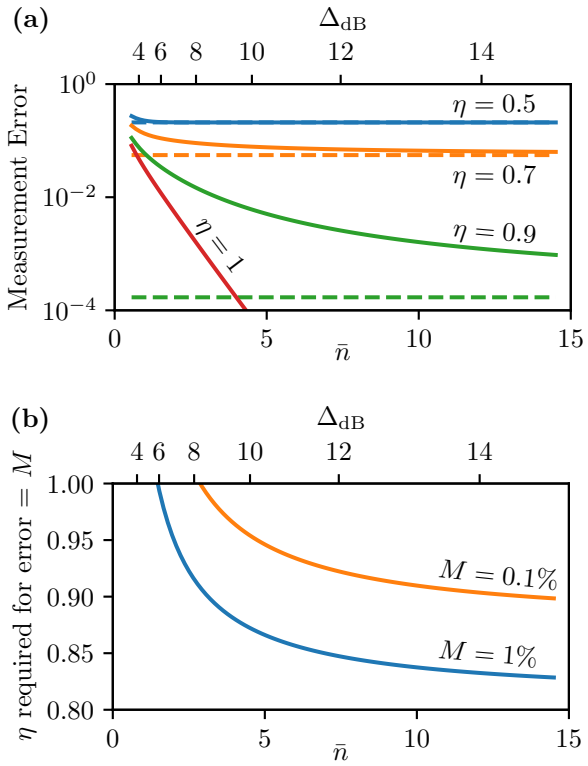

Figure 3.5: The effect of inefficient homodyne measurements on the readout errors of approximate square GKP codestates. (a) Logical Z measurement error $\big(P(0|1)+P(1|0)\big)/2$ of approximate GKP codestates (with average photon number $\bar{n}$ and squeezing Δ_{dB}) measured by a position measurement with efficiency η. For finite η, the limiting measurement error as the GKP codestate approaches the ideal limit ($\bar{n} \to \infty$) is plotted as a dashed line. (b) The efficiency required to reach a target logical Z measurement error rate M, as a function of the GKP squeezing/photon number.

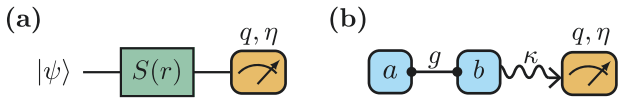

Figure 3.6: Two alternative position measurement schemes. (a) The initial state is subjected to phase-sensitive amplification, equivalent to an application of the squeezing operator $S(r) = e^{\frac{r}{2}(a^2 - a^{\dagger 2})}$, followed by a position measurement with efficiency η. (b) The initial state (stored in cavity a) is coupled to a read-out mode initialised in the vacuum state (stored in cavity b) via the Hamiltonian $H/\hbar = -gq_1p_2$, which may be implemented using the circuit QED designs presented in section 3.3. Simultaneously, the read-out mode's position quadrature is measured at a rate κ via homodyne detection with efficiency η. In both schemes, we show that the whole scheme is equivalent to a single position measurement with an effective efficiency η_{eff}, which can be made to be greater than the efficiency of the physical position measurement η.

are indistinguishable from those obtained using a direct Fock space simulation, but the semi-analytic results can be used to probe smaller values of Δ than would be possible due to the truncation dimension of the numerical simulation. We plot the measurement error M_{error} as a function of Δ and η for the square GKP code in fig. 3.5(a), and show the efficiency η required to reach various target measurement error rates in fig. 3.5(b). Additionally, we derive in section 3.B the following approximation to the measurement error, which holds for small Δ:

$$M_{\text{error}} \approx \text{erfc}\left(\frac{1}{2}\alpha_1\left(\Delta^2 + \frac{1-\eta}{\eta}\right)^{-1/2}\right) \tag{3.12}$$

where $\text{erfc}(x) \approx e^{-x^2}/x\sqrt{\pi}$ for large x. These results demonstrate that M_{error} is highly sensitive with respect to the measurement efficiency η: even at $\eta = 0.7$, which is close to the current state-of-the-art, the minimum achievable measurement error rate is $M_{\text{error}} \approx 6.7\%$ as $\Delta \to \infty$. This mirrors results obtained in [33] in their analysis of teleportation error-correction schemes, and motivates the need to improve the effective efficiency of homodyne detection for use in GKP codes. Concretely, one needs a measurement efficiency of $\eta \approx 0.85$ to reach a measurement error of 1%, or $\eta \approx 0.92$ for 0.1%, using a square GKP code with $\Delta_{\text{dB}} \approx 12$ dB.

To achieve such efficiencies, we consider two alternative schemes (illustrated in fig. 3.6) for performing homodyne detection to improve the effective efficiency of the measurement. In the first scheme, we simply consider applying a single-mode squeezing operator $S(r) = \exp\left(r(a^{\dagger 2} - a^2)/2\right)$ (where $r > 0$ corresponds to squeezing the momentum quadrature and $r < 0$ corresponds to squeezing the position quadrature), before performing homodyne detection with efficiency η. We quote the amount of squeezing in decibels as $S_{\text{dB}} = (20 \log_{10} e)\, r$. Intuitively,

amplifying the position quadrature should improve the effective efficiency of the measurement by improving the separation between different position eigenstates. We can write the effective POVM elements of the squeezing/measurement sequence as $V_{r,\eta}(X') = S^\dagger(r)W_\eta(X')S(r)$, where X' is the measurement outcome of the inefficient homodyne measurement eq. (3.10). By defining $X = e^{-r}X'$, we obtain $V_{r,\eta}(X) = W_{\eta_{\text{eff}}}(X)$, where

$$\eta_{\text{eff}} = \left(1 + e^{-2r}(\eta^{-1} - 1)\right)^{-1} \tag{3.13}$$

Using this, we can calculate the amount of amplification required to increase the effective efficiency of the measurement; for example, if the physical efficiency is $\eta = 0.7$, we can achieve $\eta_{\text{eff}} = 0.85$ (or a 1% measurement error for $\Delta_{\text{dB}} = 12$ dB) with a squeezing of $S_{\text{dB}} \approx 4$ dB, and $\eta_{\text{eff}} = 0.92$ (0.1% error) with $S_{\text{dB}} \approx 7$ dB.

Although such levels of amplification are achievable, there are a number of practical drawbacks to this simple scheme. First, this requires the ability to perform single-mode squeezing on each GKP mode, reducing the hardware efficiency of the computation. Second, such amplification is already maximised in the measurement sequence, so applying additional amplification prior to the measurement may result in the accumulation of unwanted errors. Finally, such a scheme requires the GKP mode itself to be directly released into the measurement sequence, which either requires a change in the quality factor of the GKP mode itself, or requires the coupling to a second readout mode.

To combat these issues we consider a second more realistic scheme in fig. 3.6(b). We consider a high Q GKP mode with loss rate γ coupled to a low Q read-out mode via a Hamiltonian $H = -gq_1p_2$ (where 1 refers to the GKP mode and 2 to the read-out mode). Such a Hamiltonian can be engineered using the circuits discussed in section 3.3 and is identical to the Hamiltonian for a controlled-X between two square GKP qubits. Homodyne detection is performed on the position quadrature of the read-out mode with efficiency η, which we consider to be occurring at a rate κ simultaneously with the coupling. To analyse this system we consider the behaviour of the system in short timescales $t \ll 1/\gamma$, and so we can neglect the loss γ occurring on the GKP mode. In this regime, the quantity of interest is the time required to perform the measurement with a given desired efficiency. Here, we use quantum trajectories to solve exactly for the POVM of the system using the method of Ref. [64]. We relegate the details of the derivation of the POVM to section 3.C so that we can focus on the results of our analysis here.

The experimentally observed measurement outcome is given by the observed photocurrent $I(t)$ of the detector (see section 3.C for details). We find that the

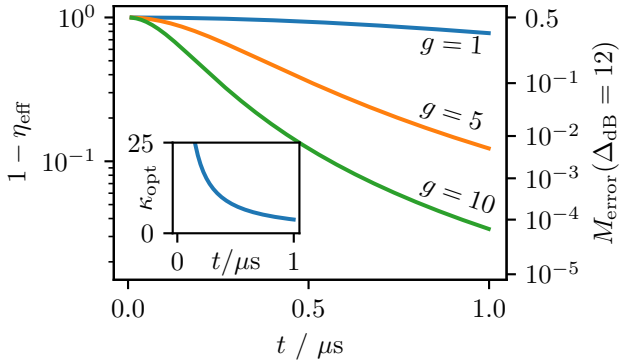

Figure 3.7: (main) Effective measurement inefficiency $1-\eta_{\text{eff}}$ of the two-mode measurement scheme, as a function of the coupling strength $g/2\pi$ MHz and measurement time t, for a fixed physical efficiency $\eta = 0.75$. The two-mode measurement scheme consists of coupling the GKP mode to an ancilla initiated in the vacuum state via the Hamiltonian $H = -gq_1p_2$, and performing homodyne detection for some finite time t on the ancilla mode at a rate κ and efficiency η. κ is optimised to maximise the effective measurement efficiency. The right axis shows the corresponding logical measurement error of a GKP state with $\Delta_{\text{dB}} = 12$ when measured with homodyne detection of efficiency η_{eff}. (inset) The optimal measurement rate $\kappa_{\text{opt}}/2\pi$ MHz as a function of the measurement time t.

resulting POVM depends not on the entire measurement record from time 0 to t, but only on one integral of the observed photocurrent $X = -\sqrt{\kappa/(8g^2\tau^2\eta)}\int_0^t dt'\,(1-e^{-\kappa t'/2})I(t')$, where $\tau = t-(1-e^{-\kappa t/2})(3-e^{-\kappa t/2})/\kappa$. Given the initial state of the ancilla mode ρ_b, we can write a single-mode POVM element corresponding to a position measurement of X:

$$T_{g,\kappa,\eta,t}(X) = \mathcal{N}\int dx\ |x\rangle_{q_1}\langle x| \int d\tilde{x}\ {}_{q_2}\langle\tilde{x}|\rho_b|\tilde{x}\rangle_{q_2} \times \exp\left\{-\frac{1}{c}\Big[X-x-\frac{1}{2g\tau}\big(1-e^{-\kappa t/2}\big)^2\tilde{x}\Big]^2\right\} \quad (3.14)$$

where X represents the position measurement outcome, $\mathcal{N}$ is a normalisation constant, and $c = \big[\kappa\tau-\eta(1-e^{-\kappa t/2})^4\big]/(4g^2\tau^2\eta)$. Substituting $\rho_b = |0\rangle\langle 0|$ results in a POVM that corresponds to an inefficient homodyne measurement (eq. (3.10)) with effective efficiency:

$$\eta_{\text{eff}} = \frac{4g^2\tau^2\eta}{4g^2\tau^2\eta+\kappa\tau-\eta(1-e^{-\kappa t/2})^4} \quad (3.15)$$

We analyse this result as follows. First, we optimise the measurement rate κ to maximise the effective measurement efficiency for fixed g, t, η. Interestingly, the resulting κ_{opt} depends only on the measurement time t and is given approximately by $\kappa_{\text{opt}}/2\pi \approx 3.79/t$ (see inset to fig. 3.7). We comment that the dependence of η_{eff} on κ is very weak near κ_{opt}, so a fine-tuning of κ is not required to achieve competitive

effective efficiencies. Next, we fix the physical efficiency of the measurement at $\eta = 0.75$, and plot η_{eff} as a function of t and g in fig. 3.7. The plot demonstrates that for a coupling strength of $g = 10 \times 2\pi$ MHz, one can achieve $\eta_{\text{eff}} = 0.85$ (1% read-out error for $\Delta_{\text{dB}} = 12$ dB) in a measurement time $t \approx 0.45$ μs, and $\eta_{\text{eff}} = 0.92$ (0.1% error) after $t \approx 0.63$ μs.

We are interested in the required measurement time to achieve a given effective efficiency (and hence logical measurement error rate) for two reasons. First, if the measurement time is too long, loss on the GKP mode will become significant. Second, we wish for logical readout to be conducted on a time scale comparable to that of other superconducting architectures, such as those based on transmon qubits. In transmon qubits, logical readout typically takes on the order of hundreds of nanoseconds [15], which is on the same order of magnitude as our estimates for the time for our measurement scheme. Stronger coupling strengths would help reduce the measurement time required in our scheme.

We remark that the effect of changing the physical efficiency η does not have a dramatic effect on the effective efficiency η_{eff} of the result shown in fig. 3.7. Indeed, increasing the physical efficiency of the measurement from $\eta = 0.5$ to $\eta = 1$ only results in a roughly 30% decrease in the required measurement time to reach a given effective efficiency. Likewise, initialising the ancilla mode in a squeezed vacuum state prior to turning on the coupling has an almost negligible effect on η_{eff}. As such, the coupling strength g is the most important factor (along with the GKP mode squeezing Δ) in reducing the logical measurement error in GKP modes.

3.6 Discussion and Conclusions

In this work, we have given a concrete proposal to implement logical Clifford gates and read-out in superconducting circuits. We began by presenting our scheme for performing Clifford circuits using generalised controlled gates implemented by four- or three-wave mixing circuits, where single-qubit Clifford gates are accounted for by updating the phase of a local oscillator in the implementation of each two-qubit gate. Next, we presented an average gate fidelity metric for quantifying the quality of logical Clifford gates, and analysed the performance of ideal logical gates when subjected to errors due to the approximate GKP codestates. We presented a general method to mitigate the spreading of errors due to these logical gates and used the newly developed stabiliser subsystem decomposition (chapter 4) to analyse the results. Finally, we considered the effects of homodyne detection inefficiencies on the rate of logical measurement errors. We proposed a scheme that can achieve feasible

error rates even with a low measurement efficiency, and analysed the system using quantum trajectories [64].

While our analysis of the performance of logical Clifford gates gave significant insight into how to mitigate the spreading of errors in the system, we did not consider the effects of non-ideal gate execution. One could use the theoretical analysis conducted in [66] to determine the leading sources of error in the implementation of a given generalised controlled gate. Then, one could include these effects to produce an estimate of the average gate fidelity of each generalised controlled gate in the presence of realistic noise sources.

Our scheme for mitigating the effects of errors spreading through logical Clifford gates works exactly in the case of ideal gate implementation and error correction. However, including the effects of approximate error correction would likely affect this result. In particular, the modified error correction patch $\Sigma_{\bar{A}}\mathcal{V}_{\boldsymbol{\alpha},\boldsymbol{\beta}}$ that must be made to accommodate the logical gate, would likely reduce the fidelity of an approximate decoding operation. Moreover, the maximum-likelihood patch for decoding may be altered by the approximate decoding, and it would be interesting to see how additional modifications to the patch could be made to balance the errors from before and after the application of the gate. Alternatively, it may be possible to use an approximate GKP codestate with a modified envelope (transformed for example by $\Sigma_{\bar{A}}$) to make the errors from the approximate correction consistent with the modified error correction patch. We leave such considerations to future studies on the topic.

To perform logical read-out of GKP codestates, we presented a scheme utilising quadrature-quadrature coupling to an ancilla mode to implement fast, high-fidelity logical read-out. However, this scheme requires the use of a low-Q read-out ancilla prepared in the vacuum state, increasing the overhead required for computation with GKP codes. Moreover, the performance of our scheme is highly sensitive to the strength g of the quadrature-quadrature coupling between the GKP and ancilla modes, but the values of g that are feasible in GKP experiments remain to be seen. Finally, the high-Q GKP mode and low-Q readout ancilla must be coupled with a device with a large on-off ratio to prevent unwanted leakage from the GKP mode to the ancilla.

Appendices

3.A Derivation of Gate Error Estimate

In this section, we derive the approximate expression of the average gate infidelity $1 - \bar{F}_{I,\mathcal{P},\Delta}$ obtained from decoding an approximate GKP codestate over a patch $\mathcal{P}$. The approximate expression is valid in the limit $\Delta \to 0$. We will use the recently developed GKP stabiliser subsystem decomposition formalism in our derivation (chapter 4). In particular, recall that $\mathcal{E}_{I,\mathcal{P},\Delta}$ is the CPTP map corresponding to an orthonormal encoding of ρ into (orthonormalised) approximate GKP codestates followed by an ideal round of error correction over the patch $\mathcal{P}$, and decoding of the resultant ideal codestate [eq. (3.7)].

We start with a formula for the average gate fidelity which is given in Ref. [67]:

$$\bar{F}(\mathcal{E}, I) = \frac{1}{d^2(d+1)} \sum_{j,k} \alpha_{jk} \mathrm{tr}\big(U_j^\dagger \mathcal{E}(\rho_k)\big) + \frac{1}{d+1} \tag{3.16}$$

where $\mathcal{E}$ is a CPTP map, d is the dimension of the system, $\{\rho_k\}$ forms a basis of density operators, $\{U_j\}$ forms an orthogonal basis of $\mathcal{L}(\mathcal{H})$, and the coefficients α_{jk} are such that $U_j = \sum_k \alpha_{jk}\rho_k$.

Applying eq. (3.16) to $\mathcal{E}_{I,\mathcal{P},\Delta}$ in the single-qubit case, we set $d = 2$, $\rho_0 = I/2$, $\rho_k = |+_k\rangle\langle+_k|$ for $k = 1, 2, 3$ (where $|+_k\rangle$ is the $+1$ eigenstate of σ_k), and $U_j = \sigma_j$ for $j = 0, 1, 2, 3$ (with $\sigma_0 = I$), giving:

$$\bar{F}(\mathcal{E}, I) = \frac{1}{6}\left(\sum_{j=0}^{3} \mathrm{tr}\big(\sigma_j \mathcal{E}(\rho_j)\big) - \sum_{j=1}^{3} \mathrm{tr}\big(\sigma_j \mathcal{E}(\rho_0)\big) \right) + \frac{1}{3} \tag{3.17}$$

First, we have that $\mathrm{tr}\big(\sigma_0 \mathcal{E}(\rho_0)\big) = 1$ since $\mathcal{E}$ is a CPTP map. Now, substituting $\mathcal{E} = \mathcal{E}_{I,\mathcal{P},\Delta}$, we can approximate $\mathrm{tr}\big(\sigma_j \mathcal{E}(\rho_0)\big) \approx 0$ for $j = 1, 2, 3$ – in other words, the expectation value of the approximate maximally mixed state with any of the logical Pauli operators is 0. This gives an estimate of the average gate infidelity as:

$$1 - \bar{F}_{I,\mathcal{P},\Delta} \approx \frac{1}{6} \sum_{j=1}^{3} \big(1 - \langle \bar{+}_{j,\Delta} | \bar{\sigma}_{j,\mathrm{id}} | \bar{+}_{j,\Delta} \rangle\big) \tag{3.18}$$

where $|\bar{+}_{j,\Delta}\rangle \propto e^{-\Delta^2 a^\dagger a} |\bar{+}_j\rangle$ and $\bar{\sigma}_{j,\mathrm{id}} = \sigma_j \otimes_{\mathcal{G}} I$ is the "ideal" logical Pauli operator σ_j, where $\otimes_{\mathcal{G}}$ is across the stabiliser subsystem decomposition of the GKP code with parameters $\mathcal{G} = (\Sigma, 2, \mathcal{P})$ (see chapter 4 for an introduction to the stabiliser subsystem decomposition). One can understand this ideal logical Pauli operator $\bar{\sigma}_{j,\mathrm{id}}$

by considering its eigenstates: $|\phi\rangle$ is a $+1$-eigenstate (respectively, -1-eigenstate) of $\bar{\sigma}_j$ if (and only if) a round of ideal error correction over $\mathcal{P}$ results in the ideal codestate $|\bar{+}_j\rangle$ ($|\bar{-}_j\rangle$) with probability 1. Note that this is *not* equivalent to the measurement Pauli operators $\bar{\sigma}_{j,\text{meas}}$ defined by a homodyne detection scheme as in section 3.5. This is also *not* equivalent to the typical definition of logical Pauli operators $\bar{\sigma}_j$ as a translation operator, which has eigenvalues $e^{i\theta}$ for all θ.

To estimate the expectation value $\langle\bar{+}_{j,\Delta}|\bar{\sigma}_{j,\text{id}}|\bar{+}_{j,\Delta}\rangle$, we use the translation-operator expansion of the envelope operator $e^{-\Delta^2 a^\dagger a}$:

$$e^{-\Delta^2 a^\dagger a} = \frac{1}{2\pi(1-e^{-\Delta^2})}\int_{\mathbb{R}^2} d^2\mathbf{v}\; e^{-\frac{|\mathbf{v}|^2}{4}\coth\left(\frac{\Delta^2}{2}\right)} T(\mathbf{v}) \tag{3.19}$$

and we notate $|\tilde{+}_{j,\Delta}\rangle = e^{-\Delta^2 a^\dagger a}\,|\bar{+}_j\rangle$ for the un-normalised approximate GKP codewords.

To continue we must first estimate the norm of $|\tilde{+}_{j,\Delta}\rangle$ as follows:

$$\langle\tilde{+}_j|\tilde{+}_j\rangle = \langle\bar{+}_j|e^{-2\Delta^2 a^\dagger a}|\bar{+}_j\rangle = \frac{1}{2\pi(1-e^{-2\Delta^2})}\int_{\mathbb{R}^2} d^2\mathbf{v}\, e^{-\frac{|\mathbf{v}|^2}{4}\coth(\Delta^2)}\langle\bar{+}_j|T(\mathbf{v})|\bar{+}_j\rangle \tag{3.20}$$

Now, $\langle\bar{+}_j|T(\mathbf{v})|\bar{+}_j\rangle$ is only non-zero if $T(\mathbf{v})$ is product of stabilisers and $\bar{\sigma}_j$ logical operators. Indeed, one can show:

$$\langle\bar{+}_j|T(\mathbf{v})|\bar{+}_j\rangle = \sum_{\mathbf{s}\in\mathcal{L}_j}\delta^2(\mathbf{v}-\mathbf{s}) \tag{3.21}$$

where $\mathcal{L}_j$ is the lattice of vectors $\mathbf{s}$ such that $T(\mathbf{s})$ is a product of stabilisers and logical $\bar{\sigma}_j$ operators. For example, when $j=3$ such that $\sigma_3 = Z$, $\mathcal{L}_3 = \{2m\boldsymbol{\alpha} + n\boldsymbol{\beta} \mid m,n \in \mathbb{Z}\}$. In the limit as $\Delta \to 0$, the prefactor $e^{-|\mathbf{s}|^2\coth(\Delta^2)/4}$ is exponentially suppressed as $|\mathbf{s}|$ increases. Thus we can approximate the integral in eq. (3.20) to 1 by keeping only the $\mathbf{s}=\mathbf{0}$ term in the sum in eq. (3.21). With this, eq. (3.20) simplifies to:

$$\langle\tilde{+}_j|\tilde{+}_j\rangle \approx \frac{1}{4\pi\Delta^2} \tag{3.22}$$

where we have also expanded $1-e^{-2\Delta^2} \approx 2\Delta^2$ to lowest order.

