Non-Markovianity in Logical Fidelity Estimation

by

Athena Marie Caesura

A thesis
presented to the University of Waterloo
in fulfillment of the
thesis requirement for the degree of
Master of Science
in
Physics (Quantum Information)

Waterloo, Ontario, Canada, 2021

Author's Declaration

I hereby declare that I am the sole author of this thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I understand that my thesis may be made electronically available to the public.

Abstract

As quantum devices are progressively scaled and refined, quantum codes will become indispensable to guarantee reliable computation in the presence of noise. Due to the difficulties of characterizing physical noise, direct measures of logical performance will be of paramount use to tune quantum codes [18]. Logical Randomized Benchmarking in [5] aimed to characterize the logical operations directly by performing randomized benchmarking using logical operators and following each operation with perfect QEC. However, due to resource limitations, physical implementations such as [14] only measure the syndrome at the end of a computation.

We develop a generalized framework for procedures that measure the fidelity of logical operations directly, logical fidelity estimation (LFE). We identify several freedoms that arise when attempting to define a fidelity which characterizes quantum error correction, specifically, in how error detection events are processed and in how often error correction is performed. We explore these freedoms by conducting a survey to eliminate inviable procedures guided by the presence of upticks, a standard signature of non-Markovianity [33]. Our survey highlights procedures that are robust to gate-dependent noise and procedures which are implementable in the near-term. We recommend applying post-selection after each gate as a benchmark for quantum codes which is robust to gate-dependent noise. We manage to eliminate some naive near-term protocols, but guaranteeing the robustness when QEC is applied at a at fixed time or when post-selection is applied at the end, are open problems.

Acknowledgements

I would like to thank the University of Waterloo as well as my colleagues Pavithran Iyer and Hakop Pashayan, whose persistent help made this thesis possible.

Dedication

To Cici...

Table of Contents

List of Figures

List of Algorithms

List of Abbreviations

RB	Randomized Benchmarking
LRB	Logical randomized benchmarking
LFE	Logical fidelity estimation
QEC	Quantum error correction
SPAM	State preparation and measurement
LHS	Left hand side
RHS	Right hand side
rep.	representation of a group
irrep.	irreducible representation of a group
indep.	independent, used in reference to gate-independent noise
dep.	dependent, used in reference to gate-dependent noise

List of Symbols

Randomized Benchmarking

m	Gate sequence length
p_{s_m}	Survival probability
$q_{\vec{G}}$	RB sequence outcome
$\mathcal{E}$	Error model
$\mathcal{E}_G$	Gate-dependent error model
$\mathcal{E}_i$	Time-dependent error model

Stabilizer Codes

$\mathbb{L}$	Logical operators
$\mathbb{S}$	Stabilizers
$\mathbb{T}$	Destabilizers
$\mathcal{C}_L$	Logical Clifford group, see definition 3.3
$\mathbb{A}_L$	Logical Asymmetric group, see eq. (7.1)
$\mathcal{R}$	QEC map compiling syndrome measurement and recovery, see definition 3.4.
D_s	Decoding applied in response to the syndrome s.
$\Pi_{\mathbb{S}}$	Code space projector
$\bar{\Pi}_{\mathbb{S}}$	normalized code space projector defined by $\bar{\Pi}_{\mathbb{S}} \lvert\rho\rangle = \frac{1}{\mathrm{Tr}\{\rho\Pi_{\mathbb{S}}\}} \lvert\Pi_{\mathbb{S}}\rho\Pi_{\mathbb{S}}\rangle$

Representation Theory

$\phi(G)$ Representation of the group element G

$\phi(\mathbb{G})$ The group defined by image of the group $\mathbb{G}$ using representation ϕ.

χ_ϕ Character of the representation ϕ. Given by $\chi_\phi(G) = \mathrm{Tr}\{\phi(G)\}$.

m_ϕ Multiplicity of the representation ϕ.

Π_ϕ Projection onto the representation ϕ.

$\mathrm{End}(V)$ The set of linear maps from the vector space V back onto itself.

$\theta(G)$ Implemented approximate representation of G

Other Symbols

$\mathbb{A}_1$ The single qubit asymmetric group. definition 2.3

$|\rho\rangle$ Quantum state given by the density matrix ρ

$\mathfrak{C}$ Commutant, see definition 2.2

$\mathcal{N}$ Normalizer, see definition 2.1

Chapter 1

Introduction

Quantum computation will solve a multitude of real-world problems considered