Now we move on to estimate $\langle\tilde{+}_{j,\Delta}|\bar{\sigma}_{j,\text{id}}|\tilde{+}_{j,\Delta}\rangle$. Equation (3.19) can be interpreted as a coherent superposition of translation errors applied to the ideal GKP codestate $|\bar{+}_j\rangle$. In particular, each state $T(\mathbf{v})\,|\bar{+}_j\rangle$ is an eigenstate of the Pauli operator $\bar{\sigma}_{j,\text{id}}$ since an ideal round of error correction will always result in a state which is equal to $\bar{P}\,|\bar{+}_j\rangle$ for some logical Pauli $\bar{P}$. Let $\mathcal{R}_\pm \subset \mathbb{R}^2$ be the subset of $\mathbb{R}^2$ such that $T(\mathbf{v})\,|\bar{+}_j\rangle$ is a ±1-eigenstate of $\bar{\sigma}_j$ when $\mathbf{v}\in\mathcal{R}_\pm$. Then these regions correspond to

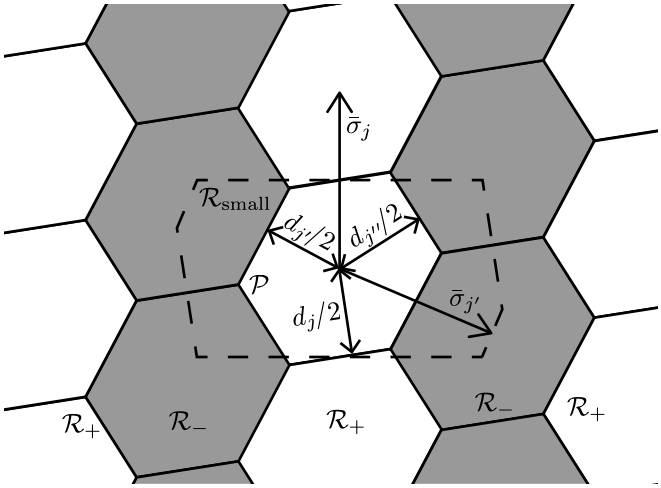

Figure 3.8: Illustration of the regions and quantities used in section 3.A for an arbitrary GKP code with decoding patch $\mathcal{P}$. $\mathcal{R}_+$ (white) is the union of patches $\mathcal{P}$ displaced by any product of stabilisers or $\bar{\sigma}_j$ operators. $\mathcal{R}_-$ (shaded) is the remaining region of $\mathbb{R}^2$, which can be equally viewed as $\mathcal{R}_+$ displaced by either of the other logical Pauli operators $\bar{\sigma}_{j'}$. $\mathcal{R}_+$ consists of the set of vectors $\mathbf{v}$ with the property that $T(\mathbf{v})\,|\bar{+}_j\rangle$ is a $+1$-eigenstate of the $\bar{\sigma}_j$ operator. $\mathcal{R}_{\text{small}}$ (dashed) is the Voronoi cell of the lattice $\mathcal{L}_j$, generated by the vectors corresponding to stabilisers and logical $\bar{\sigma}_j$ operators. $\mathcal{R}_{\text{small}}$ has the property that the overlap of any two states $T(\mathbf{v})\,|\bar{+}_j\rangle$ and $T(\mathbf{v}')\,|\bar{+}_j\rangle$ is 0 whenever $\mathbf{v} \neq \mathbf{v}'$ and $\mathbf{v}, \mathbf{v}' \in \mathcal{R}_{\text{small}}$. $d_j, d_{j'}, d_{j''}$ are the distances of the patch $\mathcal{P}$ corresponding to twice the smallest vector which would cause a $\bar{\sigma}_j$ error when applied to an ideal codestate.

translated patches (see fig. 3.8):

$$\mathcal{R}_+ = \bigcup_{\mathbf{s}\in\mathcal{L}_j} \mathbf{s} + \mathcal{P} \tag{3.23a}$$

$$\mathcal{R}_- = \mathbb{R}^2 \setminus \mathcal{R}_+ \tag{3.23b}$$

where $\mathcal{P}$ is the patch of the decoder, $\mathbf{s}+\mathcal{P} = \{\mathbf{s}+\mathbf{v} \mid \mathbf{v}\in\mathcal{P}\}$, $\mathcal{L}_j$ is the same lattice of vectors $\mathbf{s}$ as in the previous paragraph corresponding to stabilisers and logical Paulis, and $\setminus$ is the "setminus" symbol. Based on these definitions, we can write:

$$|\tilde{\mp}_j\rangle = |\phi_+\rangle + |\phi_-\rangle \tag{3.24}$$

where $|\phi_\pm\rangle$ is a ±1-eigenstate of $\bar{\sigma}_{j,\mathrm{id}}$, given by:

$$|\phi_\pm\rangle = \frac{1}{2\pi(1-e^{-\Delta^2})}\int_{\mathcal{R}_\pm} d^2\mathbf{v}\, e^{-\frac{|\mathbf{v}|^2}{4}\coth\left(\frac{\Delta^2}{2}\right)} T(\mathbf{v})\,|\bar{\mp}_j\rangle \tag{3.25}$$

With this, the expectation value can be written:

$$\begin{aligned}\langle\tilde{\mp}_{j,\Delta}|\bar{\sigma}_{j,\mathrm{id}}|\tilde{\mp}_{j,\Delta}\rangle &= \langle\phi_+|\phi_+\rangle - \langle\phi_-|\phi_-\rangle \\ &= \langle\tilde{\mp}_{j,\Delta}|\tilde{\mp}_{j,\Delta}\rangle - 2\,\langle\phi_-|\phi_-\rangle\end{aligned} \tag{3.26}$$

So we only need to estimate:

$$\begin{aligned}\langle\phi_-|\phi_-\rangle = \frac{1}{4\pi^2\left(1-e^{-\Delta^2}\right)^2}\int_{\mathcal{R}_-} d^2\mathbf{v}\int_{\mathcal{R}_-} d^2\mathbf{v}'\Bigg(e^{-\frac{|\mathbf{v}|^2+|\mathbf{v}'|^2}{4}\coth\left(\frac{\Delta^2}{2}\right)} \\ \times\,\langle\bar{\mp}_j|T(-\mathbf{v})T(\mathbf{v}')|\bar{\mp}_j\rangle\Bigg)\end{aligned} \tag{3.27}$$

Again, the inner product $\langle\bar{\mp}_j|T(-\mathbf{v})T(\mathbf{v}')|\bar{\mp}_j\rangle = e^{i(v_1v_2'-v_2v_1')/2}\,\langle\bar{\mp}_j|T(\mathbf{v}'-\mathbf{v})|\bar{\mp}_j\rangle$ is non-zero only when $\mathbf{v}'-\mathbf{v}\in\mathcal{L}_j$. As $\Delta\to0$, we can approximate the integrals over $\mathcal{R}_-$ as integrals over a much smaller region $\mathcal{R}_{\mathrm{small}}\cup\mathcal{R}_-$, where $\mathcal{R}_{\mathrm{small}}$ is the Voronoi cell of the lattice $\mathcal{L}_j$. This region has the convenient property that $\langle\bar{\mp}_j|T(\mathbf{v}'-\mathbf{v})|\bar{\mp}_j\rangle = \delta(\mathbf{v}'-\mathbf{v})$ for $\mathbf{v},\mathbf{v}'\in\mathcal{R}_{\mathrm{small}}$, and as such eq. (3.27) simplifies to:

$$\langle\phi_-|\phi_-\rangle \approx \frac{1}{4\pi^2\left(1-e^{-\Delta^2}\right)^2}\int_{(\mathcal{R}_-\cup\mathcal{R}_{\mathrm{small}})} d^2\mathbf{v}\, e^{-\frac{|\mathbf{v}|^2}{2}\coth\left(\frac{\Delta^2}{2}\right)} \tag{3.28a}$$

$$\approx \frac{1}{4\pi^2\Delta^4}\int_{(\mathcal{R}_-\cup\mathcal{R}_{\mathrm{small}})} d^2\mathbf{v}\, e^{-|\mathbf{v}|^2/\Delta^2} \tag{3.28b}$$

where in the second line we have taken the lowest order expansion of the non-

linear functions of Δ^2. At this point, we find it useful to define three distances d_j corresponding to twice the length of the shortest translation from the origin which results in a logical Pauli σ_j error. Note that these distances are in general *not* equal simply to the length of the translation $\bar{\sigma}_j$, and instead depend on the geometry of the choice of patch $\mathcal{P}$ (see fig. 3.8). For small Δ, we can use the circular symmetry of the integrand to approximate the integral in terms of these distances:

$$\langle \phi_- | \phi_- \rangle \approx \frac{1}{4\pi^2\Delta^4} \sum_{\substack{j'=1 \\ j'\neq j}}^{3} 2 \int_{d_{j'}/2}^{\infty} dv_1 \int_{-\infty}^{\infty} dv_2 \, e^{-|\mathbf{v}|^2/\Delta^2} \tag{3.29}$$

$$= \frac{1}{4\pi\Delta^2} \sum_{\substack{j'=1 \\ j'\neq j}}^{3} \operatorname{erfc}\left(\frac{d_{j'}}{2\Delta}\right) \tag{3.30}$$

where erfc is the complementary error function defined by:

$$\operatorname{erfc}(x) = \frac{2}{\sqrt{\pi}} \int_x^{\infty} dt \, e^{-t^2} \tag{3.31}$$

For large x, we have that $\operatorname{erfc}(x) \approx e^{-x^2}/(x\sqrt{\pi})$, finally giving:

$$\langle \phi_- | \phi_- \rangle \approx \frac{1}{2\pi^{3/2}\Delta} \sum_{\substack{j'=1 \\ j'\neq j}}^{3} \frac{e^{-d_{j'}^2/(4\Delta^2)}}{d_{j'}} \tag{3.32}$$

With this, we can now estimate the expectation value

$$\langle \bar{+}_{j,\Delta} | \bar{\sigma}_{j,\mathrm{id}} | \bar{+}_{j,\Delta} \rangle = 1 - \frac{2 \langle \phi_- | \phi_- \rangle}{\langle \tilde{+}_j | \tilde{+}_j \rangle} \tag{3.33}$$

$$\approx 1 - \frac{4\Delta}{\sqrt{\pi}} \sum_{\substack{j'=1 \\ j'\neq j}}^{3} \frac{e^{-d_{j'}^2/(4\Delta^2)}}{d_{j'}} \tag{3.34}$$

Substituting this back into eq. (3.18) gives the average gate infidelity:

$$1 - \bar{F}_{I,\mathcal{P},\Delta} \approx \frac{4\Delta}{3\sqrt{\pi}} \sum_{j=1}^{3} \frac{e^{-d_j^2/(4\Delta^2)}}{d_j} \tag{3.35}$$

Now, we define $d = \max_j(d_j)$ and a as the number of distances with $d_j = d$. We can neglect the contributions to the infidelity arising from all the distances $d_j > d$ and arrive at:

$$1 - \bar{F}_{I,\mathcal{P},\Delta} \approx a \frac{4\Delta}{3d\sqrt{\pi}} e^{-d^2/(4\Delta^2)} \tag{3.36}$$

which is simply eq. (3.8) evaluated for the single-qubit case.

In the two-qubit case, the derivation is much the same but with different combinatorial factors. Applying eq. (3.16) to $\mathcal{E}_{I,\mathcal{P},\Delta}$ in the two-qubit case, and again making the assumption that $\mathrm{tr}\big(\sigma_j \mathcal{E}(\rho_0)\big) \approx 0$ for the maximally mixed state ρ_0, we obtain the approximation:

$$1 - \bar{F}_{I,\mathcal{P},\Delta} \approx \frac{1}{20} \sum_{\substack{j_1,j_2=0 \\ (j_1,j_2)\neq(0,0)}}^{3} \big(1 - \langle \bar{+}_{j_1,j_2,\Delta} |\, \bar{\sigma}_{j_1,j_2,\mathrm{id}} \,| \bar{+}_{j_1,j_2,\Delta} \rangle\big) \tag{3.37}$$

where we have written $|\bar{+}_{j_1,j_2,\Delta}\rangle$ as a shorthand for $|\bar{+}_{j_1,\Delta}\rangle \otimes |\bar{+}_{j_2,\Delta}\rangle$, and $\bar{\sigma}_{j_1,j_2,\mathrm{id}} = (\sigma_{j_1} \otimes \sigma_{j_2}) \otimes_{\mathcal{G}_2} (I \otimes I)$ in the two-mode stabiliser subsystem decomposition $\mathcal{G}_2 = \big(\Sigma \oplus \Sigma, (2,2), \mathcal{P}\big)$. The intuitive definition of $\bar{\sigma}_{j_1,j_2,\mathrm{id}}$ still holds in the two-mode case. However, we note that the patch $\mathcal{P}$, in general, does not need to be a "product" of two single-mode patches $\mathcal{P}_1 \times \mathcal{P}_2$ across the two modes, so $\bar{\sigma}_{j_1,j_2,\mathrm{id}}$ is not, in general, a tensor product operator across the two modes.

At this point we introduce the notation $j = (j_1, j_2)$ and the distances d_j corresponding to twice the length of the shortest translation from the origin which results in a logical Pauli $\sigma_{j_1} \otimes \sigma_{j_2}$ error. One can follow the methods of the single-mode case to arrive at an estimate for the expectation value:

$$\langle \bar{+}_{j,\Delta} | \bar{\sigma}_{j,\mathrm{id}} | \bar{+}_{j,\Delta} \rangle \approx 1 - \frac{4\Delta}{\sqrt{\pi}} \sum_{j'} \frac{e^{-d_{j'}^2/(4\Delta^2)}}{d_{j'}} \tag{3.38}$$

where the summation occurs over $j' = (j_1', j_2') \neq (0,0)$ such that $\sigma_{j_1} \otimes \sigma_{j_2}$ and $\sigma_{j_1'} \otimes \sigma_{j_2'}$ anti-commute. Note that out of the 15 non-identity two-qubit Pauli operators, exactly 8 will anti-commute with any given Pauli. Applying this to eq. (3.37) then results in:

$$1 - \bar{F}_{I,\mathcal{P},\Delta} \approx \frac{8\Delta}{5\sqrt{\pi}} \sum_{\substack{j_1,j_2=0 \\ (j_1,j_2)\neq(0,0)}}^{3} \frac{e^{-d_j^2/(4\Delta^2)}}{d_j} \tag{3.39}$$

Finally, defining $d = \max_j(d_j)$ and a as the number of distances with $d_j = d$, we once again recover eq. (3.8) for the two-qubit case.

As a final remark, we note that by keeping track of the combinatorial factors in the general case of n modes, one finds that the prefactor c in eq. (3.8) can be written:

$$c = \frac{2^n}{2(2^n + 1)} \tag{3.40}$$

which reproduces the values of 1/3 and 2/5 in the single- and two-mode cases.

3.B Derivation of Measurement Error Estimate

In this appendix, we derive the approximate expression of the measurement error M_{error} obtained from measuring an approximate GKP codeword $|\bar{0}_\Delta\rangle$ in the Pauli Z logical basis via a binned measurement with bin size $b \approx \alpha_1$ and efficiency η. The approximate expression is valid in the limit $\Delta \to 0$. In this limit, we also approximate $P(0|1) \approx P(1|0)$, and so $M_{\text{error}} \approx P(1|0)$, the probability of obtaining a measurement outcome of -1 given an initial state $|\bar{0}_\Delta\rangle$. Recall the position wavefunction of the approximate codestate $|\bar{0}_\Delta\rangle$ is given by:

$$ {}_q\langle x|\bar{0}_\Delta\rangle = \mathcal{N}_0 \sum_{s\in\mathbb{Z}} e^{2is^2\alpha_1\alpha_2} e^{-\frac{1}{2}(2s\alpha_1)^2\tanh(\Delta^2)} e^{-\frac{1}{2}\coth(\Delta^2)[x-2s\alpha_1\operatorname{sech}(\Delta^2)]^2} \tag{3.2′} $$

where the GKP code is defined by the vectors $\boldsymbol{\alpha} = [\alpha_1, \alpha_2]^T$ and $\boldsymbol{\beta} = [0, \pi/\alpha_1]^T$ which is rotated such that a logical measurement of $\bar{Z} = e^{i\pi q/\alpha_1}$ can be done with a binned position measurement. Recall that a position measurement with efficiency η is described by the POVM elements:

$$ W_\eta(X) = \mathcal{N}_\eta \int dx \;\; |x\rangle_q\langle x| \exp\Big(-\frac{\eta}{1-\eta}(x-X)^2 \Big) \tag{3.10} $$

where $X \in \mathbb{R}$ is the recorded outcome of the measurement and $\mathcal{N}_\eta = \sqrt{\eta/\big(\pi(1-\eta)\big)}$ is a normalisation constant such that $\int dX\, W_\eta(X) = I$. Finally, recall that the probability of recording a -1 measurement outcome given an initial state $|\bar{0}_\Delta\rangle$ is:

$$ P(1|0) = \sum_{t\in\mathbb{Z}} \int_{(2t+\frac{1}{2})b}^{(2t+\frac{3}{2})b} dX \;\; \operatorname{tr}\big(W_\eta(X)\, |\bar{0}_\Delta\rangle\langle\bar{0}_\Delta| \big) \tag{3.11} $$

$$ = \mathcal{N}_\eta \sum_{t\in\mathbb{Z}} \int_{(2t+\frac{1}{2})b}^{(2t+\frac{3}{2})b} dX \int_{\mathbb{R}} dx \; e^{-\frac{\eta}{1-\eta}(x-X)^2} \big|{}_q\langle x|\bar{0}_\Delta\rangle\big|^2 \tag{3.41} $$

To begin, we approximate the modulus squared of the wavefunction eq. (3.2′) as:

$$ \big|{}_q\langle x|\bar{0}_\Delta\rangle\big|^2 \approx \mathcal{N}_0^2 \sum_{s\in\mathbb{Z}} e^{-(2s\alpha_1)^2\tanh(\Delta^2)} e^{-\coth(\Delta^2)[x-2s\alpha_1\operatorname{sech}(\Delta^2)]^2} \tag{3.42} $$

which assumes the cross-terms originating from squaring the sum over s in eq. (3.2′)

are negligible. Next, we must estimate $\mathcal{N}_0$, which we can do as follows:

$$\begin{aligned} 1 &= \int_{\mathbb{R}} dx \left|{}_q\langle x|\bar{0}_\Delta\rangle\right|^2 \\ &\approx \mathcal{N}_0^2 \sum_{s\in\mathbb{Z}} e^{-(2s\alpha_1)^2\tanh(\Delta^2)} \sqrt{\pi\tanh(\Delta^2)} \end{aligned} \tag{3.43}$$

$$\Rightarrow\ \mathcal{N}_0^2 \sum_{s\in\mathbb{Z}} e^{-(2s\alpha_1)^2\tanh(\Delta^2)} \approx \sqrt{\coth(\Delta^2)/\pi} \tag{3.44}$$

One could approximate the sum on the left-hand-side of eq. (3.44) to directly estimate $\mathcal{N}_0$, but we will find that the form given in eq. (3.44) is sufficient for our calculations.

Continuing the calculation from eq. (3.41), we substitute eq. (3.42) and perform the integral over x:

$$P(1|0) \approx \mathcal{N}_0^2 \mathcal{N}'_{\eta,\Delta} \sum_{s\in\mathbb{Z}} e^{-(2s\alpha_1)^2\tanh(\Delta^2)} \left(\sum_{t\in\mathbb{Z}} \int_{(2t+\frac{1}{2})b}^{(2t+\frac{3}{2})b} dX\, e^{-\sigma_{\eta,\Delta}^{-1}\left(X-2s\alpha_1\mathrm{sech}(\Delta^2)\right)^2} \right) \tag{3.45}$$

where we've temporarily defined

$$\mathcal{N}'_{\eta,\Delta} = \left(1 + \frac{1-\eta}{\eta}\coth(\Delta^2)\right)^{-1/2} \tag{3.46a}$$

$$\sigma_{\eta,\Delta} = \tanh(\Delta^2) + \frac{1-\eta}{\eta} \tag{3.46b}$$

for readability. Now, the sum over t and the integral over X in eq. (3.45) (which we label I_s for simplicity) can be evaluated most simply by first performing the lowest-order expansions of the non-linear functions of Δ and setting $b = \alpha_1$:

$$I_s \approx \sum_{t\in\mathbb{Z}} \int_{(2t+\frac{1}{2})\alpha_1}^{(2t+\frac{3}{2})\alpha_1} dX\, e^{-\sigma_{\eta,\Delta}^{-1}\left(X-2s\alpha_1\right)^2} \tag{3.47}$$

Next, we perform the substitutions $X \mapsto X' = X - 2s\alpha_1$ and $t \mapsto t' = t - s$:

$$I_s \approx \sum_{t'\in\mathbb{Z}} \int_{(2t'+\frac{1}{2})\alpha_1}^{(2t'+\frac{3}{2})\alpha_1} dX'\, e^{-X'^2/\left(\Delta^2+(1-\eta)/\eta\right)} \tag{3.48}$$

which no longer depends on s. Finally, we can estimate this integral by simplifying

the domain of integration (which is again valid so long as Δ is small):

$$I_s \approx 2\int_{\alpha_1/2}^{\infty} dX'\, e^{-X'^2/\left(\Delta^2+(1-\eta)/\eta\right)} \tag{3.49}$$

$$= \sqrt{\pi\Big(\Delta^2+\frac{1-\eta}{\eta}\Big)}\,\mathrm{erfc}\Big(\frac{1}{2}\alpha_1\Big(\Delta^2+\frac{1-\eta}{\eta}\Big)^{-1/2}\Big) \tag{3.50}$$

Now, one can substitute this back into eq. (3.45) and use eq. (3.44) to yield the final result:

$$M_{\text{error}} \approx \mathrm{erfc}\Big(\frac{1}{2}\alpha_1\Big(\Delta^2+\frac{1-\eta}{\eta}\Big)^{-1/2}\Big) \tag{3.12}$$

3.C Derivation of POVM of the Continuous Measurement Scheme

In this section, we provide the details of the derivations of the POVM eq. (3.14). We consider a GKP mode (described by ladder operators $a, a^\dagger$ and position/momentum q_1, p_1) coupled via a Hamiltonian $H = -gq_1p_2$ to an ancilla readout mode (described by $b, b^\dagger, q_2, p_2$). The ancilla mode is initialised in an arbitrary state ρ_b, which in the simplest case is simply the vacuum state $|0\rangle\langle 0|$. The ancilla readout mode undergoes homodyne detection at a rate κ and efficiency η, and data is collected for a total measurement time T. We assume that the GKP mode experiences loss at a rate $\gamma \ll 1/T$ which we consider negligible for the purposes of the following discussion. This is justified by the fact that typical T_1 times for microwave resonators are two to three orders of magnitude larger than measurement times that we are interested in – roughly 1 μs – to be comparable to measurement times currently achieved in transmon devices.

The ultimate goal of this appendix is to derive the POVM of the above measurement scheme in terms of the detected photocurrent record $\mathbf{I}(0,T) = \{I(t)\}_{t\in(0,T)}$ from time 0 to time T. We follow the methods of Ref. [64], which we summarise here before presenting the details of the calculations. We begin by writing down the stochastic master equation (SME) that governs the dynamics of the normalised density matrix $\rho_c(t)$ conditioned on the detected photocurrent record $\mathbf{I}(0,t)$. Since the evolution of $\rho_c(t)$ depends on the measurement current, the SME will contain terms that depend on the Wiener increment $dW(t) = I(t)dt$ of the observed photocurrent. However, as $\rho_c(t)$ is normalised, the corresponding SME is necessarily non-linear in order for it to be trace-preserving. As such we consider the corresponding linearised SME which describes the dynamics of the un-normalised density matrix $\bar{\rho}_c(t)$. In this

picture, the probability of obtaining a given measurement outcome $\mathbf{I}(0,T)$ is given by $\mathrm{Tr}\big(\bar{\rho}_c(T)\big)p_{\mathrm{ost}}\big(\mathbf{I}(0,T)\big)$, where p_{ost} is the *ostensible distribution* of the observed photocurrent record (see Ref. [68], Chapter 4 of Ref. [2], or Appendix A of Ref. [64] for details). We can then solve the linearised SME by vectorising the density matrix $\bar{\rho}_c(t) \mapsto |\bar{\rho}_c(t)\rangle$ and using Lie algebraic techniques to simplify the resultant expression. We find that the state $\bar{\rho}_c(t)$ does not in fact depend on the *entire* photocurrent record $\mathbf{I}(0,t)$, but only on two integrals of the photocurrent labelled R and S. The probability of a given measurement outcome R, S can then be determined from the trace of $\bar{\rho}_c(T)$ and the ostensible distribution of R and S, which can in turn be used to construct the POVM elements $W_{R,S}$. Finally, we convert this two-mode POVM into a single-mode POVM by tracing out the ancilla mode given its initial state ρ_b is known.

Proceeding with the derivation, we begin by writing down the stochastic master equation (SME) of the system:

$$d\rho_c(t) = -i\big[H, \rho_c(t)\big]\,dt + \kappa\,\mathcal{D}[b]\rho_c(t)\,dt + \sqrt{\kappa\eta}\,\mathcal{H}[b]\rho_c(t)\,dW(t) \tag{3.51}$$

where $H = -gq_1p_2$ is the system Hamiltonian, $\mathcal{D}[b]\rho = b\rho b^\dagger - \frac{1}{2}\big\{b^\dagger b, \rho\big\}$ is the Lindblad dissipator, $\mathcal{H}[b]\rho = b\rho + \rho b^\dagger - \mathrm{Tr}(b\rho + \rho b^\dagger)$ describes the homodyne detection, and $dW(t)$ is the Wiener increment which is related to the relevant observed photocurrent $I(t)$ via $dW(t) = I(t)dt$. We use subscript c to emphasise that $\rho_c(t)$ is the density operator conditioned on the record of the observed photocurrent $\mathbf{I}(0,t)$ from time 0 to t.

Equation (3.51) is defined such that $\mathrm{Tr}\big(\rho_c(t)\big) = 1$ for all t; however, this normalisation comes at the cost of the non-linearity in $\mathcal{H}[b]$. Furthermore, to calculate the measurement statistics, one would need to sample from the actual distribution of the photocurrent record in any simulation of eq. (3.51). One way to partially solve this problem is to instead consider the master equation for $\tilde{\rho}_c(t)$, which is the non-normalised density operator with norm $\mathrm{Tr}\big(\tilde{\rho}_c(t)\big) = p\big(\mathbf{I}(0,t)\big|\rho(0)\big)$. This way, calculating measurement statistics simply involves calculating the trace of the evolved density operator, with the measurement outcomes sampled uniformly. However, this still results in a non-linear SME, and sampling from $\mathbf{I}(0,t)$ uniformly results in the vast majority of states having negligibly small norm. Instead, we can use an ostensible distribution for $\mathbf{I}(0,t)$ to remove the non-linearity in eq. (3.51) while still resulting in states with a finite norm. In this picture, we track the evolution of a non-normalised density operator $\bar{\rho}_c(t)$, which is defined such that the probability of obtaining a measurement record $\mathbf{I}(0,t)$ is $p\big(\mathbf{I}(0,t)\big|\rho(0)\big) = \mathrm{Tr}\big(\bar{\rho}_c(t)\big)p_{\mathrm{ost}}\big(\mathbf{I}(0,t)\big)$, where we call $p_{\mathrm{ost}}\big(\mathbf{I}(0,t)\big)$ the ostensible distribution of $\mathbf{I}(0,t)$. This way, information

about the measurement statistics is contained both in the norm of the state, and in the ostensible distribution. If we choose the photocurrent to be distributed such that $dW(t)$ is a mean-zero Gaussian distribution with variance dt, the SME becomes:

$$d\bar{\rho}_c(t) = -i\big[-gq_1p_2, \bar{\rho}_c(t)\big]dt + \mathcal{D}[\sqrt{\kappa}b]\bar{\rho}_c(t)dt + \sqrt{\kappa\eta}\big(b\bar{\rho}_c(t) + \bar{\rho}_c(t)b^\dagger\big)dW(t) \quad (3.52)$$

This way, we have both removed the non-linearity in eq. (3.51), and now to reproduce the measurement statistics of the system average we only need to sample $I(t)$ from the Gaussian ostensible distribution, since the additional contribution from the state of the system is now taken into account by the norm of $\bar{\rho}_c(t)$. Furthermore, we can the properties of $dW(t)$ given that $I(t)$ is Gaussian distributed, i.e. $dW(t)^2 = dt$ (see Ref. [69] for more details).

In order to solve this linear SME, we vectorise the density operator by "stacking" its rows. For an operator $\rho \in \mathcal{L}(\mathcal{H})$ acting on a Hilbert space $\mathcal{H}$ (which in our case contains two modes), we define the corresponding vectorised operator $|\rho\rangle \in \mathcal{H} \otimes \mathcal{H}^*$, where the resultant vector space now has two modes for each mode in $\mathcal{H}$, which we refer to as the physical mode (with lowering operators a, b, etc.) and the fictitious mode ($\tilde{a}, \tilde{b}$, etc.). For simplicity, we first consider $\mathcal{H}$ consisting of just a single bosonic mode. In particular, for an operator $\rho = \sum_{m,n\in\mathbb{Z}} \rho_{m,n} |m\rangle\langle n| \in \mathcal{L}(\mathcal{H})$, we define the corresponding vectorised operator as $|\rho\rangle = \sum_{m,n\in\mathbb{Z}} \rho_{m,n} |m\rangle|n\rangle$ in the Fock basis. Alternatively, $|\rho\rangle = (\rho\otimes I)\,|0\rangle_\beta$, where $|0\rangle_\beta = \sum_{n\in\mathbb{Z}} |n\rangle|n\rangle = e^{a^\dagger\tilde{a}^\dagger}\,|0\rangle|0\rangle$ is the thermo-entangled ground state between the two modes. $|0\rangle_\beta$ is the simultaneous 0-eigenstate of $\hat{\beta} = a - \tilde{a}^\dagger$ and $\hat{\beta}^\dagger = a^\dagger - \tilde{a}$. For an operator $A = \sum_n \alpha_n a^n + \beta_n a^{\dagger n}$ written in terms of ladder operators, we define the corresponding tilde'd operator as $\tilde{A} = \sum_n \alpha_n^* \tilde{a}^n + \beta_n^* \tilde{a}^{\dagger n}$. With this definition, it can be shown that $(A\otimes I)\,|0\rangle_\beta = (I\otimes\tilde{A}^\dagger)\,|0\rangle_\beta$. These properties generalise straightforwardly to when $\mathcal{H}$ consists of two (or more) modes, and we write each mode in the order $a, \tilde{a}, b, \tilde{b}$.

To vectorise the linear SME eq. (3.52) we rearrange $d\,|\bar{\rho}_c(t)\rangle = (d\bar{\rho}_c(t)\otimes I)\,|0\rangle_\beta$ by replacing all the operators acting on the right of ρ with operators acting on the fictitious modes, giving:

$$d|\bar{\rho}_c(t)\rangle = \Big(-i(H - \tilde{H})\,dt + \kappa\,D[b]\,dt + \sqrt{\kappa\eta}\,(b + \tilde{b})\,dW(t)\Big)\,|\bar{\rho}_c(t)\rangle \quad (3.53)$$

where $D[b] = b\tilde{b} - \frac{1}{2}(b^\dagger b + \tilde{b}^\dagger\tilde{b})$. To first order in dt this is equivalent to:

$$\begin{aligned}|\bar{\rho}_c(t+dt)\rangle = &\exp\Big(\sqrt{\kappa\eta}\,(b + \tilde{b})\,dW(t)\Big)\\ &\times \exp\Big(-i(H - \tilde{H})\,dt + \kappa\,D[b]\,dt - \frac{1}{2}\kappa\eta\,(b + \tilde{b})^2\,dt\Big)\,|\bar{\rho}_c(t)\rangle \quad (3.54)\end{aligned}$$

where we use $dW(t)^2 = dt$ in the exponential Taylor expansion. Equation (3.54) demonstrates the advantage of writing the SME in a vectorised form: now, the evolution operator is an exponential that is at most quadratic in the raising and lowering operators. In particular, the set of terms that are constant, linear or quadratic in $a, a^\dagger, \tilde{a}, \tilde{a}^\dagger, b, b^\dagger, \tilde{b}, \tilde{b}^\dagger$ generate a closed Lie algebra $\mathfrak{l}$, allowing us to utilise Lie algebraic techniques to rearrange the expression. Writing the linear stochastic terms as $dL(t) = \sqrt{\kappa\eta}\,(b + \tilde{b})\,dW(t)$ and the quadratic terms as $Q = -i(H - \tilde{H}) + \kappa\, D[b] - \frac{1}{2}\kappa\eta\,(b + \tilde{b})^2$ for simplicity, we can formally write the solution to eq. (3.54) as:

$$|\bar{\rho}_c(t)\rangle = \lim_{\delta t \to 0} \prod_{j=1}^{J} \exp\big(dL(j\delta t)\big) \exp\big(Q\delta t\big)\, |\rho(0)\rangle \tag{3.55}$$

where $t = J\delta t$. The first step to simplify this expression is to commute each linear term $dL(j\delta t)$ to the right of all the quadratic terms Q, modifying the linear term in the process. To commute the jth linear term past the j quadratic terms to its right, we must calculate $dL'(t)$ such that $\exp\big(dL(j\delta t)\big) \exp\big(Qj\delta t\big) = \exp\big(Qj\delta t\big) \exp\big(dL'(j\delta t)\big)$. Using the commutation formula $e^A B e^{-A} = \sum_{n=0}^{\infty} \frac{1}{n!} \mathcal{C}_A^n[B]$, where $\mathcal{C}_A^0[B] = B$ and $\mathcal{C}_A^n[B] = [A, \mathcal{C}_A^{n-1}[B]]$ for $n \geq 1$, we find that:

$$dL'(j\delta t) = e^{-Qj\delta t} dL(j\delta t) e^{Qj\delta t} \tag{3.56}$$

$$= \left(e^{-\kappa j\delta t/2} \sqrt{\kappa\eta}(b + \tilde{b}) - g\sqrt{\frac{2\eta}{\kappa}}(1 - e^{-\kappa j\delta t/2})(q_1 + \tilde{q}_1) \right) dW(j\delta t) \tag{3.57}$$

Since $[dL'(t), dL'(t')] = 0$ for all t, t', we can combine all the linear terms into one exponential, which becomes an integral in the $\delta t \to 0$ limit:

$$|\bar{\rho}_{R,S}(t)\rangle = \exp\big(Qt\big) \exp\big(L_{R,S}(t)\big)\, |\rho(0)\rangle \tag{3.58a}$$

$$L_{R,S}(t) = \sqrt{\eta} R(b + \tilde{b}) - \frac{g\sqrt{2\eta}}{\kappa} S(q_1 + \tilde{q}_1) \tag{3.58b}$$

$$R = \sqrt{\kappa} \int_0^t dt'\, e^{-\kappa t'/2}\, I(t') \tag{3.58c}$$

$$S = \sqrt{\kappa} \int_0^t dt'\, (1 - e^{-\kappa t'/2})\, I(t') \tag{3.58d}$$

Note that we now write subscripts R, S to indicate that the state $\bar{\rho}_{R,S}$ depends not on the *entire* measurement record $\mathbf{I}(0, t)$, but only on the two integrals of the photocurrent R and S.

Now, we wish to find the POVM operator $W_{R,S}$ such that $p\big(R, S\big|\rho(0)\big) = \mathrm{Tr}\big(W_{R,S}\rho(0)\big) = \big({}_\beta\langle 0|_\beta\langle 0|\big)\, |\bar{\rho}_{R,S}(t)\rangle\, p_{\mathrm{ost}}(R, S)$. We will do this initially by instead

evaluating $\Omega_{R,S}(x,x',\beta) = {}_q\langle x|\,{}_c\langle\beta|\,W_{R,S}\,|x'\rangle_q\,|\beta\rangle_c/\pi$, which can be thought of as a mixed representation of $W_{R,S}$ in the position basis of the GKP mode a and the Husimi Q-function representation in the ancilla mode b. From $\Omega_{R,S}$ one can reconstruct $W_{R,S}$ via the equation:

$$W_{R,S} = \int_{\mathbb{R}} dx \int_{\mathbb{R}} dx' \int_{\mathbb{C}} d^2\beta \left(\Omega_{R,S}(x,x',\beta)\,|x\rangle_q\,{}_q\langle x'| \otimes \tilde{D}^{(+1)}(\beta)\right) \tag{3.59}$$

where

$$\tilde{D}^{(+1)}(\beta) = \int_{\mathbb{C}} \frac{d^2\alpha}{\pi} e^{\alpha b^\dagger} e^{-\alpha^* b} e^{\beta\alpha^* - \beta^*\alpha} \tag{3.60}$$

is the Fourier transform of the normal-ordered displacement operator, which reconstructs a given operator from its Q-function representation.

To calculate $\Omega_{R,S}$ we begin by evaluating the inner product $\mathrm{Tr}\big(W_{R,S}\,|x\rangle_q\,|\beta\rangle_c\,{}_q\langle x'|\,{}_c\langle\beta|\,\big) = {}_\beta\langle 0|_\beta\langle 0|e^{Qt}e^{L_{R,S}(t)}\,|x\rangle_q|x'\rangle_q|\beta\rangle_c|\beta^*\rangle_c$. Noting that ${}_\beta\langle 0| = \langle 0|\langle 0|\,e^{a\tilde{a}}$, our strategy to evaluate this inner product is to partially normal order operators acting on the b and $\tilde{b}$ modes such to remove all $b^\dagger$ to the left by annihilating them on the vacuum state. The remaining exponents written in terms of $q_1, \tilde{q}_1, b, \tilde{b}$ act trivially on the initial state $|\rho(0)\rangle = |x\rangle_q|x'\rangle_q|\beta\rangle_c|\beta^*\rangle_c$ and thus the inner product can be evaluated. To do this, we make use of a faithful 10-dimensional matrix representation of the Lie algebra $\mathfrak{l}$ given in Table 1 of Ref. [64]. In particular, it can be shown that:

$$\begin{aligned} e^{a\tilde{a}+b\tilde{b}}e^{Qt} = {} & e^{a\tilde{a}}e^{c_1(b^\dagger b+\tilde{b}^\dagger\tilde{b})}e^{c_2(q_1b^\dagger+\tilde{q}_1\tilde{b}^\dagger)}e^{b\tilde{b}} \\ & \times \exp\Big(c_3(q_1b+\tilde{q}_1\tilde{b}) + c_4(\tilde{q}_1b+q_1\tilde{b}) \\ & \qquad + c_5(q_1^2+\tilde{q}_1^2) + c_6q_1\tilde{q}_1 + c_7(b+\tilde{b})^2\Big) \end{aligned} \tag{3.61}$$

where:

$$c_1 = -\kappa t/2 \tag{3.62a}$$
$$c_2 = (\sqrt{2}g/\kappa)(1-e^{\kappa t/2}) \tag{3.62b}$$
$$c_3 = (\sqrt{2}g/\kappa)(1-e^{-\kappa t/2})(1+\eta-\eta e^{-\kappa t/2}) \tag{3.62c}$$
$$c_4 = (\sqrt{2}g/\kappa)(1-e^{-\kappa t/2})(\eta-1-\eta e^{-\kappa t/2}) \tag{3.62d}$$
$$c_5 = (g^2/\kappa^2)\big(2(1-e^{-\kappa t/2}) - \kappa t - \eta\kappa\tau\big) \tag{3.62e}$$
$$c_6 = (2g^2/\kappa^2)\big(-2(1-e^{-\kappa t/2}) + \kappa t - \eta\kappa\tau\big) \tag{3.62f}$$
$$c_7 = (\eta/2)(e^{-\kappa t}-1) \tag{3.62g}$$

and $\tau = t - \frac{1}{\kappa}(1 - e^{-\kappa t/2})(3 - e^{-\kappa t/2})$. We can now evaluate the following:

$$\langle 0|\langle 0|\langle 0|\langle 0| e^{a\tilde{a}} e^{c_1(b^\dagger b + \tilde{b}^\dagger \tilde{b})} e^{c_2(q_1 b^\dagger + \tilde{q}_1 \tilde{b}^\dagger)} e^{b\tilde{b}} = {}_\beta\langle 0|_\beta \langle 0| \tag{3.63}$$

$${}_\beta\langle 0|_\beta\langle 0| |x\rangle_q |x'\rangle_q |\beta\rangle_c |\beta^*\rangle_c = \delta(x - x') \tag{3.64}$$

$${}_\beta\langle 0|_\beta\langle 0| e^{Qt} e^{L_{R,S}(t)} |x\rangle_q |x'\rangle_q |\beta\rangle_c |\beta^*\rangle_c = \delta(x - x') e^{f(x,x,\beta,\beta^*)} \tag{3.65}$$

where $f(q, \tilde{q}, b, \tilde{b})$ is equal to the sum of the right exponent in eq. (3.61) and $L_{R,S}(t)$. This in turn can be evaluated to give:

$$\begin{aligned} f(x, x, \beta, \beta^*) = &-\frac{4g^2\eta\tau}{\kappa} x^2 + \frac{4\sqrt{2}g\eta}{\kappa}\big(1 - e^{-\kappa t/2}\big)^2 x \mathrm{Re}(\beta) \\ &- 2\eta\big(1 - e^{-\kappa t}\big)\mathrm{Re}(\beta)^2 + 2\sqrt{\eta} R \mathrm{Re}(\beta) - \frac{2\sqrt{2}g\sqrt{\eta}}{\kappa} S x \end{aligned} \tag{3.66}$$

Now that we have calculated the contribution to $\Omega_{R,S}$ from the norm of $\bar{\rho}_{R,S}(t)$, we must now determine the ostensible distribution $p_{\mathrm{ost}}(R, S)$, given that $dW(t)$ is a Wiener increment, i.e. independent Gaussian random variables with mean 0 and variance dt. Since this is a Gaussian distribution (as R and S are integrals of Gaussian distributed random variables), we only need the mean and covariances to determine the distribution. Using $E[dW(t)] = 0$ we have that $E[R] = E[S] = 0$, and from $E[dW(t)dW(t')] = \delta(t - t')dtdt'$, we obtain:

$$\begin{aligned} E[R^2] &= \kappa \int_0^t \int_0^t e^{-\kappa(t' + t'')/2} E[dW(t')dW(t'')] \\ &= \kappa \int_0^t dt'\, e^{-\kappa t'} = 1 - e^{-\kappa t} \end{aligned} \tag{3.67}$$

And similarly:

$$E[S^2] = \kappa \int_0^t dt'\, (1 - e^{-\kappa t'/2})^2 = \kappa\tau \tag{3.68}$$

$$E[RS] = \kappa \int_0^t dt'\, (e^{-\kappa t'/2} - e^{-\kappa t'}) = (1 - e^{-\kappa t/2})^2 \tag{3.69}$$

Defining the covariance matrix

$$\Sigma = \begin{bmatrix} E[R^2] & E[RS] \\ E[RS] & E[S^2] \end{bmatrix} \tag{3.70}$$

we thus obtain the ostensible distribution:

$$p_{\text{ost}}(R,S) = \frac{1}{2\pi\sqrt{\det\Sigma}}\exp\left(-\frac{1}{2}\big(R^2\Sigma_{11}^{-1} + 2RS\Sigma_{12}^{-1} + S^2\Sigma_{22}^{-1}\big)\right) \tag{3.71}$$

$$= \frac{1}{2\pi\sqrt{\sigma}}\exp\left(-\frac{1}{2}\Big(\frac{\kappa\tau R^2}{\sigma} + \frac{2RS}{4-\kappa t\coth(\kappa t/4)} + \frac{S^2}{\kappa t - 4\tanh(\kappa t/4)}\Big)\right) \tag{3.72}$$

where we define $\sigma = \det\Sigma = \kappa t(1-e^{-\kappa t}) - 4(1-e^{-\kappa t/2})^2$ for convenience.

Using eqs. (3.59), (3.65), (3.66) and (3.72), we can now write the POVM elements as:

$$W_{R,S} = p_{\text{ost}}(R,S)\int_{\mathbb{R}} dx \int_{\mathbb{C}} \frac{d^2\beta}{\pi}\Big(e^{f(x,x,\beta,\beta^*)}\,|x\rangle_{q\,q}\langle x| \otimes \tilde{D}^{(+1)}(\beta)\Big) \tag{3.73}$$

Next, we wish to turn this two-mode POVM into a single-mode POVM by tracing out the ancilla mode (initialised in the state ρ_b) via the equation $V_{R,S,\rho_b} = \text{Tr}_b\big(W_{R,S}\rho_b\big)$. Given in the position representation $\rho_b(y,y') = {}_q\langle y|\,\rho_b\,|y'\rangle_q$, we can formally write

$$\text{Tr}\big(\tilde{D}^{(+1)}(\beta)\rho_b\big) = \int_{\mathbb{R}} dx \int_{\mathbb{C}} \frac{d^2\alpha}{\pi}\Big(\rho_b(x, x-\sqrt{2}\alpha_R)e^{|\alpha|^2/2 - i\alpha_R\alpha_I + \beta\alpha^* - \alpha\beta^* + i\sqrt{2}\alpha_I x}\Big) \tag{3.74}$$

where $\alpha_R = \text{Re}(\alpha)$ and $\alpha_I = \text{Im}(\alpha)$. Note however that we cannot practically evaluate the integral eq. (3.74) on its own due to the divergent $e^{|\alpha|^2/2}$ term. However, substituting into eq. (3.73) allows us to perform the resulting integrals and arrives us at our desired result:

$$V_{S,\rho_b} = \int_{\mathbb{R}} dR\ \text{Tr}_b\big(W_{R,S}\rho_b\big) \tag{3.75}$$

$$= \mathcal{N}\int dx\ |x\rangle_{q_1}\langle x| \int d\tilde{x}\ {}_{q_2}\langle\tilde{x}|\rho_b|\tilde{x}\rangle_{q_2}$$

$$\times \exp\left\{-\frac{1}{c}\Big[X - x - \frac{1}{2g\tau}\big(1-e^{-\kappa t/2}\big)^2\tilde{x}\Big]^2\right\} \tag{3.14}$$

where

$$\mathcal{N} = 1/\sqrt{2\pi\big(\kappa\tau - \eta(1-e^{-\kappa t/2})^4\big)} \tag{3.76a}$$

$$X = -\frac{1}{2\sqrt{2}g\tau\sqrt{\eta}}S \tag{3.76b}$$

$$c = \frac{\kappa\tau - \eta\big(1-e^{-\kappa t/2}\big)^4}{4g^2\tau^2\eta} \tag{3.76c}$$

Chapter 4

Stabiliser subsystem decompositions for single- and multi-mode Gottesman-Kitaev-Preskill codes

Abstract:

The Gottesman-Kitaev-Preskill (GKP) error correcting code encodes a finite-dimensional logical space in one or more bosonic modes, and has recently been demonstrated in trapped ions and superconducting microwave cavities. In this work, we introduce a new subsystem decomposition for GKP codes, analogous to the usual approach to quantum stabiliser codes. The decomposition has the defining property that a partial trace over the non-logical stabiliser subsystem is equivalent to ideal decoding of the logical state. Besides providing a convenient theoretical view on GKP codes, such a decomposition is also of practical use. We use the stabiliser subsystem decomposition to efficiently simulate GKP codes subject to noise, and in contrast to more conventional Fock basis simulations, we are able to consider essentially arbitrarily large photon numbers for realistic noise channels such as loss and dephasing.

4.1 Introduction

Bosonic codes encode digital quantum information in continuous variable (CV) quantum systems, and are an alternative approach to quantum error correction that has received both theoretical [70, 71, 61] and experimental [26, 72, 73, 74, 75, 76] attention. The Gottesman-Kitaev-Preskill (GKP) code [23] is one of the most intensively studied encodings of this type, and has recently been realised in both trapped ions [29, 30] and superconducting microwave resonators [27, 28].

From a theoretical perspective, bosonic codes can be understood as defining a logical subspace $\mathcal{L}$ of the CV Hilbert space $\mathcal{H} = \mathcal{L} \oplus \mathcal{L}^*$, with the remaining infinite dimensional Hilbert space $\mathcal{L}^*$ providing the redundancy required for error correction.

However, in the case of GKP codes, the non-normalisability of the codewords [23] means that the GKP logical "subspace" is formally not in the CV Hilbert space. An alternative formulation, which can be applied to any error correcting code, is to consider a decomposition of the Hilbert space such that the logical information in the error correcting code forms a *subsystem* $\mathcal{H} = \mathcal{L} \otimes \mathcal{S}$ [77, 78]. In such a decomposition, the partial trace over the non-logical subsystem corresponds to a decoding map $\mathcal{H} \mapsto \mathcal{L}$. In Ref. [3], Pantaleoni *et al.* introduced the concept of a bosonic subsystem decomposition, and defined a subsystem decomposition for single-mode GKP codes based on defining a "modular" position (or momentum) quadrature. This subsystem decomposition has been used in numerical studies of GKP codes [37, 39, 40, 41].

The subsystem decomposition is, however, not unique and there are good motivations to investigate alternatives. Specifically, the subsystem decomposition of Ref. [3] has lower symmetry than the GKP code itself: the logical subsystem differs if one chooses position or momentum as the "modular quadrature." In either case, the decomposition does not represent the logical information one would retrieve by performing noiseless decoding of the GKP code.

In this work, we introduce a subsystem decomposition that resolves these issues. In particular, this new decomposition has the desirable property that tracing over the non-logical subsystem $\mathcal{S}$ corresponds to a noiseless decoding map for the GKP code. We refer to this decomposition as the GKP stabiliser subsystem decomposition, as different stabiliser eigenstates correspond to orthogonal basis states of the non-logical subsystem $\mathcal{S}$.

The stabiliser subsystem decomposition can be applied to all multi-mode qubit or qudit GKP codes (including concatenation of GKP and binary stabiliser codes). For any GKP encoding, we show how to write an arbitrary CV state in the corresponding stabiliser subsystem decomposition in terms of a Gaussian unitary applied to a Zak basis [79]. Moreover, we show how to transform between subsystem decompositions – corresponding to different GKP codes – through three elementary transformations we call cell, Gaussian, and dimension transformations.

One practical challenge with GKP codes is the difficulty of numerically simulating GKP codes using a truncated Fock basis since both the mean and variance of the photon number distribution of physically realisable GKP codestates increases as the codestates approach the infinitely squeezed "ideal" codewords. Logical gates can also increase the photon number of the codestates, providing a further need to find new numerical methods to efficiently store and manipulate GKP states [80].

Using the stabiliser subsystem decomposition and the Zak basis for the CV stabiliser subsystem $\mathcal{S}$, we are able to study realistic noise channels such as loss and white-noise

dephasing for essentially arbitrary photon numbers. In the case of the square GKP code, our treatment is exact. We find that GKP codes are far more resilient against pure loss than against dephasing: a square single-mode code state with ten decibels of squeezing achieves an average gate infidelity below 10^{-3} for a loss rate up to $\sim$5%, while it can only tolerate a dephasing rate of $\sim$0.2% to reach the same fidelity. In the case of pure-dephasing, i.e. with white-noise dephasing as the only noise channel, there is a pseudo-threshold value for the GKP squeezing value and dephasing rate for the GKP code to "break even" in the following sense: the GKP code only performs better than a qubit defined using Fock states $|0\rangle$ and $|1\rangle$ given the GKP squeezing is above 10 dB *and simultaneously* the dephasing rate is below 0.08%. We also find that for both pure loss and dephasing, there is an optimal finite photon number that minimises the logical error rate. This optimal photon number is much larger for loss than for dephasing at the same rate.

Our results are organised as follows. Beginning in section 4.2, we provide an overview of the established formalism of multi-mode GKP lattices and set up the notation we will use in the remainder of the manuscript. In section 4.3, we define the stabiliser subsystem decomposition and show that the partial trace over the stabiliser subsystem corresponds to noiseless decoding. In section 4.4, we show how to transform the states of the stabiliser subsystem decomposition of one GKP code to any other code, and make explicit the connection between states written in the stabiliser subsystem decomposition and Zak states [79]. Finally, we show how to write many practical components of GKP codes conveniently in the stabiliser subsystem decomposition, namely logical Clifford gates (section 4.5), approximate GKP codewords, and noise channels such as pure loss, Gaussian displacements and white-noise dephasing (section 4.6). We provide concluding remarks in section 4.7.

4.2 Notation and Preliminaries

We start with a discrete variables system consisting of the tensor product of n finite-dimensional Hilbert spaces each with dimension $d_j, j = 1, \ldots, n$, which we write as a single vector $\mathbf{d} = (d_1, \ldots, d_n)$. We will allow for $d_j = 1$ for any of the dimensions, which we refer to as a "qunaught" since no logical information can be stored in the system. The Hilbert space of the total system, $\mathcal{H}_{\mathbf{d}} = \bigotimes_{j=1}^{n} \mathbb{C}^{d_j}$, is spanned by an orthonormal computational basis $|\mu\rangle$, where $\mu \in B_{\mathbf{d}} = \bigoplus_{j=1}^{n} \mathbb{Z}_{d_j}$ are dit-string labels. On each qudit, we define the Pauli $\hat{X}_{(d)}$ and $\hat{Z}_{(d)}$ operators that

have action

$$\hat{X}_{(d)}\,|a\rangle = \,|a+1 \pmod d\rangle\,, \tag{4.1a}$$

$$\hat{Z}_{(d)}\,|a\rangle = e^{2i\pi a/d}\,|a\rangle\,. \tag{4.1b}$$

where from now on we will not explicitly write (d) to indicate the dimension of the Pauli operators. We define a general Pauli operator acting on this discrete variables system as

$$\hat{P}_{\mathbf{d}}(\mathbf{s}) = \bigotimes_{j=1}^{n} \exp\left(\frac{i\pi}{d_j} s_j s_{j+n}\right) \hat{X}^{s_j} \hat{Z}^{s_{j+n}}, \tag{4.2}$$

where $\mathbf{s} \in \mathbb{Z}^{2n}$. Note that the Pauli operators of two vectors of integers $\hat{P}_{\mathbf{d}}(\mathbf{s})$ and $\hat{P}_{\mathbf{d}}(\mathbf{s}')$ may be identical even if $\mathbf{s} \neq \mathbf{s}'$.

Next, consider a continuous variables system consisting of n modes with Hilbert space $\mathcal{H}$. Such a system can be described by position and momentum operators which we write in a column vector $\hat{\boldsymbol{\xi}} = [\hat{q}_1 \ \cdots \ \hat{q}_n \ \hat{p}_1 \ \cdots \ \hat{p}_n]^T$, with canonical commutation relations $[\hat{q}_j, \hat{p}_{j'}] = i\delta_{jj'}$, or more compactly, $\hat{\boldsymbol{\xi}}^T \Omega\, \hat{\boldsymbol{\xi}} = i$, where

$$\Omega = \begin{bmatrix} 0_n & I_n \\ -I_n & 0_n \end{bmatrix} \tag{4.3}$$

defines the standard symplectic form $\omega(\mathbf{u}, \mathbf{v}) = \mathbf{u}^T \Omega \mathbf{v}$ in $\mathbb{R}^{2n}$.

We introduce the displacement operators

$$\hat{W}(\mathbf{v}) = \exp\big(\sqrt{2\pi} i\, \hat{\boldsymbol{\xi}}^T \Omega\, \mathbf{v}\big), \tag{4.4}$$

for $\mathbf{v} \in \mathbb{R}^{2n}$, which form an operator basis of $\mathcal{L}(\mathcal{H})$, the space of all linear operators acting on $\mathcal{H}$. The displacement operators obey the commutation relation $[\![\hat{W}(\mathbf{u}), \hat{W}(\mathbf{v})]\!] = e^{-2i\pi \mathbf{u}^T \Omega \mathbf{v}}$, where $[\![A, B]\!] = ABA^{-1}B^{-1}$ is the group commutator; and "displace" the position and momentum operators such that $\hat{W}(\mathbf{v})^\dagger \hat{\boldsymbol{\xi}} \hat{W}(\mathbf{v}) = \hat{\boldsymbol{\xi}} + \sqrt{2\pi}\mathbf{v}$, where on the left-hand-side the displacement operators are acting component-wise on the vector $\hat{\boldsymbol{\xi}}$. Note that eq. (4.4) differs by a factor of $\sqrt{\pi}$ from the more standard definition $\hat{D}(\alpha) = e^{\alpha \hat{a}^\dagger - \alpha^* \hat{a}}$ (in the single-mode case).

A Gaussian unitary operator $\hat{U}_S$ is parametrised by a $2n \times 2n$ real symplectic matrix S, satisfying $S^T \Omega S = \Omega$. The action of a Gaussian unitary on the position and momentum operators is a linear transformation: $\hat{U}_S^\dagger \hat{\boldsymbol{\xi}} \hat{U}_S = S\hat{\boldsymbol{\xi}}$. For any given S, $\hat{U}_S$ is the unitary metaplectic representation of S [81, 82]. Moreover, every Gaussian operator $\hat{U}_S$ can be written as the product of unitaries generated by Hamiltonians quadratic in $\hat{\boldsymbol{\xi}}$. We can alternatively interpret S as defining a canonically

transformed set of modes $\hat{\boldsymbol{\Xi}} = S^{-1}\hat{\boldsymbol{\xi}}$ such that we can write $\hat{U}_S \hat{W}(\mathbf{v})\hat{U}_S^\dagger = \hat{W}(S\mathbf{v}) = \exp(\sqrt{2\pi} i\, \hat{\boldsymbol{\Xi}}^T \Omega \mathbf{v})$.

4.2.1 Multi-mode GKP encodings

We can now introduce multi-mode qudit GKP codes, which encode a discrete variables system $\mathcal{H}_{\mathbf{d}}$ consisting of k qudits and $k-n$ qunaught states into an n-mode continuous variables system $\mathcal{H}$. To specify the GKP encoding we provide two pieces of information $(\Sigma, \mathbf{d})$, where Σ is a $2n \times 2n$ real symplectic matrix, and $\mathbf{d}$ a list of dimensions with positive integer elements and length n. Together, they specify a set of $2n$ mutually commuting stabiliser generators

$$\hat{S}_J = \hat{W}(\mathbf{m}_J), \quad \mathbf{m}_J = d^{1/2}_{J \,(\mathrm{mod}\, n)} (\Sigma)_J, \tag{4.5}$$

where $(\Sigma)_J$ is the J-th column of Σ. Note our index convention of using j when the index runs from 1 to n, and J when the index runs from 1 to $2n$. The codespace is defined by the simultaneous $+1$-eigenspace of the stabiliser generators. We define a set of logical Pauli operators

$$\bar{X}_j = \bar{P}_{\mathbf{d}}(\mathbf{e}_j) = \hat{W}(\mathbf{m}_j^\perp) \tag{4.6a}$$

$$\bar{Z}_j = \bar{P}_{\mathbf{d}}(\mathbf{e}_{j+n}) = \hat{W}(\mathbf{m}_{j+n}^\perp) \tag{4.6b}$$

where $\mathbf{m}_J^\perp = \mathbf{m}_J / d_{J \,(\mathrm{mod}\, n)}$ and $\mathbf{e}_J$ is the orthonormal basis vector of $\mathbb{R}^{2n}$ with a 1 in the J-th entry and 0's elsewhere.

As an illustrative example, consider the case of a single qudit with dimension d encoded in a single-mode square GKP code. The parameters for this encoding are $(\Sigma_{\mathrm{sq}}, d)$, with $\Sigma_{\mathrm{sq}} = I_2$, such that the stabilisers and logical Paulis are simply:

$$\hat{S}_1 = \hat{W}(\sqrt{d}\,\mathbf{e}_1) = e^{-i\sqrt{2\pi d}\hat{p}}, \tag{4.7a}$$

$$\hat{S}_2 = \hat{W}(\sqrt{d}\,\mathbf{e}_2) = e^{i\sqrt{2\pi d}\hat{q}}, \tag{4.7b}$$

$$\bar{X} = \bar{P}_d(\mathbf{e}_1) = \hat{W}(\mathbf{e}_1/\sqrt{d}) = e^{-i\sqrt{2\pi/d}\hat{p}}, \tag{4.7c}$$

$$\bar{Z} = \bar{P}_d(\mathbf{e}_2) = \hat{W}(\mathbf{e}_2/\sqrt{d}) = e^{i\sqrt{2\pi/d}\hat{q}} \tag{4.7d}$$

The encoding is called square because of the square lattice generated by $\mathbf{m_1} = \sqrt{d}\,\mathbf{e}_1$ and $\mathbf{m_2} = \sqrt{d}\,\mathbf{e}_2$.

One way to understand a general multi-mode GKP encoding defined by arbitrary $(\Sigma, \mathbf{d})$ is as follows: as Σ is a symplectic matrix, it defines a new set of canonically transformed modes, $\bar{\boldsymbol{\xi}} = \Sigma^{-1}\hat{\boldsymbol{\xi}}$, which we refer to as the *logical* modes. Written in

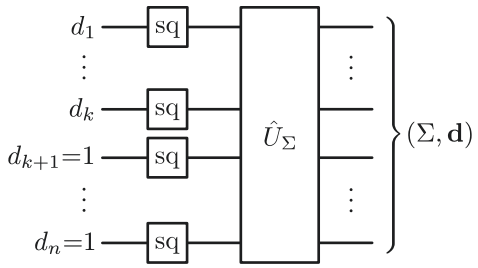

Figure 4.1: A circuit representing the encoding of k qudits into an ideal GKP code described by $(\Sigma, \mathbf{d})$. On the left-hand side, a state is described in the Hilbert space $\mathcal{H}_{\mathbf{d}}$ consisting of a tensor product k qudit and $k-n$ qunaught states. The qudits are encoded first into the ideal codespace of independent square GKP codes, followed by a Gaussian unitary operator $\hat{U}_\Sigma$.

terms of the logical modes, the stabiliser generators and logical Pauli operators take the form

$$\hat{S}_j = e^{-i\sqrt{2\pi d_j}\bar{p}_j}, \qquad \hat{S}_{j+n} = e^{i\sqrt{2\pi d_j}\bar{q}_j}, \tag{4.8a}$$

$$\bar{X}_j = e^{-i\sqrt{2\pi/d_j}\bar{p}_j}, \qquad \bar{Z}_j = e^{i\sqrt{2\pi/d_j}\bar{q}_j} \tag{4.8b}$$

which we recognise as n independent square-GKP encodings into the logical modes $\bar{q}_j, \bar{p}_j$. This picture is illustrated in fig. 4.1, where k qudits and $n-k$ qunaught states are first encoded into n square-lattice logical GKP modes, before applying a unitary Gaussian transformation $\hat{U}_\Sigma$.

To clarify the role of the qunaught states, note that the corresponding logical mode is constrained to a unique state that could equivalently be defined as the $+1$ eigenstate of $\bar{Z}$ for a square lattice GKP code with arbitrary d. The qunaught modes thus play precisely the same role as stabilisers in conventional qudit stabiliser codes [83], and can be used to provide additional protection against errors beyond that offered by single-mode GKP encodings. A special case is when $\hat{U}_\Sigma$ can be decomposed into a tensor product of single-mode Gaussian unitaries, followed by a logical Clifford circuit acting on the encoded GKP qudits. This case can be understood as a concatenation of n single-mode GKP codes with a conventional discrete qudit stabiliser code. For example, the GKP-surface code [44, 45] belongs to this class, where there are n physical modes, $n-1$ logical qunaught modes (corresponding to the surface code stabiliser generators), leaving a single logical GKP mode encoding a qubit with $d = 2$.

4.2.2 GKP lattices and primitive cell decoding

We define the GKP lattice generator matrix M, which has rows corresponding to the stabiliser generators $\mathbf{m}_J^T$, and generates the GKP lattice $\Lambda(M) = \left\{\boldsymbol{\ell} = \sum_{J=1}^{2n} s_J \mathbf{m}_J;\ \mathbf{s} \in \mathbb{Z}^{2n}\right\}$ [23, 84, 47, 48]. The symplectic dual lattice $\Lambda^\perp(M) = \left\{\boldsymbol{\ell}^\perp;\ \boldsymbol{\ell}^{\perp T}\Omega\boldsymbol{\ell} \in \mathbb{Z},\ \forall\, \boldsymbol{\ell} \in \Lambda(M)\right\}$ is generated by the vectors $\mathbf{m}_J^\perp$ and corresponds to the logical Pauli operators in eq. (4.6).

Conversely, if one starts with a lattice Λ which defines a GKP code, one can always find the corresponding pair $(\Sigma, \mathbf{d})$. In particular, for the lattice Λ to define a GKP code, it must be full-rank and symplectic, i.e. $\boldsymbol{\ell}^T\Omega\boldsymbol{\ell}' \in \mathbb{Z},\ \forall\ \boldsymbol{\ell}, \boldsymbol{\ell}' \in \Lambda$. There is freedom in the choice of matrix M that generates the lattice Λ; in particular, left-multiplication of any integral unimodular matrix N transforms $M \mapsto NM$ via row operations that preserve the lattice. This amounts to a different choice of stabiliser generators. Using Gaussian elimination, one can always find a generator M written in *standard form* such that $M^T = \Sigma D^{1/2}$ for a symplectic matrix Σ and diagonal matrix $D = \mathrm{diag}(\mathbf{d}, \mathbf{d})$, which provide the parameters in our original description $(\Sigma, \mathbf{d})$. We note that the standard form of M, and equivalently the choice of Σ, is not unique for a given lattice Λ. However, specifying Σ fixes the choice of stabiliser generators $\hat{S}_J$ and the labelling of each logical Pauli operator.

To perform an ideal round of GKP error correction, one first measures the eigenvalues of each stabiliser generator $\hat{W}(\mathbf{m}_J)$, giving a measurement outcome which we write as $e^{2i\pi\mathbf{v}^T\Omega\mathbf{m}_J)}$ for each stabiliser. Here, the choice of $\mathbf{v}$ is unique only up to the addition of vectors in the dual lattice. As such, to uniquely assign a translation that returns the state to the codespace, one must choose a *primitive cell* $\mathcal{P} \subset \mathbb{R}^{2n}$ of $\Lambda^\perp$, such that any vector $\mathbf{v} \in \mathbb{R}^{2n}$ can be written uniquely as $\mathbf{v} = \boldsymbol{\ell}^\perp + \{\mathbf{v}\}_\mathcal{P}$ for $\boldsymbol{\ell}^\perp \in \mathcal{L}^\perp$ and $\{\mathbf{v}\}_\mathcal{P} \in \mathcal{P}$. The choice of $\mathcal{P}$ defines a decoder, in which we pick the unique vector $\mathbf{v} = \{\mathbf{v}\}_\mathcal{P} \in \mathcal{P}$ that reproduces the syndrome $e^{2i\pi\mathbf{v}^T\Omega\mathbf{m}_J)}$ and perform the correction $\hat{W}(-\mathbf{v})$, returning the state to the ideal codespace.

The canonical choice of $\mathcal{P}$ is given by the Voronoi cell $\mathcal{V}(\Lambda^\perp)$ of the dual lattice, which contains the set of points closer to the origin than any other point in the lattice. However, other choices of $\mathcal{P}$ can be made, for example, to account for the effects of logical Clifford gates (section 4.5) and maximum likelihood decoding [47]. A further generalisation can be made to consider a codespace that is defined by the simultaneous $e^{2i\pi\mathbf{v}'^T\Omega\mathbf{m}_J)}$-eigenspace of the stabiliser generators. In this case, the canonical choice of primitive cell is given by the Voronoi cell of the lattice shifted to be centred around the codespace $\mathcal{V}(\Lambda^\perp) + \mathbf{v}'$, and the corresponding ideal decoder would apply a correction $\hat{W}(\mathbf{v}' - \mathbf{v})$. In the special case that each eigenvalue is ± 1, this reduces to a choice of "gauge" discussed in Ref. [48].

4.3 Stabiliser Subsystem Decomposition

In this section, we describe how to construct the stabiliser subsystem decomposition for arbitrary multi-mode GKP codes. We do this by first defining the stabiliser states which are labelled by two variables μ (a ditstring label) and $\mathbf{k} \in \mathbb{R}^{2n}$, where the former encodes logical information and the latter the stabiliser values. The stabiliser states span the Hilbert space $\mathcal{H}$, but are not linearly independent. Instead, we limit the range of $\mathbf{k}$ to a subset of $\mathbb{R}^{2n}$ such that the resulting states form a linearly independent basis, allowing us to define a subsystem decomposition between the logical and stabiliser labels. We then outline two key features of the decomposition: the decomposition of stabilisers and logical Paulis as products of operators on each subsystem, and the correspondence of the partial trace to a noiseless primitive cell decoding of the GKP code. Finally, we provide a visualisation of the decomposition and discuss some of its properties in the context of single-mode square and hexagonal GKP codes.

4.3.1 Stabiliser States

For any pair of parameters $(\Sigma, \mathbf{d})$, we begin by defining the state $|0_n, \mathbf{0}_{2n}\rangle_{(\Sigma,\mathbf{d})}$ as the simultaneous $+1$-eigenstate of the stabiliser group and each logical Pauli $\hat{Z}$ operator $\bar{Z}_j = \bar{P}_{\mathbf{d}}(\mathbf{e}_{j+n})$ for $j = 1, \ldots, n$, which corresponds to the ideal GKP state encoding the codeword $|0\rangle^{\otimes n}$. Next, we define the remaining ideal GKP codewords:

$$|\mu, \mathbf{0}_{2n}\rangle_{(\Sigma,\mathbf{d})} = \bigotimes_{j=1}^{n} \bar{X}_j^{\mu_j} |0_n, \mathbf{0}_{2n}\rangle_{(\Sigma,\mathbf{d})} \tag{4.9}$$

where $\mu \in B_{\mathbf{d}}$ is a ditstring. Here, $\bigotimes_{j=1}^{n} \bar{X}_j^{\mu_j} = \bar{P}_{\mathbf{d}}(\mu \oplus 0_n)$ corresponds to a product of Pauli X operators such that $|\mu, \mathbf{0}_{2n}\rangle_{(\Sigma,\mathbf{d})}$ is the ideal codeword $|\bar{\mu}\rangle$. We define the stabiliser states as:

$$|\mu, \mathbf{k}\rangle_{(\Sigma,\mathbf{d})} = \hat{W}(\mathbf{k}) |\mu, \mathbf{0}_{2n}\rangle_{(\Sigma,\mathbf{d})} \tag{4.10}$$

where $\mathbf{k} \in \mathbb{R}^{2n}$. As such, the state $|\mu, \mathbf{k}\rangle_{(\Sigma,\mathbf{d})}$ can be viewed as an ideal GKP codeword $|\bar{\mu}\rangle$ that has incurred a translation error $\hat{W}(\mathbf{k})$. We will also notate

$$|\psi, \mathbf{k}\rangle_{(\Sigma,\mathbf{d})} = \sum_{\mu \in B_{\mathbf{d}}} c_\mu |\mu, \mathbf{k}\rangle_{(\Sigma,\mathbf{d})} \tag{4.11}$$

where $|\psi\rangle = \sum_\mu c_\mu |\mu\rangle$.

We call the states defined in eq. (4.10) "stabiliser" states due to the property:

$$\hat{S}_J \left|\psi, \mathbf{k}\right\rangle_{(\Sigma,\mathbf{d})} = e^{2i\pi \mathbf{k}^T \Omega \mathbf{m}_J} \left|\psi, \mathbf{k}\right\rangle_{(\Sigma,\mathbf{d})}, \tag{4.12}$$

where we recall that $\hat{S}_J = \hat{W}(\mathbf{m}_J)$ is a displacement by $\mathbf{m}_J$, eq. (4.5). In other words, the subspace spanned by $\{|\mu, \mathbf{k}\rangle_{(\Sigma,\mathbf{d})}\}_{\mu \in B_\mathbf{d}}$ corresponds to the simultaneous eigenspace of the stabilisers with eigenvalues $e^{2i\pi\omega(\mathbf{v},\mathbf{m}_J)}$ respectively. The stabiliser states span the Hilbert space but are overcomplete, obeying the quasi-periodic boundary conditions:

$$\left|\mu, \mathbf{k} + \mathbf{m}_j^\perp\right\rangle_{(\Sigma,\mathbf{d})} = e^{i\pi\, \mathbf{k}^T \Omega \mathbf{m}_j^\perp} \left|\mu + \mathbf{e}_j, \mathbf{k}\right\rangle_{(\Sigma,\mathbf{d})} \tag{4.13a}$$

$$\left|\mu, \mathbf{k} + \mathbf{m}_{j+n}^\perp\right\rangle_{(\Sigma,\mathbf{d})} = e^{i\pi\, \mathbf{k}^T \Omega \mathbf{m}_{j+n}^\perp}\, e^{2i\pi\mu_j/d_j} \left|\mu, \mathbf{k}\right\rangle_{(\Sigma,\mathbf{d})} \tag{4.13b}$$

where the addition $\mu + \mathbf{e}_j$ is taken mod d_j. Intuitively, Equation (4.13a) can be thought of as applying a logical $\hat{X}$ to the qudit label μ and a $\mathbf{k}$-dependent phase to the stabiliser label upon application of the boundary condition, while Equation (4.13b) can be thought of as applying a logical $\hat{Z}$ to μ along with a phase, c.f. eq. (4.1).

4.3.2 The Subsystem Decomposition

To obtain a non-overcomplete basis of $\mathcal{H}$ from the stabiliser states, we must restrict $\mathbf{k}$ to a primitive cell $\mathcal{P}$ of $\Lambda^\perp$, such that the set of states $|\mu, \mathbf{k}\rangle_{(\Sigma,\mathbf{d})}$ for $\mathbf{k} \in \mathcal{P}$ forms a (linearly independent) basis of $\mathcal{H}$. We shall justify this claim in section 4.4.4 by making a direct connection to Zak states, which have already been shown to form a basis over a given primitive cell [79]. From this we construct a subsystem decomposition:

$$\mathcal{H} = \mathcal{L} \otimes_\mathcal{G} \mathcal{S} \tag{4.14}$$

where $\mathcal{G} = (\Sigma, \mathbf{d}, \mathcal{P})$ represents the three parameters required to specify the decomposition. We define the tensor product such that:

$$\left|\psi\right\rangle_\mathcal{L} \otimes_\mathcal{G} \left|\mathbf{k}\right\rangle_\mathcal{S} = \left|\psi, \mathbf{k}\right\rangle_{(\Sigma,\mathbf{d})} \tag{4.15}$$

for $\mathbf{k} \in \mathcal{P}$. We may also define the stabiliser states eq. (4.10) to be normalised such that:

$$\left(\sum_{\mu \in B_\mathbf{d}} |\mu\rangle\langle\mu|\right) \otimes_\mathcal{G} \left(\int_\mathcal{P} d^{2n}\mathbf{k}\, |\mathbf{k}\rangle\langle\mathbf{k}|\right) = I. \tag{4.16}$$

We call $\mathcal{L}$ the logical subsystem, which is isomorphic to the discrete-variables Hilbert space $\mathcal{H}_\mathbf{d}$. We call $\mathcal{S}$ the stabiliser subsystem, which is isomorphic to the full

Hilbert space $\mathcal{H}$ via the correspondence

$$|\mathbf{k}\rangle_{\mathcal{S}} \leftrightarrow |0_n, \Sigma D^{1/2}\Sigma^{-1}\mathbf{k}\rangle_{(\Sigma,\mathbf{1}_n)} \tag{4.17}$$

where $D = \mathrm{diag}(\mathbf{d}, \mathbf{d})$ and the right-hand-side is a stabiliser state of a qunaught GKP code with dimension vector given by $\mathbf{1}_n = (1, \ldots, 1)$. We will show in section 4.4.4 that the set of qunaught states $|0_n, \Sigma D^{1/2}\Sigma^{-1}\mathbf{k}\rangle_{(\Sigma,\mathbf{1}_n)}$ for $\mathbf{k} \in \mathcal{P}$ is related by a unitary transformation to a basis of Zak states for the full Hilbert space $\mathcal{H}$. In this sense, one can view the states $|\mathbf{k}\rangle_{\mathcal{S}}$ as a Zak basis for the stabiliser subsystem.

We can here make a connection to the stabiliser-destabiliser formalism of discrete-variables codes [85]. The set of *destabilisers* of the decomposition $\mathcal{G}$ is given by $\mathcal{D}_{\mathcal{G}} = \{\hat{W}(\mathbf{v})\}_{\mathbf{v}\in\mathcal{P}}$, which represent the lowest weight displacements that give rise to a given error syndrome. The full set of displacement operators $\{\hat{W}(\mathbf{v})\}_{\mathbf{v}\in\mathbb{R}^{2n}}$ is generated by the destabilisers $\mathcal{D}_{\mathcal{G}}$ and logical operators $\hat{W}(\mathbf{m}_J^{\perp})$ via the equation $\mathbf{v} = \boldsymbol{\ell}^{\perp} + \{\mathbf{v}\}_{\mathcal{P}}$, which in turn forms an operator basis for $\mathcal{L}(\mathcal{H})$, the space of linear operators that take $\mathcal{H} \to \mathcal{H}$. This is analogous to how qubit stabilisers, destabilisers and logical operators generate the Pauli group, which in turn forms an operator basis for the n-qubit Hilbert space.

4.3.3 Stabilisers and Logical Paulis

Next, we define a vector of mutually commuting operators $\hat{\mathbf{k}} = (\hat{k}_1, \ldots, \hat{k}_{2n})$ such that

$$\hat{k}_J\big(|\psi\rangle \otimes_{\mathcal{G}} |\mathbf{k}\rangle\big) = k_J\big(|\psi\rangle \otimes_{\mathcal{G}} |\mathbf{k}\rangle\big), \tag{4.18}$$

where $\mathbf{k} = (k_1, \ldots, k_{2n})$. These operators can be interpreted as modular quadrature operators $\hat{\mathbf{k}} = \{\hat{\boldsymbol{\xi}}/\sqrt{2\pi}\}_{\mathcal{P}}$, where $\{\mathbf{v}\}_{\mathcal{P}}$ is the remainder of $\mathbf{v}$ in the primitive cell $\mathcal{P}$ such that $\mathbf{v} = \ell^{\perp} + \{\mathbf{v}\}_{\mathcal{P}}$ for some $\ell^{\perp} \in \Lambda^{\perp}$. Using this, we can decompose the stabiliser generators as:

$$\hat{S}_J = \hat{W}(\mathbf{m}_J) = \hat{I} \otimes_{\mathcal{G}} e^{2i\pi\,\hat{\mathbf{k}}^T\Omega\mathbf{m}_J}. \tag{4.19}$$

Physically, the eigenvalues of $\hat{\mathbf{k}}$, which lie within the primitive cell $\mathcal{P}$, can be measured simply by measuring each stabiliser generator, which has eigenvalues $e^{2i\pi\,\omega(\hat{\mathbf{k}},\mathbf{m}_J)}$. For logical Pauli operators, we have:

$$\bar{X}_j = \hat{W}(\mathbf{m}_j^{\perp}) = \hat{X}_j \otimes_{\mathcal{G}} e^{2i\pi\,\hat{\mathbf{k}}^T\Omega\mathbf{m}_j^{\perp}} \tag{4.20a}$$

$$\bar{Z}_j = \hat{W}(\mathbf{m}_{j+n}^{\perp}) = \hat{Z}_j \otimes_{\mathcal{G}} e^{2i\pi\,\hat{\mathbf{k}}^T\Omega\mathbf{m}_{j+n}^{\perp}} \tag{4.20b}$$

where $\bar{P}_{\mathbf{d}}(\mathbf{e}_J)$ represents the logical Pauli operator acting on $\mathcal{H}$, while $P_{\mathbf{d}}(\mathbf{e}_J)$ represents the finite-dimensional Pauli operator acting on $\mathcal{L}$. The fact that $\bar{P}_{\mathbf{d}}(\mathbf{e}_J)$ can be written as a tensor product of operators acting on $\mathcal{L}$ and $\mathcal{S}$ ensures that it perfectly applies the logical gate $P_{\mathbf{d}}(\mathbf{e}_J)$ to the logical information of any given state. The phase acting on $\mathcal{S}$ does not affect the logical information nor the probability distribution of stabiliser measurement outcomes. However, the presence of this phase ensures that the equation $\hat{W}(d_{J \,(\mathrm{mod}\, n)}\mathbf{m}_J^{\perp}) = \hat{W}(\mathbf{m}_J)$ is satisfied. Practically, this phase affects neither the information stored in the logical subsystem nor the stabiliser measurement statistics of any state.

4.3.4 The Partial Trace

The central feature of using the stabiliser subsystem decomposition is the property that the partial trace over the stabiliser subsystem $\mathrm{tr}_{\mathcal{S}}$ corresponds to an ideal decoding map over a primitive cell $\mathcal{P}$. This holds even when the codespace is defined by the simultaneous $e^{2i\pi\omega(\mathbf{v}',\mathbf{m}_J)}$-eigenspace of each stabiliser generator $\hat{W}(\mathbf{m}_J)$, for some vector $\mathbf{v}'$.

We consider an ideal decoding map composed of a round of ideal error correction over $\mathcal{P}$, followed by a read-out of the logical information in the resultant ideal codestate. The round of error correction over $\mathcal{P}$ can be described in the subsystem decomposition first by a measurement of $\hat{\mathbf{k}}$ with some measurement outcome $\mathbf{v} \in \mathcal{P}$. The action on the state is to project into the $\hat{\mathbf{k}} = \mathbf{v}$ eigenspace via the projection operator $|\mathbf{v}\rangle_{\mathcal{S}}\langle\mathbf{v}|$ (up to normalisation). To return the state to the $\hat{\mathbf{k}} = \mathbf{v}'$ eigenspace (the ideal codespace), we apply the translation $\hat{W}(\mathbf{v}' - \mathbf{v})$. Ideal read-out is then described by the map $\hat{\rho} \otimes_{\mathcal{G}} |\mathbf{v}'\rangle\langle\mathbf{v}'| \mapsto \hat{\rho}$. Considering these three steps together and averaging over the possible measurement outcomes results a quantum channel $\mathcal{H} \mapsto \mathcal{L}$ described by the Kraus operators $\left\{{}_{\mathcal{S}}\langle\mathbf{v}|\right\}_{\mathbf{v}\in\mathcal{P}}$, where ${}_{\mathcal{S}}\langle\mathbf{v}|$ is an operator which maps $\mathcal{H} \to \mathcal{L}$. Thus, the post-decoding state of an arbitrary density operator $\hat{\rho}$, unconditional on the stabiliser measurement outcomes, can be found by taking the partial trace over the $\mathcal{S}$ subsystem:

$$\int_{\mathcal{P}} d^{2n}\mathbf{k}\ {}_{\mathcal{S}}\langle\mathbf{k}|\hat{\rho}|\mathbf{k}\rangle_{\mathcal{S}} = \mathrm{tr}_{\mathcal{S}}(\hat{\rho}) \tag{4.21}$$

This partial trace can in some sense be considered an ideal measure of the logical information in a state $\hat{\rho}$.

In some contexts, it is also useful to consider the state $\hat{\rho}_{\mathbf{v}} \propto {}_{\mathcal{S}}\langle\mathbf{v}|\hat{\rho}|\mathbf{v}\rangle_{\mathcal{S}}$, which is the decoded state conditioned on the measurement outcome $\mathbf{v}$ (not averaged over them), because some stabiliser outcomes can result in decoded states $\hat{\rho}_{\mathbf{v}}$ with less noise than

others. Such considerations have previously been explored in qubit codes [86] and in GKP magic state distillation [32].

4.3.5 Visualising $\otimes_{\mathcal{G}}$

The subsystem decomposition can be visualised by a *primitive cell diagram*, which is a generalisation of the framework presented in Ref. [87]. In this picture, we write an arbitrary state $|\phi\rangle \in \mathcal{H}$ as:

$$|\phi\rangle = \int_{\mathcal{P}} d^{2n}\mathbf{k}\, c(\mathbf{k}) \, |\psi(\mathbf{k})\rangle \otimes_{\mathcal{G}} |\mathbf{k}\rangle \tag{4.22}$$

Thus we can imagine the state $|\phi\rangle$ represented by two functions $c(\mathbf{k}) \in \mathbb{C}$ and $|\psi(\mathbf{k})\rangle \in \mathcal{H}_{\mathbf{d}}$ of $\mathbf{k} \in \mathcal{P}$. If one considers a projective measurement of $\hat{\mathbf{k}}$, then the probability of obtaining an outcome $\mathbf{k}$ is given by $|c(\mathbf{k})|^2 d^{2n}\mathbf{k}$, and the post-measurement state is $|\psi(\mathbf{k})\rangle \otimes_{\mathcal{G}} |\mathbf{k}\rangle$.

This is easiest to depict for single-mode GKP qubit codes such as the square (sq) and hexagonal (hex) GKP codes, defined by $\mathcal{G} = (\Sigma, \mathbf{d}, \mathcal{P})$ with respective parameters:

$$\begin{aligned} \Sigma_{\text{sq}} &= I_2 \quad & \Sigma_{\text{hex}} &= \begin{bmatrix} 2^{1/2}\,3^{-1/4} & -2^{-1/2}\,3^{-1/4} \\ 0 & 2^{-1/2}\,3^{1/4} \end{bmatrix} \\ \mathbf{d}_{\text{sq}} &= (2) \quad & \mathbf{d}_{\text{hex}} &= (2) \\ \mathcal{P}_{\text{sq}} &= \mathcal{V}_{\text{sq}} \quad & \mathcal{P}_{\text{hex}} &= \mathcal{V}_{\text{hex}} \end{aligned} \tag{4.23}$$

where $\mathcal{V}_{\text{sq}} = \left(-\frac{1}{2\sqrt{2}}, \frac{1}{2\sqrt{2}}\right]^2$ and $\mathcal{V}_{\text{hex}}$, illustrated in fig. 4.2(a) and (d), are the Voronoi cells of the respective lattices.

For illustrative purposes, consider a toy error model consisting of a random walk of translation errors applied to an ideal codestate of the square GKP code with logical information $|\psi\rangle_{\mathcal{L}}$. The logical information remains unchanged as long as the random walk keeps the state within $\mathcal{P}_{\text{sq}}$ without crossing a boundary. Once it crosses a boundary, applying Equations (4.13a) and (4.13b) causes a logical Pauli operator to be applied to the logical subsystem, corresponding to a logical error [fig. 4.2(a)].

For the square GKP code, one can consider each boundary of the primitive Voronoi cell as either an $\hat{X}$ or a $\hat{Z}$ boundary, depending on the logical error that it causes. The hexagonal code has $\hat{X}$, $\hat{Y}$ and $\hat{Z}$ boundaries due to the symmetry of each of these operators. Meanwhile, the single-mode square GKP qunaught code (defined by $\Sigma_{\emptyset,\text{sq}} = I_2$, $\mathbf{d}_{\emptyset,\text{sq}} = (1)$, $\mathcal{P}_{\emptyset,\text{sq}} = \sqrt{2}\mathcal{V}_{\text{sq}}$) has only logical identity $\hat{I}$ boundaries, and as such crossing a boundary only applies an overall phase to the state.

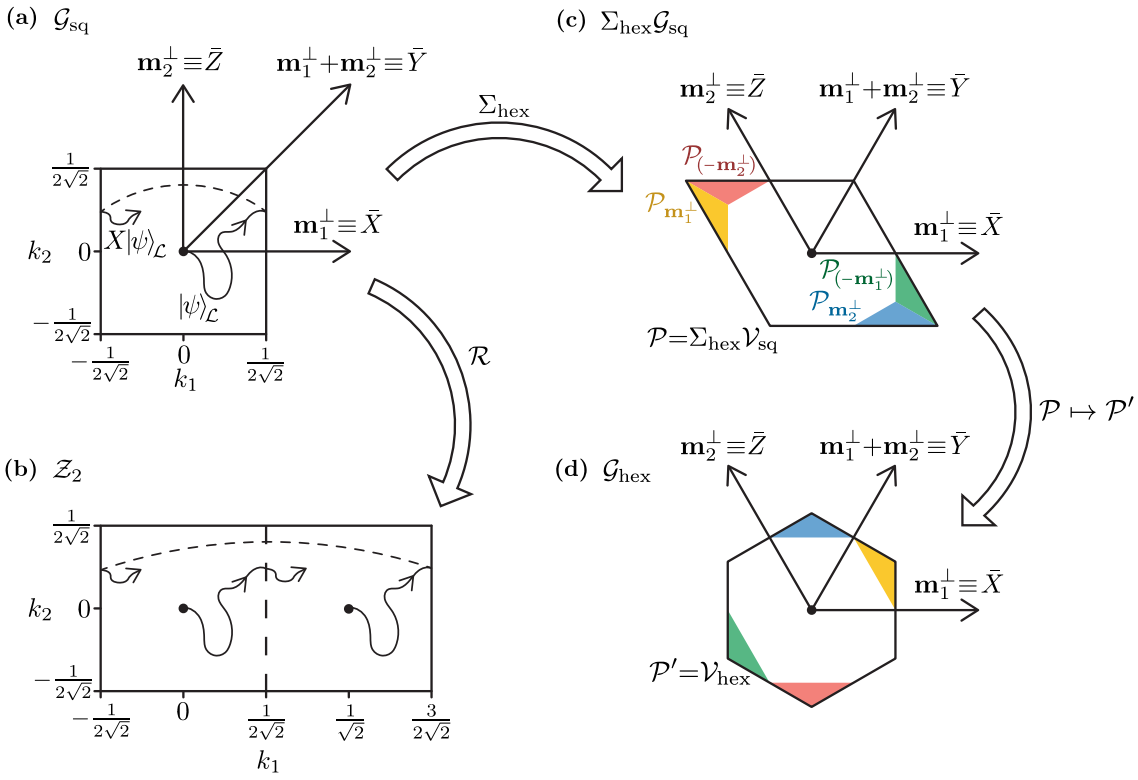

Figure 4.2: Primitive cell diagrams for (a) the single-mode square GKP qubit code $\mathcal{G}_{\text{sq}}$, (b) the Zak basis $\mathcal{Z}_2$, (c) the Gaussian transform of the square code $\Sigma_{\text{hex}}\mathcal{G}_{\text{sq}}$, and (d) the single-mode hexagonal GKP qubit code $\mathcal{G}_{\text{hex}}$. (a) and (b) are related by an unfolding operation $\mathcal{R}$; (a) and (c) are related by a Gaussian transformation Σ_{hex}; and (c) and (d) are related by a cell transformation $\mathcal{P} = \Sigma_{\text{hex}}\mathcal{V}_{\text{sq}} \mapsto \mathcal{P}' = \mathcal{V}_{\text{hex}}$, which is performed by shifting each of the coloured regions $\mathcal{P}_{\boldsymbol{\ell}^\perp}$ by the dual lattice vector $\boldsymbol{\ell}^\perp$ to form $\mathcal{P}'$. In subplots (a), (c) and (d), each point on the plot represents the two-dimensional simultaneous eigenspace of the GKP stabilisers; while in (b) each point represents a single state. Applying a random walk of translation operators to an ideal GKP codestate $|\psi\rangle_{\mathcal{L}} \otimes_{\mathcal{G}_{\text{sq}}} |\mathbf{0}\rangle_{\mathcal{S}}$ does not affect the logical subsystem until the state reaches one of the quasi-periodic boundaries of the cell; for example, causing an $\hat{X}$ error as shown in (a). The corresponding path is traced out twice in (b) since the codespace of the square GKP code consists of superpositions of states located at the points $(0, 0)$ and $(1/\sqrt{2}, 0)$.

4.4 Transformations of $\otimes_{\mathcal{G}}$

Now that we have introduced the stabiliser subsystem decomposition, and discussed some of the insights it can give us, we now turn our attention to transformations of the subsystem decomposition $\otimes_{\mathcal{G}}$. In particular, we are interested in describing operations that transform basis states from one subsystem decomposition to another: $|\psi\rangle \otimes_{\mathcal{G}} |\mathbf{k}\rangle \mapsto |\psi'\rangle \otimes_{\mathcal{G}'} |\mathbf{k}'\rangle$. To do so, we must describe three transformations – cell transformations, Gaussian transformations, and dimension transformations – that allow one to relate any two arbitrary stabiliser subsystem decompositions $\mathcal{G}$ and $\mathcal{G}'$ over n modes to each other. Then, we apply these transformations to relate an arbitrary state in an arbitrary subsystem decomposition to a Zak state [79], allowing us to decompose arbitrary CV states via their Zak representation. We will also make use of these transformations in our discussion of logical Clifford gates in section 4.5.

4.4.1 Cell transformations

The first transformation we consider is a primitive cell transformation $(\Sigma, \mathbf{d}, \mathcal{P}) \mapsto (\Sigma, \mathbf{d}, \mathcal{P}')$, which can be achieved simply by applying the boundary conditions eqs. (4.13a) and (4.13b) to regions of $\mathcal{P}$.

To understand how such a transformation can be achieved, consider the example shown in fig. 4.2(c) and (d), which shows the transformation of a rhombus primitive cell $\mathcal{P} = \Sigma_{\text{hex}}\mathcal{V}_{\text{sq}}$ to a hexagonal primitive cell $\mathcal{P}' = \mathcal{V}_{\text{hex}}$ (the hexagonal Voronoi cell). To perform this transformation, the initial rhombus cell $\mathcal{P}$ is split into disjoint regions $\mathcal{P}_{\boldsymbol{\ell}^\perp} \subset \mathcal{P}$, each of which will be shifted by the dual GKP hexagonal lattice vector $\boldsymbol{\ell}^\perp$ to form the new hexagonal cell. In particular, the yellow, green, blue and red regions are shifted by $+\mathbf{m}_1^\perp$, $-\mathbf{m}_1^\perp$, $+\mathbf{m}_2^\perp$ and $-\mathbf{m}_2^\perp$ respectively; while the unshaded region is "shifted" by the zero vector $\mathbf{0} \in \Lambda^\perp$. The basis vectors in each state are transformed by the boundary conditions eqs. (4.13a) and (4.13b), which apply a Pauli operator to the logical subsystem and a phase to the stabiliser subsystem. As we will see shortly, each region $\mathcal{P}_{\boldsymbol{\ell}^\perp}$ can be found explicitly by taking the intersection of $\mathcal{P}$ with $\mathcal{P}' - \boldsymbol{\ell}^\perp$ for each $\boldsymbol{\ell}^\perp \in \Lambda^\perp$.

We can formalise and generalise the cell transformation procedure to allow for the transformation of arbitrary primitive cells $\mathcal{P} \mapsto \mathcal{P}'$. We begin by partitioning $\mathcal{P}$ into a (finite) number of disjoint regions $\mathcal{P}_{\boldsymbol{\ell}^\perp} = \left(\mathcal{P}' - \boldsymbol{\ell}^\perp\right) \cap \mathcal{P}$, each of which are labelled by a dual lattice vector $\boldsymbol{\ell}^\perp \in \Lambda^\perp$. Since $\mathcal{P}$ and $\mathcal{P}'$ are primitive cells, we have that $\mathcal{P} = \bigcup_{\boldsymbol{\ell}^\perp \in \Lambda^\perp} \mathcal{P}_{\boldsymbol{\ell}^\perp}$ and $\mathcal{P}_{\boldsymbol{\ell}_1^\perp} \cap \mathcal{P}_{\boldsymbol{\ell}_2^\perp} = \emptyset$ whenever $\boldsymbol{\ell}_1^\perp \neq \boldsymbol{\ell}_2^\perp$. In other words, each vector $\mathbf{k} \in \mathcal{P}$ is contained within exactly one region $\mathcal{P}_{\boldsymbol{\ell}^\perp}$. Then, to construct the new primitive cell $\mathcal{P}'$, we shift each region $\mathcal{P}_{\boldsymbol{\ell}^\perp}$ by the vector $\boldsymbol{\ell}^\perp$, such that

$\mathcal{P}' = \bigcup_{\boldsymbol{\ell}^\perp \in \Lambda^\perp} \left(\mathcal{P}_{\boldsymbol{\ell}^\perp} + \boldsymbol{\ell}^\perp\right)$. With such a partitioning, we can specify the action of the cell transformation $\mathcal{P} \mapsto \mathcal{P}'$ by:

$$|\psi\rangle \otimes_{\mathcal{G}} |\mathbf{k}\rangle = \left(\hat{P}_{\mathbf{d}}(\mathbf{s})^\dagger |\psi\rangle\right) \otimes_{\mathcal{G}'} \left(e^{-i\pi\omega(\mathbf{k},\boldsymbol{\ell}^\perp)} |\mathbf{k} + \boldsymbol{\ell}^\perp\rangle\right) \tag{4.24}$$

for $\mathbf{k} \in \mathcal{P}_{\boldsymbol{\ell}^\perp}$, and $\boldsymbol{\ell}^\perp = M^{\perp T}\mathbf{s}$ for $\mathbf{s} \in \mathbb{Z}^{2n}$. As such, one can always transform the cell of the subsystem decomposition via the above procedure without altering either Σ or $\mathbf{d}$.

4.4.2 Gaussian transformations

Applying a Gaussian unitary operator $\hat{U}_S$ to each state in the subsystem decomposition transforms $\mathcal{G} \mapsto S(\mathcal{G}) = (S\Sigma, \mathbf{d}, S\mathcal{P})$ such that:

$$\hat{U}_S\left(|\psi\rangle \otimes_{\mathcal{G}} |\mathbf{k}\rangle\right) = |\psi\rangle \otimes_{S(\mathcal{G})} |S\mathbf{k}\rangle \tag{4.25}$$

Unlike the cell transformation, a Gaussian transformation affects both the generators of the lattice $\Sigma \mapsto S\Sigma$ and the cell $\mathcal{P} \mapsto S\mathcal{P}$. As such, to obtain a transformation that only alters the lattice generators in Σ, one must combine a Gaussian transformation with a cell transformation that restores the original cell $\mathcal{P}$. In the special case where $S = \Sigma N \Sigma^{-1}$ for some symplectic integral matrix N, then $\hat{U}_S$ is the Gaussian unitary operator which implements the logical Clifford gate with action $\bar{P}_{\mathbf{d}}(\mathbf{s}) \to \hat{U}_S \bar{P}_{\mathbf{d}}(\mathbf{s}) \hat{U}_S^\dagger = \bar{P}_{\mathbf{d}}(N\mathbf{s})$, which we discuss in more detail in section 4.5.

4.4.3 Dimension transformations

Next, we describe how to transform the dimension, $\mathbf{d}$, of the subsystem decomposition. Central to this description is the *unfolding* operation, which transforms the dimension of the j-th mode $d_j \mapsto 1$, creating a qunaught mode. The net effect of this operation is to take the logical information in the j-th mode and instead label it in the stabiliser mode, thus increasing the size of the primitive cell [see fig. 4.2(b)]. One can then consider a d-fold *folding* operation, consisting of the inverse of an unfolding operation, which turns a qunaught mode back into a qudit mode with dimension d_j. Thus, we can describe arbitrary dimension transformations $\mathbf{d} \mapsto \mathbf{d}'$ as unfolding and folding operations on each of the modes.

While there are many possible transformations that map a given mode from a qudit mode to a qunaught mode, we choose to describe the unfolding operation $\mathcal{R}_j$ (defined below) due to its simple action on the states in $\mathcal{G}$. Intuitively, the unfolding operation consists of demoting the logical $\bar{Z}_j = \hat{W}(\mathbf{m}^\perp_{j+n})$ operator on the j-th mode

to a stabiliser of the new code, and removing the corresponding $\bar{X}_j = \hat{W}(\mathbf{m}_j^\perp)$ from the logical Pauli group entirely.

We can see how this affects $\mathcal{G}$ by considering the generator matrix $M^T = \Sigma D^{1/2}$ that has $\mathbf{m}_j$ as its columns. The transformation of the generators $\mathbf{m}_{j+n} \mapsto \mathbf{m}_{j+n}/d_j$ can be described by the transformation $d_j \mapsto 1$ and $\Sigma \mapsto \Sigma A_j(\sqrt{d_j})$ (while the other columns are left untouched). Here, $A_j(\lambda)$ is a diagonal matrix with λ and λ^{-1} on the j-th and $(j+n)$-th positions respectively, and ones elsewhere.

Since $\hat{W}(\mathbf{m}_{j+n}^\perp)$ is now in the stabiliser group, states related by a translation $\hat{W}(\mathbf{m}_j^\perp)$ which previously had identical support in the stabiliser subsystem are now distinct. As a result, $\mathcal{R}_j$ "unfolds" (or copies) the cell $\mathcal{P}$ d_j times along the vector $\mathbf{m}_j^\perp$, with the new primitive cell given by $\mathcal{R}_j(\mathcal{P}) = \bigcup_{a=0}^{d_j-1}(\mathcal{P} + a\mathbf{m}_j^\perp)$. Correspondingly, the action of $\mathcal{R}_1$ (acting on the first mode for simplicity) on states in the subsystem decomposition is given by:

$$|\mu_1 \oplus \mu\rangle \otimes_{\mathcal{G}} |\mathbf{k}\rangle = e^{i\pi\omega(\mu\mathbf{m}_1^\perp, \mathbf{k})} |0 \oplus \mu\rangle \otimes_{\mathcal{R}_1(\mathcal{G})} |\mathbf{k} + \mu_1\mathbf{m}_1^\perp\rangle \tag{4.26}$$

where $\oplus$ represents the direct sum of two dit-strings $\mu_1 \in \mathbb{Z}_{d_1}$ and $\mu \in B_{(d_2,\ldots,d_n)}$, and $\mathbf{k} \in \mathcal{P}$. Note that the 0 label on the right-hand side is redundant since it represents a qunaught degree of freedom.

In order to transform a given mode from dimension $d_j \mapsto d_j'$, we can simply perform an unfolding operation $\mathcal{R}_j$ followed by a folding operation that results in a code with dimension d_j' on mode j. In section 4.4.4 we will also make use of the all-mode unfolding operation $\mathcal{R} = \mathcal{R}_1 \circ \cdots \circ \mathcal{R}_n$, which results in a trivial subsystem decomposition $\mathcal{R}(\mathcal{G})$ with dimension vector $\mathbf{1}_n$.

4.4.4 Zak states

We can now make explicit the connection between states in a stabiliser subsystem decomposition $\mathcal{G}$ and Zak states [79], which have received recent attention due to their applicability to GKP codes [87, 88]. We start with the single-mode case. The Zak basis [79] is a complete basis parametrised by two real numbers, k_1 and k_2, which can be written in the position basis

$$|k_1, k_2\rangle_{\mathcal{Z}_d} = \sqrt[4]{2\pi d}\, e^{i\pi k_1 k_2} \sum_{s\in\mathbb{Z}} e^{2i\pi\sqrt{d}k_2 s} \big|\sqrt{2\pi}(k_1 + \sqrt{d}s)\big\rangle_q \tag{4.27}$$

where $|x\rangle_q$ is the x-eigenstate of the position operator $\hat{q}$. Note that compared to Ref. [79] our definition here is different by a scaling factor.

The relationship between d in eq. (4.27) and the dimension of a GKP code will

become apparent later; for now, it simply serves as a scaling factor in the definition of the Zak states. Each state $|\mathbf{k}\rangle_{\mathcal{Z}_d}$ is a simultaneous eigenstate of $\hat{W}(\sqrt{d}\,\mathbf{e}_1)$ and $\hat{W}(\mathbf{e}_2/\sqrt{d})$. Moreover, $\{|\mathbf{k}\rangle_{\mathcal{Z}_d}\}_{\mathbf{k}\in\mathcal{P}_{\mathcal{Z}_d}}$ forms a orthonormal basis of $\mathcal{H}$ satisfying

$$ {}_{\mathcal{Z}_d}\langle k_1, k_2 | k_1', k_2' \rangle_{\mathcal{Z}_d} = \delta(k_1 - k_1')\delta(k_2 - k_2') \tag{4.28} $$

where

$$ \mathcal{P}_{\mathcal{Z}_d} = \left(-\frac{1}{2\sqrt{d}}, \frac{2d-1}{2\sqrt{d}}\right] \times \left(-\frac{1}{2\sqrt{d}}, \frac{1}{2\sqrt{d}}\right] \tag{4.29} $$

As such, one can view the Zak states as a trivial stabiliser subsystem decomposition with parameters $\mathcal{Z}_d = \big(\mathrm{diag}(d^{1/2}, d^{-1/2}), (1), \mathcal{P}_{\mathcal{Z}_d}\big)$, where we have labelled $|0\rangle \otimes_{\mathcal{Z}_d} |\mathbf{k}\rangle = |\mathbf{k}\rangle_{\mathcal{Z}_d}$ for simplicity. Finally, in order to generalise to n-modes, we can define the n-mode Zak states

$$ |\mathbf{k}\rangle_{\mathcal{Z}_{\mathbf{d}}} = |k_1, \ldots, k_{2n}\rangle_{\mathcal{Z}_{\mathbf{d}}} = \bigotimes_{j=1}^{n} |k_j, k_{j+n}\rangle_{\mathcal{Z}_{d_j}} \tag{4.30} $$

With this connection, we can now apply any of the transformations discussed above in sections 4.4.1 to 4.4.3 to relate arbitrary GKP codes to Zak states. Indeed, the states $|\mathbf{k}\rangle_{\mathcal{Z}_d}$ are the direct result of applying an unfolding operation to a single-mode square qudit GKP code. Starting from an arbitrary $\mathcal{G}$, we can apply three transformations to result in a decomposition described by $\mathcal{Z}_{\mathbf{d}}$: We first apply a Gaussian transformation $\hat{U}_\Sigma^\dagger$, followed by a cell transformation $\Sigma^{-1}\mathcal{P} \mapsto \left(\prod_{j=1}^{n}\left(-\frac{1}{2\sqrt{d_j}}, \frac{1}{2\sqrt{d_j}}\right]\right)^{\times 2}$. These two steps result in an n-mode square GKP qudit code with dimension $\mathbf{d}$. Finally, we apply an all-mode unfolding operation $\mathcal{R}$, which results in the decomposition $\mathcal{Z}_{\mathbf{d}}$.

We comment here that it was shown in Ref. [79] that the Zak states $|\mathbf{k}\rangle_{\mathcal{Z}_d}$ form a basis over the primitive cell $\mathcal{P}_{\mathcal{Z}_d}$. Moreover, the transformations described in sections 4.4.1 to 4.4.3 preserve the linear independence and completeness of the states in each subsystem decomposition. This can be used to justify in the general case that the restriction of $\mathbf{k}$ to a primitive cell $\mathcal{P}$ of the dual lattice $\Lambda^\perp$ does indeed result in a valid basis $\big\{|\mu, \mathbf{k}\rangle\big\}_{\mu\in B_{\mathbf{d}}, \mathbf{k}\in\mathcal{P}}$ of the full Hilbert space $\mathcal{H}$. As such this procedure provides a practical method of writing arbitrary states $|\phi\rangle \in \mathcal{H}$ in terms of an arbitrary stabiliser subsystem decomposition $\mathcal{G}$.

A specific example is given by the square single-mode GKP code, where we have:

$$ |\mu\rangle \otimes_{\mathcal{G}_{\mathrm{sq}}} |\mathbf{k}\rangle = e^{i\pi\mu k_2/\sqrt{2}} |\mathbf{k} + \mu\hat{\mathbf{i}}/\sqrt{2}\rangle_{\mathcal{Z}_2} \tag{4.31} $$

It is interesting to compare such a decomposition with the Zak-basis representation

of the modular-position subsystem decomposition [88]. Once a rescaling of $\mathbf{k}$ is taken into account, the only difference between the two decompositions is the k_2-dependent phase (see section 4.A). In the stabiliser subsystem decomposition, this phase is chosen specifically so that the logical $\hat{X}$ and $\hat{Z}$ operators act as a product of operators on the logical and stabiliser subsystems. In this sense, the stabiliser subsystem decomposition for the single-mode square GKP qubit code can be thought of as a "rephasing" of the modular-position subsystem decomposition that symmetrises the treatment of position and momentum.

A similar but distinct application of Zak states in the stabiliser subsystem decomposition formalism is in the basis of the stabiliser subsystem $\mathcal{S}$. Given a state $|\mathbf{k}\rangle_{\mathcal{S}}$, we defined in section 4.3.1 a correspondence

$$|\mathbf{k}\rangle_{\mathcal{S}} \leftrightarrow |0_n, \Sigma D^{1/2}\Sigma^{-1}\mathbf{k}\rangle_{(\Sigma,\mathbf{1}_n)} \tag{4.17}$$

between the stabiliser subsystem $\mathcal{S}$ and the full Hilbert space $\mathcal{H}$. The qunaught state on the right-hand-side can be written in terms of square Zak states

$$|0_n, \Sigma D^{1/2}\Sigma^{-1}\mathbf{k}\rangle_{(\Sigma,\mathbf{1}_n)} = \hat{U}_{\Sigma}^{\dagger}\,|D^{1/2}\Sigma^{-1}\mathbf{k}\rangle_{\mathcal{Z}_{\mathbf{1}_n}} \tag{4.32}$$

When $\mathbf{k}$ runs over a primitive cell $\mathcal{P}$ of the dual GKP lattice $\Lambda^{\perp}$, $D^{1/2}\Sigma^{-1}\mathbf{k}$ runs over a primitive cell of the square qunaught lattice generated by the vectors $\mathbf{e}_J$. As such, the set of qunaught states $|0_n, \Sigma D^{1/2}\Sigma^{-1}\mathbf{k}\rangle_{(\Sigma,\mathbf{1}_n)}$ form a basis of the full Hilbert space $\mathcal{H}$, and we can conclude that the stabiliser subsystem is isomorphic to the full Hilbert space (as a vector space).

4.5 Logical Clifford Gates

One appealing feature of GKP codes is that Gaussian operators, which are often easy to implement experimentally, are sufficient to generate the logical Clifford group. Here, to simplify our discussion we consider GKP codes that encode k qubits in n modes, i.e. $\mathbf{d} = \big(\underbrace{2,\ldots,2}_{k},\underbrace{1,\ldots,1}_{n-k}\big)$. Given a k-qubit Clifford operator $\hat{A}$, we can write $\hat{A}\hat{P}_{\mathbf{d}}(\mathbf{s})\hat{A}^{\dagger} = \hat{P}_{\mathbf{d}}\big(\tilde{A}\mathbf{s}\big)$, where $\tilde{A}$ is an integral symplectic matrix that acts trivially on the $d = 1$ modes. An equivalent definition of $\tilde{A}$ is that it is the symplectic representation of the Gaussian operator which applies a logical $\hat{A}$ gate on the square GKP k-qubit code. The logical implementation of such an operator when the qubits are encoded in an n-mode GKP code is then given by $\bar{A} = \hat{U}_{S_A}$ with $S_A = \Sigma\tilde{A}\Sigma^{-1}$ such that $\bar{A}\bar{P}_{\mathbf{d}}(\mathbf{s})\bar{A}^{\dagger} = \bar{P}_{\mathbf{d}}(\tilde{A}\mathbf{s})$.

Before discussing how $\bar{A}$ transforms states in the subsystem decomposition, we first consider its action only on the stabiliser states:

$$\bar{A}\,|\psi, \mathbf{k}\rangle_{(\Sigma,\mathbf{d})} = |A(\psi), S_A\mathbf{k}\rangle_{(\Sigma,\mathbf{d})} \tag{4.33}$$

where $A(\psi)$ is the qubit label representing the state $|A(\psi)\rangle = A\,|\psi\rangle$. From a practical perspective, this makes it clear how logical Clifford gates may cause errors to spread based on the action of S_A. However, one can exactly counteract the effects of such a spreading if one simply performs a round of error correction over a modified primitive cell $S_A\mathcal{P}$, an idea which was first introduced in Ref. [46] and generalised in chapter 3.

However, an interesting alternative viewpoint is to consider $\bar{A}$ as a Gaussian transformation of the stabiliser subsystem decomposition $\mathcal{G} \mapsto \bar{A}(\mathcal{G}) = (S_A\Sigma, \mathbf{d}, S_A\mathcal{P})$ via eq. (4.25). Here, the right multiplication of Σ by the integral symplectic matrix $\tilde{A}$ can be viewed as a column operation that relabels the generators of the GKP lattice and dual lattice, which is equivalent to a row operation N acting on the generator matrix M, while leaving the overall lattice invariant. The relabelling occurs in such a way that the Clifford operator A is applied to the logical Pauli group labels of $\Lambda^{\perp}$:

$$|\psi\rangle \otimes_{\bar{A}(\mathcal{G})} |\Sigma\tilde{A}\Sigma^{-1}\mathbf{k}\rangle = \big(A\,|\psi\rangle\big) \otimes_{(\Sigma,\mathbf{d},\Sigma\tilde{A}\Sigma^{-1}\mathcal{P})} |\Sigma\tilde{A}\Sigma^{-1}\mathbf{k}\rangle \tag{4.34}$$

However, to write the right-hand side of eq. (4.34) in terms of the original subsystem decomposition $\mathcal{G}$, one must perform a cell transformation $\Sigma\tilde{A}\Sigma^{-1}\mathcal{P} \mapsto \mathcal{P}$ via eq. (4.24), which in general applies Pauli operators to the logical information conditioned on the stabiliser subsystem. It is for this reason that we suggest that a solution is to first perform error correction on the code $(\Sigma, \mathbf{d}, \Sigma\tilde{A}\Sigma^{-1}\mathcal{P})$ to ensure one is in the ideal codespace, before performing the cell transformation back to $\mathcal{G}$, which acts trivially on the ideal codespace.

Our formalism also allows for other logical gate proposals such as those using gauge fixing techniques [89, 48], which could be used to implement code deformation and lattice surgery schemes in concatenated codes. The key element of lattice surgery we wish to describe here is the projection into the +1-eigenspace of an arbitrary logical Pauli operator P. We describe this in the stabiliser subsystem decomposition as demoting the logical Pauli operator to a stabiliser via an appropriate unfolding operation, followed by a round of error correction to project into the +1-eigenspace. This is described by the sequence $\mathcal{R}_1 \circ \bar{A}$, where A is a Clifford operator such that $APA^{\dagger} = Z \otimes I^{\otimes(n-1)}$. We leave it to future work to apply this formalism to practical problems involving GKP gauge fixing.

4.6 Efficient numerical modeling of noise

One appealing feature of subsystem decompositions is that the partial trace operation provides a straightforward method of extracting qubit-level information from CV states. Here, we apply the stabiliser subsystem decomposition partial trace to sources of imperfections in GKP codes, including approximate GKP codestates and noise sources such as loss and dephasing. To do this, we will consider an arbitrary noise channel $\mathcal{N}$, and provide a method of evaluating the corresponding logical noise channel $\mathcal{N}_\mathcal{L} = \mathrm{tr}_\mathcal{S}\big(\mathcal{N}(\hat{\rho} \otimes_\mathcal{G} |\mathbf{0}\rangle\langle\mathbf{0}|)\big)$, which represents the transformation of logical information due to the action of $\mathcal{N}$. From $\mathcal{N}_\mathcal{L}$, we can then extract various quantities of interest, such as the average gate fidelity.

We begin by writing the map $\mathcal{N}$ in terms of translation operators in its *chi*-representation:

$$\mathcal{N}(\hat{\rho}) = \iint_{\mathbb{R}^{2n}} d^{2n}\mathbf{u}\, d^{2n}\mathbf{v}\, c(\mathbf{u},\mathbf{v}) \hat{W}(\mathbf{u}) \hat{\rho} \hat{W}(\mathbf{v})^\dagger \tag{4.35}$$

where the coefficients can be evaluated from a Kraus decomposition $\{\hat{E}_j\}$ of $\mathcal{N}$: $c(\mathbf{u},\mathbf{v}) = \sum_j c_j(\mathbf{u}) c_j(\mathbf{v})^*,\ c_j(\mathbf{u}) = \mathrm{tr}(\hat{E}_j \hat{W}(\mathbf{u})^\dagger)$. Applying $\mathcal{N}$ to an ideal codestate is straightforward given this representation due to the action of translations on the stabiliser states [eq. (4.10)]. However since the integral in eq. (4.35) runs over $\mathcal{R}^{2n}$, one must first apply the boundary conditions [eqs. (4.13a) and (4.13b)] before taking the partial trace [eq. (4.21)]. Thus the resulting logical noise channel is given by:

$$\mathcal{N}_\mathcal{L}(\hat{\rho}) = \sum_{\mathbf{s},\mathbf{t} \in \mathbb{Z}^{2n}} \left(\int_\mathcal{P} d^{2n}\mathbf{v}\, c_{\mathbf{s},\mathbf{t}}(\mathbf{v},\mathbf{v}) \right) \hat{P}_\mathbf{d}(\mathbf{s}) \hat{\rho} \hat{P}_\mathbf{d}(\mathbf{t}) \tag{4.36}$$

where

$$c_{\mathbf{s},\mathbf{t}}(\mathbf{u},\mathbf{v}) = e^{i\pi(\omega(\mathbf{u}, M^{\perp T}\mathbf{s}) - \omega(\mathbf{v}, M^{\perp T}\mathbf{t}))} c(\mathbf{u} + M^{\perp T}\mathbf{s}, \mathbf{v} + M^{\perp T}\mathbf{t}) \tag{4.37}$$

and $M^{\perp T} = \Sigma D^{-1/2}$ has the generators $\mathbf{m}_i^\perp$ of $\Lambda^\perp$ as its columns.

When the modulus of the chi-representation $|c(\mathbf{u},\mathbf{v})|$ of $\mathcal{N}$ goes to 0 as $|\mathbf{u}|, |\mathbf{v}| \to \infty$, eq. (4.36) can be evaluated by truncating the sum over $\mathbf{s}, \mathbf{t}$ at some finite value. However, this property is not guaranteed, and as such there are some limitations to our method. In the following subsections, we apply our methods to four sources of noise: the envelope operator $e^{-\Delta^2 \hat{a}^\dagger \hat{a}}$, pure loss, random Gaussian displacements, and white-noise dephasing, each acting on a single-mode square GKP qubit code, and discuss our results.

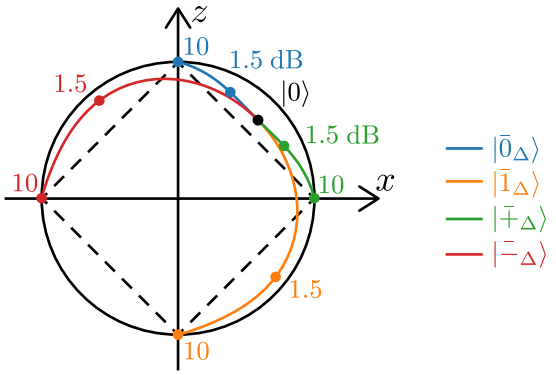

Figure 4.3: Decoded states $\mathrm{tr}_{\mathcal{S}}\left(|\bar{\psi}_{\Delta,\mathrm{sq}}\rangle\langle\bar{\psi}_{\Delta,\mathrm{sq}}|\right)$ for $|\psi\rangle = |0\rangle, |1\rangle, |+\rangle, |-\rangle$ in the single-mode square GKP code, as a function of Δ (labelled on the plot) projected onto the xz-plane of the Bloch sphere (solid outline). As $\Delta_{\mathrm{dB}} \to +\infty$ ($\Delta \to 0$), each state approaches the ideal logical $|\psi\rangle_{\mathcal{L}}$ state respectively; while as $\Delta_{\mathrm{dB}} \to -\infty$ ($\Delta \to +\infty$), each state approaches the vacuum state $|0\rangle$, which is outside the stabiliser octahedron (dotted line) and is thus distillable to a magic state [32].

4.6.1 Envelope Operator

We begin with the (non-unitary) envelope operator $e^{-\Delta^2 \hat{a}^\dagger \hat{a}}$, which we use to define the approximate GKP codestates and has been shown to be simply related to most other choices of approximate GKP codestates [54]. The envelope operator has chi-representation:

$$c_{\mathcal{E}}^{\Delta}(\mathbf{u}, \mathbf{v}) \propto e^{-\frac{\pi}{2}\coth\left(\frac{\Delta^2}{2}\right)(|\mathbf{u}|^2+|\mathbf{v}|^2)} \tag{4.38}$$

Δ is also commonly quoted in decibels as $\Delta_{\mathrm{dB}} = -10\log_{10}(\Delta^2)$. Importantly, since the chi-representation of the envelope operator decreases exponentially as $|\mathbf{u}|, |\mathbf{v}| \to \infty$, we can truncate the sum in eq. (4.36) to numerically obtain the logical channel $\mathcal{E}_{\mathcal{L}}^{\Delta}$. In this case, the logical channel is not trace-preserving due to the non-unitarity of the envelope operator. As such we must either normalise the states after applying $\mathcal{E}_{\mathcal{L}}^{\Delta}$, or apply an orthonormalisation procedure to obtain a CPTP map $\mathcal{E}_{\mathcal{L},\mathrm{o}}^{\Delta}$ (see section 4.B for details).

For the square GKP code, we evaluate each integral in eq. (4.36) analytically, and calculate the truncated sum over $\mathbf{s}, \mathbf{t}$ numerically. We plot the location on the Bloch sphere of the normalised states $\mathcal{E}_{\mathcal{L}}^{\Delta}(|\psi\rangle\langle\psi|)/N_{\Delta,\psi}$ for $|\psi\rangle = |0\rangle, |1\rangle, |+\rangle, |-\rangle$ in fig. 4.3, and the average gate fidelity of $\mathcal{E}_{\mathcal{L},\mathrm{o}}^{\Delta}$ with the identity $\hat{I}_2$ in the $\gamma = 0$ curve of fig. 4.4(a). Since $c_{\mathcal{E}}^{\Delta}(\mathbf{u}, \mathbf{v})$ becomes sharper around the origin as $\Delta \to 0$, this method becomes more accurate and less computationally expensive as the approximation becomes more ideal. Indeed, our technique easily enables the simulation of highly squeezed states, and there is no in principle limit to the squeezing one could simulate (see section 4.C).

4.6.2 Pure Loss

We define loss from the Kraus operators

$$\mathcal{L}^{\gamma}(\hat{\rho}) = \sum_{j=0}^{\infty} \hat{L}_j^{\gamma} \hat{\rho} \hat{L}_j^{\gamma\dagger} \qquad \hat{L}_j^{\gamma} = \Big(\frac{\gamma}{1-\gamma}\Big)^{j/2} \frac{\hat{a}^j}{\sqrt{j!}} (1-\gamma)^{\hat{n}/2} \tag{4.39}$$

with chi-representation [90]:

$$c_{\mathcal{L}}^{\gamma}(\mathbf{u},\mathbf{v}) \propto \big(e^{-\frac{\pi}{2}|\mathbf{u}-\mathbf{v}|^2} e^{i\pi\omega(\mathbf{u},\mathbf{v})}\big)^{(1+\sqrt{1-\gamma})^2/\gamma} \tag{4.40}$$

Here, $\gamma = 1 - e^{-\kappa t}$ represents the amount of loss applied to a system evolving under the master equation $\dot{\hat{\rho}} = \kappa \mathcal{D}[\hat{a}]\hat{\rho}$ for some time t, where $\mathcal{D}[\hat{a}]\hat{\rho} = \hat{a}\hat{\rho}\hat{a}^\dagger - \frac{1}{2}\{\hat{a}^\dagger\hat{a}, \hat{\rho}\}$ is the Lindblad jump operation and $\{A, B\} = AB + BA$ is the anticommutator. Simulating loss acting directly on ideal GKP codestates using our method is not possible since the modulus $|c_{\mathcal{L}}^{\gamma}(\mathbf{u},\mathbf{v})|$ is constant for $\mathbf{u} = \mathbf{v}$, even as $|\mathbf{u}| \to \infty$. Instead, we consider a composition of maps $\mathcal{L}^{\gamma} \circ \mathcal{E}^{\Delta}$, whose chi-representation can be determined from eqs. (4.38) and (4.40).

The average gate fidelity of the orthonormalised logical map $(\mathcal{L}^{\gamma} \circ \mathcal{E}^{\Delta})_{\mathcal{L},\mathrm{o}}$ is given in fig. 4.4, and is compared to the average gate fidelity of loss acting on the trivial encoding, $\{|0\rangle, |1\rangle\}$. We find that for any fixed loss rate γ, there is an optimal Δ that maximises the average gate fidelity. One can understand this intuitively by noting that loss has a larger effect on states with a large photon number. Since the average photon number of an approximate GKP codestate is given approximately by $\langle \hat{a}^\dagger \hat{a} \rangle \approx 1/(2\Delta^2) - 1/2$ [31], one can see that as the GKP codestate becomes more ideal, the noise due to $\mathcal{E}^{\Delta}$ decreases while the noise due to $\mathcal{L}^{\gamma}$ increases, giving rise to a cross-over point which represents the optimal Δ that protects against the loss.

Based on experimental parameters used in a recent paper which produced GKP eigenstates [28], we can provide an order-of-magnitude estimate of the loss rates we might expect in near-term GKP experiments. Given a cavity $T_{1,\mathrm{c}} \sim 500$ μs, and an error-correction cycle time of 1 μs (chapter 3), we can estimate the amount of loss we can expect during an error correction cycle to be on the order of $\gamma \sim 0.2\%$. At this loss rate, GKP codes perform very well, and the optimal level of squeezing is far beyond what is experimentally feasible. We note that in the context of a more complex experiment, such a low T_1 time may not be achievable, and the error-correction cycle may take longer than 1 μs. Nevertheless, the square GKP code still performs well below the break-even point even for much larger amounts of loss.

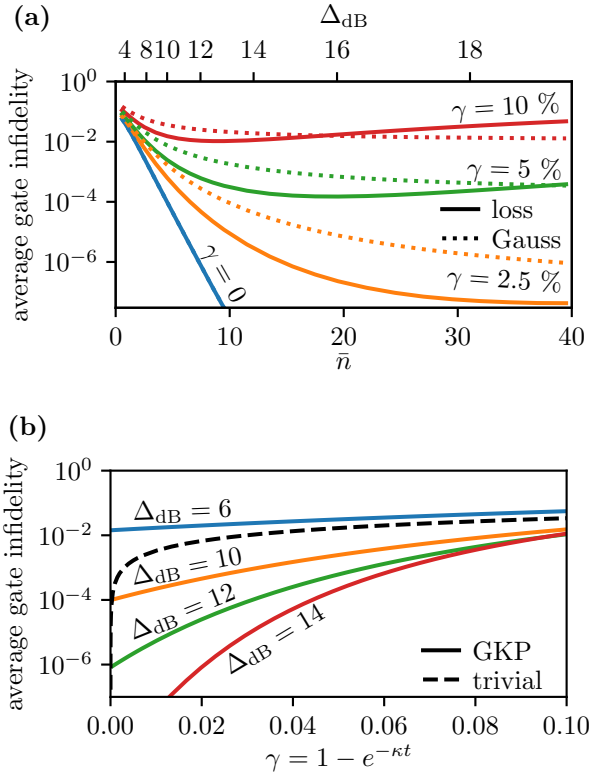

Figure 4.4: Average gate infidelities of the logical noise channels corresponding to loss acting on approximate single-mode square GKP qubit codestates $(\mathcal{L}^\gamma \circ \mathcal{E}^\Delta)_{\mathcal{L},\mathrm{o}}$, and random Gaussian displacements acting on approximate GKP codestates $(\mathcal{G}^\sigma \circ \mathcal{E}^\Delta)_{\mathcal{L},\mathrm{o}}$, where $\sigma^2 = \gamma/(1-\gamma)$ is scaled such that the Gaussian displacement channel is equivalent to loss followed by a quantum-limited amplification: $\mathcal{G}^\sigma = \mathcal{A}^{1/(1-\gamma)} \circ \mathcal{L}^\gamma$. (a) The average gate infidelities plotted as a function of Δ, where $\bar{n} \approx \frac{1}{2\Delta^2} - \frac{1}{2}$ is the average photon number of the approximate GKP encoded maximally mixed state. The $\gamma = 0$ curve represents the errors solely resulting from the approximation of the GKP codestates. (b) Average gate fidelity of the loss channel as a function of $\gamma \approx \kappa t$, which represents the loss of logical information as approximate GKP codestates evolve in time under loss. This is compared to the loss of logical information stored in a trivial Fock encoding under the same loss channel.

4.6.3 Gaussian Displacements

Finally, we compare this average gate fidelity to that of a Gaussian random displacement noise model $\mathcal{G}^\sigma$, which is defined by its chi-representation:

$$c^\sigma_{\mathcal{G}}(\mathbf{u},\mathbf{v}) = \sigma^{-2} e^{-\pi|\mathbf{u}|^2/\sigma^2}\delta^2(\mathbf{u}-\mathbf{v}) \tag{4.41}$$

Similarly to loss, $\sigma^2 = \kappa_G t$ represents the variance of random displacements applied to a system evolving under the master equation $\dot{\hat{\rho}} = \kappa_G\big(\mathcal{D}[\hat{a}] + \mathcal{D}[\hat{a}^\dagger]\big)\hat{\rho}$ for some time t. $\mathcal{G}^\sigma$ can be used as a noise model for GKP codes due to its simplicity, and since it is equivalent to a loss channel $\mathcal{L}^\gamma$ followed by a quantum-limited amplification channel $\mathcal{A}^{1/(1-\gamma)}$, where $\sigma^2 = \gamma/(1-\gamma)$ (see Ref. [38] for details). To compare $\mathcal{G}^\sigma = \mathcal{A}^{1/(1-\gamma)} \circ \mathcal{L}^\gamma$ to $\mathcal{L}^\gamma$, we again consider the average gate fidelity of the logical noise channel $(\mathcal{G}^\sigma \circ \mathcal{E}^\Delta)_{\mathcal{L},\mathrm{o}}$, the average gate fidelity of which is plotted in fig. 4.4(a). Unsurprisingly, for equivalent values of σ and γ, $\mathcal{G}^\sigma$ introduces more noise into the system than $\mathcal{L}^\gamma$ in the region of small Δ_{dB} and γ. However, the infidelity of $\mathcal{G}^\sigma$ overtakes that of $\mathcal{L}^\gamma$ at large values of Δ_{dB} or γ, reflecting the fact that $c^\sigma_{\mathcal{G}}(\mathbf{u},\mathbf{v})$ tends to 0 as $|\mathbf{u}|, |\mathbf{v}| \to \infty$ even when acting on ideal codestates. These results motivate our analysis of loss exactly as a more accurate representation of the errors that occur on a system housing a GKP qubit.

The consequences of our results for correcting GKP codes against loss are interesting. In particular, in the large loss/large squeezing regime, it is indeed better to apply a quantum-limited amplification channel before performing a standard round of GKP error correction. The fact that this scheme outperforms pure loss followed by standard error correction is not necessarily surprising, since the amplification is designed with knowledge of the noise model acting on the system. However, non-amplified loss does outperform amplified loss in the likely experimental regimes of interest, leaving open the possibility of other schemes, such as using amplification levels less than $1/(1-\gamma)$, which may optimally correct for these lower levels of loss.

4.6.4 White-noise Dephasing

Finally, we conclude our numerical analysis by considering a white-noise dephasing channel, which we define by its continuous Kraus decomposition:

$$\mathcal{D}^\sigma(\hat{\rho}) = \frac{1}{\sqrt{2\pi\sigma^2}} \int_{\mathbb{R}} d\phi\; e^{-\phi^2/(2\sigma^2)} e^{i\phi\hat{a}^\dagger\hat{a}}\hat{\rho}e^{-i\phi\hat{a}^\dagger\hat{a}} \tag{4.42}$$

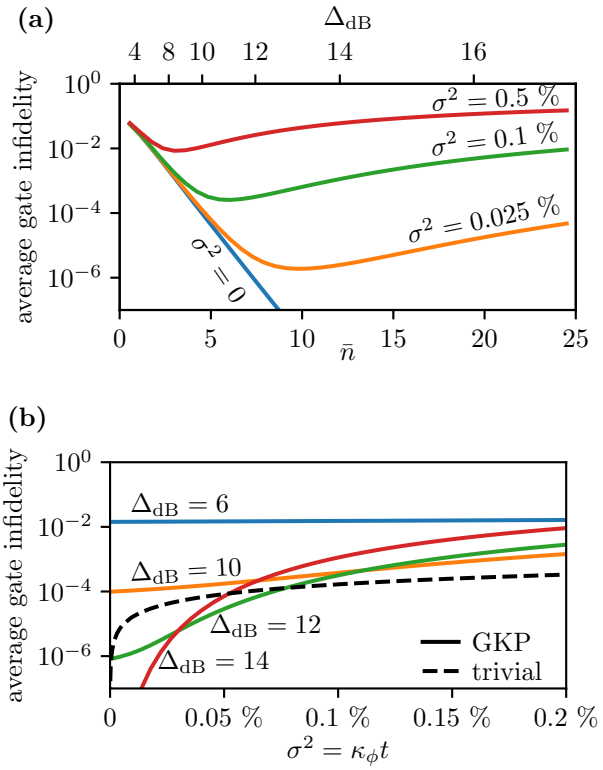

Figure 4.5: Average gate infidelities of the logical noise channels corresponding to dephasing acting on approximate single-mode square GKP qubit codestates $(\mathcal{D}^\sigma \circ \mathcal{E}^\Delta)_{\mathcal{L},\mathrm{o}}$. (a) The average gate infidelities plotted as a function of Δ, where $\bar{n} \approx \frac{1}{2\Delta^2} - \frac{1}{2}$ is the average photon number of the approximate GKP encoded maximally mixed state. (b) Average gate infidelities as a function of $\sigma^2 = \kappa_\phi t$, which represents the loss of logical information as approximate GKP codestates evolve in time under dephasing. This is compared to the loss of logical information stored in a trivial Fock encoding under the same pure dephasing channel.

with chi-representation:

$$c^{\sigma}_{\mathcal{D}}(\mathbf{u},\mathbf{v}) = \frac{1}{\sqrt{2\pi\sigma^2}}\int_{\mathbb{R}} d\phi\; e^{-\phi^2/(2\sigma^2)} e^{-\frac{i\pi}{2}\cot(\phi)\left(|\mathbf{u}|^2-|\mathbf{v}|^2\right)} \tag{4.43}$$

This time, $\sigma^2 = \kappa_\phi t$ represents the amount of dephasing applied to a system evolving under the master equation $\dot{\hat{\rho}} = \kappa_\phi \mathcal{D}[\hat{a}^\dagger \hat{a}]\hat{\rho}$ for some time t. The term "white noise" refers to the Gaussian distribution of the Kraus operators and serves as a convenient analytical tool to analyse general dephasing errors. We note that although our results should give a general indication of the performance of GKP codes against dephasing, typical experiments observe dephasing which is far from white noise and instead has some non-Gaussian distribution.

Again, we precompose the dephasing channel with the envelope operator, and plot the average gate infidelity of the logical noise channel $(\mathcal{D}^\sigma \circ \mathcal{E}^\Delta)_{\mathcal{L}}$ as a function of both Δ and σ in fig. 4.5. In this case, we perform the integrals over $\mathbf{v}$ analytically and perform the resulting integral over ϕ numerically. For any given σ, we again find an optimal Δ which corrects against the dephasing. This can be understood intuitively by noting that dephasing applies random rotations to a state in phase space. As $\Delta \to 0$, the approximate GKP codestate becomes more widely distributed in phase space and is thus more vulnerable to the effects of dephasing.

Comparing our results to experiments is more difficult in the case of dephasing since measurements of an effective white-noise dephasing rate are less common. However, state-of-the-art cavities have been constructed with $T_\phi \approx 1$ ms [91]. At this rate, dephasing acting over a time of 1 μs results in $\sigma^2 \approx 0.1\%$. In contrast to loss, this amount of dephasing severely affects GKP codestates, performing below the break-even point for a pure dephasing channel. Indeed, for a GKP squeezing of $\Delta_{\text{dB}} = 10$, one needs to be performing error correction on a timescale of less than 2 μs to remain below an average gate infidelity of 10^{-3} even with state-of-the-art dephasing rates. While these results may be discouraging, there are a number of factors which may mitigate these results. First, realistic dephasing may perform significantly differently from white-noise dephasing. Second, we are viewing the system in the paradigm of active error correction cycles in between which noise is acting unmitigated. However, there are alternative dissipative schemes which may prevent dephasing from having such an effect on the codestates. Finally, our estimate of the error correction cycle time is very crude, so realistic experiments may be able to reduce this time enough for GKP codes to cope with the amount of dephasing.

4.7 Conclusion

We began the paper by providing an overview of multi-mode GKP codes by describing the parameters $(\Sigma, \mathbf{d})$ that specify the lattice generators of a given GKP code. Then in section 4.3, we introduced the stabiliser subsystem decomposition $\mathcal{H} = \mathcal{L} \otimes_{\mathcal{G}} \mathcal{S}$ of the CV Hilbert space into a logical and stabiliser subspace. We moreover showed that the partial trace $\mathrm{tr}_{\mathcal{S}}$ corresponds to ideal decoding over the primitive cell $\mathcal{P}$, justifying our use of the stabiliser subsystem decomposition over previously-developed alternatives. We then explored the properties of the subsystem decomposition, and used three transformations (cell, Gaussian and dimension transformations) to connect states in the stabiliser subsystem decomposition to Zak states (section 4.4) and to describe logical Clifford gates in our formalism (section 4.5). Finally, in section 4.6 we introduced a general method to calculate the logical effect of various noise operators on GKP codes, which we then used to analyse the envelope operator, pure loss channels, and Gaussian displacement channels in regimes that are unreachable by Fock space simulations.

In sections 4.5 and 4.6 we provided a number of exciting applications of the stabiliser subsystem decomposition to implement logical gates and analyse the effects of noise. However, there is significantly more work that must be done to provide analysis that is useful for experiments. In particular, one could incorporate other techniques from fault-tolerant literature, such as gate teleportation, subsystem codes [83] and gauge fixing [89], into the GKP setting using our formalism. Furthermore, analysing more realistic noise sources such as non-white-noise dephasing, Kerr non-linearities, and approximate error correction is required to model the dominant sources of errors in current experiments. In particular, analysing the dissipative GKP error-correction schemes used in Ref. [35, 30] in terms of the subsystem decomposition may provide insights into how the schemes work. We add that our formalism may allow for more accurate simulations of concatenated GKP codes using the logical action of noise channels and logical gates defined by the stabiliser decomposition.

Finally, we note that our simulation methods rely on the integration of characteristic functions over a cell $\mathcal{P}$. In this work, we were able to produce our results by evaluating these integrals analytically over the single-mode square GKP Voronoi cell. However, higher-dimensional integrals over non-rectangular and/or multi-mode primitive cells cannot be evaluated analytically and thus require numerical integration which may not scale favourably, and as such work can be done to optimise the numerical integration methods for multi-mode GKP codes.

Appendices

4.A Comparison to Ref. [3]

In this appendix, we briefly compare the stabiliser subsystem decomposition to the modular subsystem decomposition [3], for the single-mode square qubit GKP code $\mathcal{G}_{\text{sq}} = (I_2, 2, \mathcal{V}_{\text{sq}})$. We show that the stabiliser subsystem decomposition is not equivalent to the decomposition of Ref. [3] by explicitly decomposing example states and operators into each subsystem decomposition. We also show explicitly the different choices of phase in the definitions of each decomposition in the Zak basis which lead to their different properties. Since the stabiliser subsystem decomposition is designed to describe ideal GKP codes, it is our opinion that the decompositions of Ref. [3] define a bosonic code that shares the same ideal codespace as the GKP code, but has a non-ideal error correction procedure defined by the partial trace operation.

In order to define a modular subsystem decomposition following Ref. [3], one must choose a quadrature in which to decompose the Hilbert space $\mathcal{H}$. We choose the position basis for this purpose and refer to this decomposition as the modular-position subsystem decomposition ($\mathcal{Q}$). Any real number x can be decomposed into a sum

$$x = \sqrt{\pi}(2s + \mu) + r \tag{4.44}$$

where $s \in \mathbb{Z}$, $\mu \in \mathbb{Z}_2$, and $r \in (-\sqrt{\pi}/2, \sqrt{\pi}/2]$. Introducing the modular notation: $x = a\lfloor x \rceil_a + \{x\}_a$, where $\lfloor x \rceil_a \in \mathbb{Z}$ and $\{x\}_a \in (-a/2, a/2]$, we can write $r = \{x\}_{\sqrt{\pi}}$, $s = \left\lfloor \lfloor x \rceil_{\sqrt{\pi}} \right\rceil_2$ and $\mu = \left\{ \lfloor x \rceil_{\sqrt{\pi}} \right\}_2$. Then, the modular-position subsystem decomposition is defined on the position eigenstates as:

$$|x\rangle_q = |\mu\rangle \otimes_{\mathcal{Q}} |\sqrt{\pi}s + r\rangle_q \tag{4.45}$$

where we use the subscript q on the non-logical "gauge" mode to signify that the state is a position eigenstate of the gauge mode.

Due to this choice, the position and momentum quadratures are not treated symmetrically, resulting in a number of undesirable properties in the subsystem decomposition. To make our point explicit, let us give two concrete examples demonstrating the asymmetries of the modular-position subsystem decomposition. First, the modular-position subsystem decomposition treats the position and momentum quadratures asymmetrically, and as such the Fourier transform operator $e^{i\pi\hat{a}^\dagger\hat{a}/2}$ does not have a neat decomposition over the subsystem. In contrast, the Fourier transform operator can be written in the stabiliser subsystem decomposition as a product of

CV state	Stabiliser partial trace	Mod-q partial trace
$\|\phi_+\rangle$	$\frac{1}{2}\hat{I} + \frac{1}{\pi}\hat{X}$	$\frac{1}{2}(\hat{I} + \hat{X})$
$\|\phi_-\rangle$	$\frac{1}{2}\hat{I} + \frac{1}{\pi}\hat{X}$	$\frac{1}{2}\hat{I}$

Table 4.1: The partial trace of the states $|\phi_\pm\rangle = \frac{1}{\sqrt{2}}\big(|0\rangle_q + |\pm\sqrt{\pi}\rangle_q\big)$ in the square single-mode stabiliser and modular-position subsystem decompositions. The stabiliser subsystem decomposition gives the same result for both states, while the modular-position subsystem decomposition gives different results, revealing an asymmetry in superpositions of left and right translations of the position quadrature.

operators acting on $\mathcal{L}$ and $\mathcal{S}$:

$$e^{i\pi\hat{a}^\dagger\hat{a}/2} = \bar{H}_{\text{sq}} = \hat{H} \otimes_{\mathcal{G}_{\text{sq}}} \hat{R}(\pi/2) \tag{4.46}$$

where $\hat{R}(\pi/2)$ provides a $\pi/2$ anticlockwise rotation to the vector $\mathbf{k}$.

Second, we take the partial trace of the two CV states $|\phi_\pm\rangle = \big(|0\rangle_q + |\pm\sqrt{\pi}\rangle_q\big)/\sqrt{2}$ (see table 4.1). These states are chosen to reveal an asymmetry between left and right translations in position in the modular-position subsystem decomposition, since the partial trace results in a pure final state $|+\rangle\langle+|$ for $|\phi_+\rangle$ but the maximally mixed state $\hat{\rho} = \hat{I}/2$ for $|\phi_-\rangle$. In contrast, the stabiliser subsystem decomposition gives the same final state for both initial states.

Now we turn our attention to the Zak basis representation of each decomposition. Recall our (rescaled) definition of Zak states $\mathcal{Z}_2$:

$$|\mathbf{k}\rangle_{\mathcal{Z}_2} = \sqrt{2}\sqrt[4]{\pi}e^{i\pi k_1 k_2} \sum_{s\in\mathbb{Z}} e^{2\sqrt{2}i\pi k_2 s} \, |\sqrt{2\pi}k_1 + 2\sqrt{\pi}s\rangle_q \tag{4.27$'$}$$

for $\mathbf{k} \in \mathbb{R}^2$. The set of states $\{|\mathbf{k}\rangle_{\mathcal{Z}_2}\}_{\mathbf{k}\in\mathcal{P}_{\mathcal{Z}_2}}$ forms a basis, where $\mathcal{P}_{\mathcal{Z}_2} = \big(-\frac{1}{2\sqrt{2}}, \frac{3}{2\sqrt{2}}\big] \times \big(-\frac{1}{2\sqrt{2}}, \frac{1}{2\sqrt{2}}\big]$. Next, recall the Zak representation of the $\mathcal{G}_{\text{sq}}$ decomposition:

$$|\mu\rangle \otimes_{\mathcal{G}_{\text{sq}}} |\mathbf{k}\rangle = e^{i\pi\mu k_2/\sqrt{2}} \, |\mathbf{k} + \mu\hat{\mathbf{e}}_1/\sqrt{2}\rangle_{\mathcal{Z}_2} \tag{4.31}$$

for $\mu \in \mathbb{Z}_2$ and $\mathbf{k} \in \mathcal{V}_{\text{sq}} = \big(-\frac{1}{2\sqrt{2}}, \frac{1}{2\sqrt{2}}\big]^2$.

In recent work [88], a similar equation was developed for the modular-position subsystem decomposition. Here we provide a similar derivation using our notation in order to directly compare the two decompositions. One can see from the definition

in eq. (4.45) that for any gauge mode state $|\phi\rangle$:

$$\hat{W}\big(\hat{\mathbf{e}}_1/\sqrt{2}\big)\,|0\rangle \otimes_{\mathcal{Q}} |\phi\rangle = |1\rangle \otimes_{\mathcal{Q}} |\phi\rangle \tag{4.47}$$

Next, consider the "left half" of the Zak states $|\mathbf{k}\rangle_{\mathcal{Z}_2}$ for which $\mathbf{k} \in \mathcal{V}_{\text{sq}}$. These states have support only on position eigenstates x with $\mu = 0$, and thus can be decomposed into a state $|0\rangle \otimes_{\mathcal{Q}} |\phi\rangle$. We can then choose a Zak basis $\{|\mathbf{k}\rangle_\zeta\}_{\mathbf{k}\in\mathcal{V}_{\text{sq}}}$ of the gauge mode such that:

$$|0\rangle \otimes_{\mathcal{Q}} |\mathbf{k}\rangle_\zeta = |\mathbf{k}\rangle_{\mathcal{Z}_2} = |0\rangle \otimes_{\mathcal{G}_{\text{sq}}} |\mathbf{k}\rangle \tag{4.48}$$

Now, we can apply eq. (4.47) to find:

$$|1\rangle \otimes_{\mathcal{Q}} |\mathbf{k}\rangle_\zeta = e^{-i\pi k_2/\sqrt{2}}\,|\mathbf{k} + \hat{\mathbf{e}}_1/\sqrt{2}\rangle_{\mathcal{Z}_2} \tag{4.49}$$

which differs from eq. (4.27′) only by a $e^{-\sqrt{2}i\pi k_2}$ phase. This in turn can be viewed as applying a k_2-dependent Z-axis rotation of the logical Bloch sphere, thus altering the properties of the partial trace operation. It is because of this that the stabiliser subsystem decomposition can be seen as a "rephasing" of the modular-position subsystem decomposition which symmetrises the treatment of the position and momentum quadratures.

4.B Orthonormalisation procedure

In this appendix, we present the derivation of the procedure to orthonormalise the codewords of a single-mode GKP code. In particular, we are interested in a logical noise channel in which the first noise source is an envelope operator $e^{-\Delta^2\hat{a}^\dagger\hat{a}}$:

$$\mathcal{N}_{\mathcal{L}}(\hat{\rho}) = \text{tr}_S\Big(\mathcal{N}_2 \circ \mathcal{J}[e^{-\Delta^2\hat{a}^\dagger\hat{a}}]\big(\hat{\rho} \otimes_{\mathcal{G}} |\mathbf{0}\rangle\langle\mathbf{0}|\big)\Big) \tag{4.50}$$

where $\mathcal{J}[\hat{O}]\hat{\rho} = \hat{O}\hat{\rho}\hat{O}^\dagger$ and $\mathcal{N}_2$ is the second noise source (such as loss) and is a CPTP map. Since $e^{-\Delta^2\hat{a}^\dagger\hat{a}}$ is non-unitary, the overall logical noise channel is CP but not TP. However, in order to define a valid quantum channel, one can orthogonalise the approximate codewords $e^{-\Delta^2\hat{a}^\dagger\hat{a}}\,|\bar{0}\rangle$ and $e^{-\Delta^2\hat{a}^\dagger\hat{a}}\,|\bar{1}\rangle$ (where $|\bar{\psi}\rangle = |\psi\rangle \otimes_{\mathcal{G}} |\mathbf{0}\rangle$) via the following equations:

$$|\bar{0}_{\Delta,\text{o}}\rangle = \frac{R_+}{2N_0}e^{-\Delta^2\hat{a}^\dagger\hat{a}}\,|\bar{0}\rangle + \frac{e^{-i\phi}R_-}{2N_1}e^{-\Delta^2\hat{a}^\dagger\hat{a}}\,|\bar{1}\rangle \tag{4.51a}$$

$$|\bar{1}_{\Delta,\text{o}}\rangle = \frac{e^{i\phi}R_-}{2N_0}e^{-\Delta^2\hat{a}^\dagger\hat{a}}\,|\bar{0}\rangle + \frac{R_+}{2N_1}e^{-\Delta^2\hat{a}^\dagger\hat{a}}\,|\bar{1}\rangle \tag{4.51b}$$

where:

$$N_\mu = \| e^{-\Delta^2 \hat{a}^\dagger \hat{a}} |\bar{\mu}\rangle \| \tag{4.52a}$$

$$R_\pm = \frac{1}{\sqrt{1+R}} \pm \frac{1}{\sqrt{1-R}} \tag{4.52b}$$

$$R = \frac{\left| \langle \bar{0}| e^{-2\Delta^2 \hat{a}^\dagger \hat{a}} |\bar{1}\rangle \right|}{N_0 N_1} \tag{4.52c}$$

$$\phi = \arg\left(\langle 0| e^{-2\Delta^2 \hat{a}^\dagger \hat{a}} |1\rangle \right) \tag{4.52d}$$

Conveniently, the inner products $\langle \bar{\mu}| e^{-2\Delta^2 \hat{a}^\dagger \hat{a}} |\bar{\nu}\rangle$ which define the orthonormalisation can be obtained solely from the logical envelope channel $\mathcal{E}_\mathcal{L}^\Delta$ via:

$$\begin{aligned} \mathrm{tr}(\mathcal{E}_\mathcal{L}^\Delta |\nu\rangle\langle\mu|) &= \mathrm{tr}_\mathcal{L}\left(\mathrm{tr}_\mathcal{S}\left(\mathcal{J}[e^{-\Delta^2 \hat{a}^\dagger \hat{a}}] |\bar{\nu}\rangle \langle\bar{\mu}| \right) \right) \\ &= \langle \bar{\mu}| e^{-2\Delta^2 \hat{a}^\dagger \hat{a}} |\bar{\nu}\rangle \end{aligned} \tag{4.53}$$

since tracing over the stabiliser subsystem followed by the logical subsystem is equivalent to the total trace over the entire mode.

To apply this to our original problem, we define the orthonormalisation matrix $C(\Delta) = \{c_{ij}(\Delta)\}$ with coefficients given in eq. (4.51) such that $|\bar{\psi}_{\Delta,\mathrm{o}}\rangle = e^{-\Delta^2 \hat{a}^\dagger \hat{a}} C(\Delta) |\bar{\psi}\rangle$. Thus, we can define a CPTP map $\mathcal{N}_{\mathcal{L},\mathrm{o}} = \mathcal{N}_\mathcal{L} \circ \mathcal{J}[C(\Delta)]$. This resulting map is identical to the original map but with the codewords orthonormalised, as required.

4.C Envelope operator simulations as $\Delta \to 0$

In this appendix, we briefly discuss the utility of our simulations in the ideal limit $\Delta \to 0$. In particular, our simulations of the envelope operator, loss, and Gaussian displacements for the square GKP code are all analytic. As long as the error rate is sufficiently low, one can truncate each of the sums in eq. (4.36) to $s_i, t_i = -1, 0, 1$ to retain only the leading order sources of error. Then, the simulations can be run for arbitrarily squeezed states by simply evaluating each analytic expression for the integral and adding the relevant terms to the superoperator $\mathcal{E}^\Delta$. Following this, we again use section 4.B to calculate $\mathcal{E}_\mathrm{o}^\Delta$ from $\mathcal{E}^\Delta$, and then we can directly extract the average gate infidelity. The run-time of this procedure is limited only by the analytic evaluation of the expressions for the superoperator with large enough precision to reach the extremely low infidelities.

As a proof of principle, in fig. 4.6 we present the average gate infidelities of an approximate GKP codestate with no other noise, and an approximate GKP

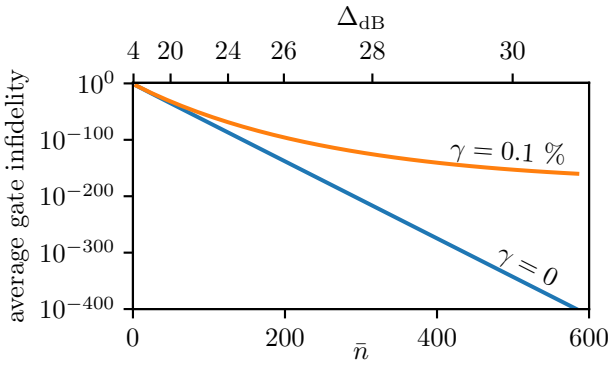

Figure 4.6: Average gate infidelity of the logical noise channel $(\mathcal{L}^{\gamma} \circ \mathcal{E}^{\Delta})_{\mathcal{L},\text{o}}$ corresponding to loss applied to approximate single-mode square GKP qubit codestates. We show two different values of loss given by $\gamma = 0$, 0.1 %, and show the plot as a function of Δ, where $\bar{n} \approx \frac{1}{2\Delta^2} - \frac{1}{2}$ is the average photon number of the approximate GKP encoded maximally mixed state. This demonstrates that our methods can be applied easily to approximate GKP codestates with a comically large average photon number.

codestate with a loss-rate of $\gamma = 0.1$ %, up to an average GKP photon number of ~ 600 ($\Delta \approx 0.029$, $\Delta_{\text{dB}} \approx 30.8$). At this level of squeezing, Fock space simulations would require a truncation dimension which excludes at most $\sim 10^{-400}$ of the total support of the state so that the leading source of error in the simulation is due to the approximate GKP codestate. Moreover, the variance in the photon number distribution of the approximate GKP codestate is also roughly 600, rendering Fock space simulations completely infeasible. While the $\gamma = 0$ curve has a well-known approximate analytic expression which tends to be exact as $\Delta \to 0$, it is less clear how one would determine a similar analytic expression for the infidelity associated with $\gamma = 0.1$ %, particularly in the regime around $\bar{n} \approx 600$ as the curve reaches the optimal Δ. Although such photon numbers are unlikely to ever be experimentally realised, fig. 4.6 demonstrates the efficiency and numerical stability of our simulations when applied to square GKP codewords with arbitrary amounts of squeezing.

Chapter 5

Conclusion

In this thesis, we have presented a number of practical and theoretical proposals which bring fault-tolerant quantum computers with GKP codes closer to reality. We began with a background of quantum computation with GKP codes in chapter 2. Then in chapter 3, we presented proposals for performing logical gates and measurements for GKP codes implemented in superconducting circuits. In particular, we described a scheme for reducing the number of logical Clifford gates which are needed to implement arbitrary Clifford circuits. We then presented a modification to the error-correction scheme which can reduce the error rate of logical Clifford gates by up to two orders of magnitude. In the last section of chapter 3, we proposed a scheme to reduce the effect of measurement inefficiency on the read-out error rate of logical Pauli measurements. Following this, in chapter 4, we presented the stabiliser subsystem decomposition for GKP codes, and discussed its mathematical properties. Finally, we used the stabiliser subsystem decomposition to simulate various sources of noise exactly, and demonstrated that GKP codes perform comparatively poorly against dephasing compared to against pure loss.

The work in this thesis provides a clear view of both the advantages of quantum computing with GKP codes and its remaining challenges. Our proposal for performing logical Clifford gates using only generalised controlled gates, and for modifying the error-correction patch after each gate, take advantage of the Gaussianity of logical GKP Clifford gates. Meanwhile, our simulations of GKP codes demonstrate its ability to convert even relatively large amounts of pure loss into a manageable logical error rate. In contrast, our analysis of dephasing noise and measurement inefficiency reveals GKP codes' relative weakness against correcting both of these types of errors. The effect of measurement inefficiency may be reduced by using an additional readout ancilla mode (as discussed in chapter 3), but it remains to be seen whether schemes can be devised to reduce the effect of dephasing noise on GKP codes. We do note, however, that we have not considered noise channels consisting of a simultaneous application of pure loss and dephasing, and it is likely that if loss rates are larger than dephasing rates, GKP codes will still perform favourably.

Our work also provides interesting insights into the nature of "ideal" decoding. In chapter 4, for the purposes of constructing the subsystem stabiliser decomposition,

we defined ideal decoding as consisting of a measurement of the stabilisers followed by a translation lying within a primitive cell $\mathcal{P}$ that returns the state to the codespace. This framework allows a large set of "decoders", each of which is defined by a different primitive cell $\mathcal{P}$. This flexibility allows some level of optimisation, for instance, by modifying the primitive cell after the application of a logical Clifford gate, or to implement "maximum likelihood" decoding in multi-mode codes [90]. However, this formalism excludes decoders that cannot be described as a stabiliser measurement followed by a translation. If one has knowledge of the noise model acting on the system, one could design decoders that have error rates that are lower than any "ideal" decoders discussed in chapter 4. Indeed, one can calculate the *optimal* decoder given a set of codestates using semi-definite programming [31]. Concretely, we have seen that applying quantum-limited amplification following a pure loss noise channel before applying an "ideal" decoder can improve performance for sufficiently large amounts of loss. In general, it may be interesting to construct decoders that perform better than our "ideal" decoders but are still possible to implement experimentally, and see if they can be usefully described in the stabiliser subsystem decomposition or a modified version of it.

Finally, we introduced a new method for simulating GKP circuits in chapter 4, but in this work, we only considered single-mode square GKP codes. It is therefore of significant interest to find a method of efficiently scaling such simulations up to multi-mode GKP codes to assess the performance of GKP code concatenations against pure loss and dephasing acting exactly on the system. Such simulations would allow a far more precise assessment of the potential capabilities of both GKP code concatenations and novel multi-mode GKP codes when compared to previous studies which have relied on a Gaussian random displacement model. The stabiliser subsystem decomposition more generally may also provide a convenient theoretical framework for the development and analysis of native-to-GKP fault tolerant gates and distillation schemes.

Bibliography

[1] Michael A Nielsen and Isaac Chuang. Quantum computation and quantum information, 2002.

[2] Howard M Wiseman and Gerard J Milburn. Quantum measurement and control. Cambridge university press, 2009.

[3] Giacomo Pantaleoni, Ben Q Baragiola, and Nicolas C Menicucci. Modular bosonic subsystem codes. Physical Review Letters, 125(4):040501, 2020.

[4] Peter W Shor. Algorithms for quantum computation: discrete logarithms and factoring. In Proceedings 35th annual symposium on foundations of computer science, pages 124–134. Ieee, 1994.

[5] Yudong Cao, Jonathan Romero, Jonathan P Olson, Matthias Degroote, Peter D Johnson, Mária Kieferová, Ian D Kivlichan, Tim Menke, Borja Peropadre, Nicolas PD Sawaya, et al. Quantum chemistry in the age of quantum computing. Chemical reviews, 119(19):10856–10915, 2019.

[6] Bela Bauer, Sergey Bravyi, Mario Motta, and Garnet Kin-Lic Chan. Quantum algorithms for quantum chemistry and quantum materials science. Chemical Reviews, 120(22):12685–12717, 2020.

[7] Nicolas Gisin, Grégoire Ribordy, Wolfgang Tittel, and Hugo Zbinden. Quantum cryptography. Reviews of modern physics, 74(1):145, 2002.

[8] Stefano Pirandola, Ulrik L Andersen, Leonardo Banchi, Mario Berta, Darius Bunandar, Roger Colbeck, Dirk Englund, Tobias Gehring, Cosmo Lupo, Carlo Ottaviani, et al. Advances in quantum cryptography. Advances in optics and photonics, 12(4):1012–1236, 2020.

[9] Joss Bland-Hawthorn, Matthew J Sellars, and John G Bartholomew. Quantum memories and the double-slit experiment: implications for astronomical interferometry. JOSA B, 38(7):A86–A98, 2021.

[10] Frank Arute, Kunal Arya, Ryan Babbush, Dave Bacon, Joseph C Bardin, Rami Barends, Rupak Biswas, Sergio Boixo, Fernando GSL Brandao, David A Buell, et al. Quantum supremacy using a programmable superconducting processor. Nature, 574(7779):505–510, 2019.

[11] Sebastian Krinner, Nathan Lacroix, Ants Remm, Agustin Di Paolo, Elie Genois, Catherine Leroux, Christoph Hellings, Stefania Lazar, Francois Swiadek, Johannes Herrmann, et al. Realizing repeated quantum error correction in a distance-three surface code. Nature, 605(7911):669–674, 2022.

[12] Simon J Devitt, Austin G Fowler, and Lloyd CL Hollenberg. Investigating the practical implementation of Shor's algorithm. In Micro-and Nanotechnology: Materials, Processes, Packaging, and Systems II, volume 5650, pages 483–494. International Society for Optics and Photonics, 2005.

[13] Peter W Shor. Scheme for reducing decoherence in quantum computer memory. Phys. Rev. A, 52, 10 1995.

[14] Exponential suppression of bit or phase errors with cyclic error correction. Nature, 595(7867):383–387, 2021.

[15] Rajeev Acharya, Igor Aleiner, Richard Allen, Trond I Andersen, Markus Ansmann, Frank Arute, Kunal Arya, Abraham Asfaw, Juan Atalaya, Ryan Babbush, et al. Suppressing quantum errors by scaling a surface code logical qubit. arXiv preprint arXiv:2207.06431, 2022.

[16] A. Yu. Kitaev. Fault-tolerant quantum computation by anyons. Ann. Phys. (N. Y.), 303(1), 1 2003.

[17] Sergey B Bravyi and A Yu Kitaev. Quantum codes on a lattice with boundary. arXiv preprint quant-ph/9811052, 1998.

[18] Austin G Fowler, Matteo Mariantoni, John M Martinis, and Andrew N Cleland. Surface codes: Towards practical large-scale quantum computation. Physical Review A, 86(3):032324, 2012.

[19] Daniel Litinski. A game of surface codes: Large-scale quantum computing with lattice surgery. Quantum, 3:128, 2019.

[20] Felix Thomsen, Markus S Kesselring, Stephen D Bartlett, and Benjamin J Brown. Low-overhead quantum computing with the color code. arXiv preprint arXiv:2201.07806, 2022.

[21] Isaac L Chuang, Debbie W Leung, and Yoshihisa Yamamoto. Bosonic quantum codes for amplitude damping. Physical Review A, 56(2):1114, 1997.

[22] Paul T Cochrane, Gerard J Milburn, and William J Munro. Macroscopically distinct quantum-superposition states as a bosonic code for amplitude damping. Physical Review A, 59(4):2631, 1999.

[23] Daniel Gottesman, Alexei Kitaev, and John Preskill. Encoding a qubit in an oscillator. Phys. Rev. A, 64, 6 2001.

[24] Marios H Michael, Matti Silveri, RT Brierley, Victor V Albert, Juha Salmilehto, Liang Jiang, and Steven M Girvin. New class of quantum error-correcting codes for a bosonic mode. Physical Review X, 6(3):031006, 2016.

[25] Arne L Grimsmo, Joshua Combes, and Ben Q Baragiola. Quantum computing with rotation-symmetric bosonic codes. Physical Review X, 10(1):011058, 2020.

[26] Nissim Ofek, Andrei Petrenko, Reinier Heeres, Philip Reinhold, Zaki Leghtas, Brian Vlastakis, Yehan Liu, Luigi Frunzio, SM Girvin, Liang Jiang, et al. Extending the lifetime of a quantum bit with error correction in superconducting circuits. Nature, 536(7617):441–445, 2016.

[27] P Campagne-Ibarcq, A Eickbusch, S Touzard, E Zalys-Geller, NE Frattini, VV Sivak, P Reinhold, S Puri, S Shankar, RJ Schoelkopf, et al. Quantum error correction of a qubit encoded in grid states of an oscillator. Nature, 584(7821):368–372, 2020.

[28] Alec Eickbusch, Volodymyr Sivak, Andy Z Ding, Salvatore S Elder, Shantanu R Jha, Jayameenakshi Venkatraman, Baptiste Royer, SM Girvin, Robert J Schoelkopf, and Michel H Devoret. Fast universal control of an oscillator with weak dispersive coupling to a qubit. arXiv preprint arXiv:2111.06414, 2021.

[29] Christa Flühmann, Thanh Long Nguyen, Matteo Marinelli, Vlad Negnevitsky, Karan Mehta, and Jonathan P Home. Encoding a qubit in a trapped-ion mechanical oscillator. Nature, 566(7745):513–517, 2019.

[30] Brennan de Neeve, Thanh Long Nguyen, Tanja Behrle, and Jonathan Home. Error correction of a logical grid state qubit by dissipative pumping. arXiv preprint arXiv:2010.09681, 2020.

[31] Victor V Albert, Kyungjoo Noh, Kasper Duivenvoorden, Dylan J Young, RT Brierley, Philip Reinhold, Christophe Vuillot, Linshu Li, Chao Shen, SM Girvin, et al. Performance and structure of single-mode bosonic codes. Physical Review A, 97(3):032346, 2018.

[32] Ben Q Baragiola, Giacomo Pantaleoni, Rafael N Alexander, Angela Karanjai, and Nicolas C Menicucci. All-Gaussian universality and fault tolerance with the Gottesman-Kitaev-Preskill code. Physical Review Letters, 123(20):200502, 2019.

[33] Timo Hillmann, Fernando Quijandría, Arne L Grimsmo, and Giulia Ferrini. Performance of teleportation-based error correction circuits for bosonic codes with noisy measurements. arXiv preprint arXiv:2108.01009, 2021.

[34] BM Terhal and Daniel Weigand. Encoding a qubit into a cavity mode in circuit qed using phase estimation. Physical Review A, 93(1):012315, 2016.

[35] Baptiste Royer, Shraddha Singh, and SM Girvin. Stabilization of finite-energy Gottesman-Kitaev-Preskill states. arXiv preprint arXiv:2009.07941, 2020.

[36] Hayata Yamasaki, Takaya Matsuura, and Masato Koashi. Cost-reduced all-Gaussian universality with the Gottesman-Kitaev-Preskill code: Resource-theoretic approach to cost analysis. Physical Review Research, 2(2):023270, 2020.

[37] Ilan Tzitrin, J Eli Bourassa, Nicolas C Menicucci, and Krishna Kumar Sabapathy. Progress towards practical qubit computation using approximate Gottesman-Kitaev-Preskill codes. Physical Review A, 101(3):032315, 2020.

[38] Kyungjoo Noh, Victor V Albert, and Liang Jiang. Quantum capacity bounds of gaussian thermal loss channels and achieable rates with Gottesman-Kitaev-Preskill codes. IEEE Transactions on Information Theory, 65(4):2563–2582, 2018.

[39] Jacob Hastrup, Mikkel V Larsen, Jonas S Neergaard-Nielsen, Nicolas C Menicucci, and Ulrik L Andersen. Unsuitability of cubic phase gates for non-clifford operations on gottesman-kitaev-preskill states. Physical Review A, 103(3):032409, 2021.

[40] Giacomo Pantaleoni, Ben Q Baragiola, and Nicolas C Menicucci. Subsystem analysis of continuous-variable resource states. Physical Review A, 104(1):012430, 2021.

[41] Giacomo Pantaleoni, Ben Q Baragiola, and Nicolas C Menicucci. Hidden qubit cluster states. Physical Review A, 104(1):012431, 2021.

[42] Kosuke Fukui, Akihisa Tomita, Atsushi Okamoto, and Keisuke Fujii. High-threshold fault-tolerant quantum computation with analog quantum error correction. Physical review X, 8(2):021054, 2018.

[43] Kosuke Fukui. High-threshold fault-tolerant quantum computation with the gkp qubit and realistically noisy devices. arXiv preprint arXiv:1906.09767, 2019.

[44] Christophe Vuillot, Hamed Asasi, Yang Wang, Leonid P Pryadko, and Barbara M Terhal. Quantum error correction with the toric Gottesman-Kitaev-Preskill code. Physical Review A, 99(3):032344, 2019.

[45] Kyungjoo Noh and Christopher Chamberland. Fault-tolerant bosonic quantum error correction with the surface–Gottesman-Kitaev-Preskill code. Physical Review A, 101(1):012316, 2020.

[46] Kyungjoo Noh, Christopher Chamberland, and Fernando GSL Brandão. Low overhead fault-tolerant quantum error correction with the surface-gkp code. arXiv preprint arXiv:2103.06994, 2021.

[47] Jonathan Conrad, Jens Eisert, and Francesco Arzani. Gottesman-kitaev-preskill codes: A lattice perspective. Quantum, 6:648, 2022.

[48] Baptiste Royer, Shraddha Singh, and SM Girvin. Encoding qubits in multimode grid states. PRX Quantum, 3(1):010335, 2022.

[49] Robert Raussendorf, Daniel E Browne, and Hans J Briegel. Measurement-based quantum computation on cluster states. Physical review A, 68(2):022312, 2003.

[50] Sara Bartolucci, Patrick Birchall, Hector Bombin, Hugo Cable, Chris Dawson, Mercedes Gimeno-Segovia, Eric Johnston, Konrad Kieling, Naomi Nickerson, Mihir Pant, et al. Fusion-based quantum computation. arXiv preprint arXiv:2101.09310, 2021.

[51] Daniel Gottesman. The heisenberg representation of quantum computers. arXiv preprint quant-ph/9807006, 1998.

[52] Bryan Eastin and Emanuel Knill. Restrictions on transversal encoded quantum gate sets. Physical review letters, 102(11):110502, 2009.

[53] Sergey Bravyi and Alexei Kitaev. Universal quantum computation with ideal Clifford gates and noisy ancillas. Physical Review A, 71(2):022316, 2005.

[54] Takaya Matsuura, Hayata Yamasaki, and Masato Koashi. Equivalence of approximate gottesman-kitaev-preskill codes. Physical Review A, 102(3):032408, 2020.

[55] Nicolas C Menicucci. Fault-tolerant measurement-based quantum computing with continuous-variable cluster states. Physical review letters, 112(12):120504, 2014.

[56] Uri Vool and Michel Devoret. Introduction to quantum electromagnetic circuits. International Journal of Circuit Theory and Applications, 45(7):897–934, 2017.

[57] Alexandre Blais, Arne L Grimsmo, SM Girvin, and Andreas Wallraff. Circuit quantum electrodynamics. arXiv preprint arXiv:2005.12667, 2020.

[58] SE Rasmussen, KS Christesen, SP Pedersen, LB Kristensen, T Bækkegaard, NJS Loft, and NT Zinner. Superconducting circuit companion—an introduction with worked examples. PRX Quantum, 2(4):040204, 2021.

[59] BD Josephson. Possible new effects in superconductive tunneling. Phys. Lett, 1(7):251, 1962.

[60] NE Frattini, U Vool, S Shankar, A Narla, KM Sliwa, and MH Devoret. 3-wave mixing Josephson dipole element. Applied Physics Letters, 110(22):222603, 2017.

[61] Arne L Grimsmo and Shruti Puri. Quantum error correction with the gottesman-kitaev-preskill code. PRX Quantum, 2(2):020101, 2021.

[62] Chris Macklin, K O'Brien, D Hover, ME Schwartz, V Bolkhovsky, X Zhang, WD Oliver, and I Siddiqi. A near–quantum-limited josephson traveling-wave parametric amplifier. Science, 350(6258):307–310, 2015.

[63] S Touzard, A Kou, NE Frattini, VV Sivak, S Puri, A Grimm, L Frunzio, S Shankar, and MH Devoret. Gated conditional displacement readout of superconducting qubits. Physical review letters, 122(8):080502, 2019.

[64] Prahlad Warszawski, Howard M Wiseman, and Andrew C Doherty. Solving quantum trajectories for systems with linear heisenberg-picture dynamics and gaussian measurement noise. Physical Review A, 102(4):042210, 2020.

[65] Christopher Chamberland, Pavithran Iyer, and David Poulin. Fault-tolerant quantum computing in the pauli or clifford frame with slow error diagnostics. Quantum, 2:43, 2018.

[66] Yaxing Zhang, Brian J Lester, Yvonne Y Gao, Liang Jiang, RJ Schoelkopf, and SM Girvin. Engineering bilinear mode coupling in circuit QED: Theory and experiment. Physical Review A, 99(1):012314, 2019.

[67] Michael A Nielsen. A simple formula for the average gate fidelity of a quantum dynamical operation. Physics Letters A, 303(4):249–252, 2002.

[68] HM Wiseman. Quantum trajectories and quantum measurement theory. Quantum and Semiclassical Optics: Journal of the European Optical Society Part B, 8(1):205, 1996.

[69] Crispin W Gardiner et al. Handbook of stochastic methods, volume 3. springer Berlin, 1985.

[70] Barbara Terhal, Jonathan Conrad, and Christophe Vuillot. Towards scalable bosonic quantum error correction. Quantum Science and Technology, 2020.

[71] Atharv Joshi, Kyungjoo Noh, and Yvonne Y Gao. Quantum information processing with bosonic qubits in circuit qed. Quantum Science and Technology, 6(3):033001, 2021.

[72] Reinier W Heeres, Philip Reinhold, Nissim Ofek, Luigi Frunzio, Liang Jiang, Michel H Devoret, and Robert J Schoelkopf. Implementing a universal gate set on a logical qubit encoded in an oscillator. Nature communications, 8(1):1–7, 2017.

[73] Ling Hu, Yuwei Ma, Weizhou Cai, Xianghao Mu, Yuan Xu, Weiting Wang, Yukai Wu, Haiyan Wang, YP Song, C-L Zou, et al. Quantum error correction and universal gate set operation on a binomial bosonic logical qubit. Nature Physics, 15(5):503–508, 2019.

[74] Alexander Grimm, Nicholas E Frattini, Shruti Puri, Shantanu O Mundhada, Steven Touzard, Mazyar Mirrahimi, Steven M Girvin, Shyam Shankar, and Michel H Devoret. Stabilization and operation of a kerr-cat qubit. Nature, 584(7820):205–209, 2020.

[75] Raphaël Lescanne, Marius Villiers, Théau Peronnin, Alain Sarlette, Matthieu Delbecq, Benjamin Huard, Takis Kontos, Mazyar Mirrahimi, and Zaki Leghtas. Exponential suppression of bit-flips in a qubit encoded in an oscillator. Nature Physics, 16(5):509–513, 2020.

[76] Jeffrey M Gertler, Brian Baker, Juliang Li, Shruti Shirol, Jens Koch, and Chen Wang. Protecting a bosonic qubit with autonomous quantum error correction. Nature, 590(7845):243–248, 2021.

[77] Emanuel Knill and Raymond Laflamme. Theory of quantum error-correcting codes. Phys. Rev. A, 55:900–911, Feb 1997.

[78] Emanuel Knill, Raymond Laflamme, and Lorenza Viola. Theory of quantum error correction for general noise. Phys. Rev. Lett., 84:2525–2528, Mar 2000.

[79] J Zak. Finite translations in solid-state physics. Physical Review Letters, 19(24):1385, 1967.

[80] J Eli Bourassa, Nicolás Quesada, Ilan Tzitrin, Antal Száva, Theodor Isacsson, Josh Izaac, Krishna Kumar Sabapathy, Guillaume Dauphinais, and Ish Dhand. Fast simulation of bosonic qubits via Gaussian functions in phase space. PRX Quantum, 2(4):040315, 2021.

[81] R Simon and N Mukunda. The two-dimensional symplectic and metaplectic groups and their universal cover. In Symmetries in Science VI, pages 659–689. Springer, 1993.

[82] Biswadeb Dutta, Narasimhaiengar Mukunda, Rajiah Simon, et al. The real symplectic groups in quantum mechanics and optics. Pramana, 45(6):471–497, 1995.

[83] David Poulin. Stabilizer formalism for operator quantum error correction. Physical review letters, 95(23):230504, 2005.

[84] James William Harrington. Analysis of quantum error-correcting codes: symplectic lattice codes and toric codes. PhD thesis, 2004.

[85] Scott Aaronson and Daniel Gottesman. Improved simulation of stabilizer circuits. Physical Review A, 70(5):052328, 2004.

[86] Christopher Chamberland, Joel Wallman, Stefanie Beale, and Raymond Laflamme. Hard decoding algorithm for optimizing thresholds under general markovian noise. Phys. Rev. A, 95:042332, Apr 2017.

[87] A Ketterer, A Keller, SP Walborn, T Coudreau, and P Milman. Quantum information processing in phase space: A modular variables approach. Physical Review A, 94(2):022325, 2016.

[88] Giacomo Pantaleoni, Ben Q Baragiola, and Nicolas C Menicucci. [in preparation].

[89] Christophe Vuillot, Lingling Lao, Ben Criger, Carmen García Almudéver, Koen Bertels, and Barbara M Terhal. Code deformation and lattice surgery are gauge fixing. New Journal of Physics, 21(3):033028, 2019.

[90] Jonathan Conrad. Twirling and hamiltonian engineering via dynamical decoupling for gottesman-kitaev-preskill quantum computing. Physical Review A, 103(2):022404, 2021.

[91] Matthew Reagor, Wolfgang Pfaff, Christopher Axline, Reinier W Heeres, Nissim Ofek, Katrina Sliwa, Eric Holland, Chen Wang, Jacob Blumoff, Kevin Chou, et al. Quantum memory with millisecond coherence in circuit qed. Physical Review B, 94(1):014506, 2016.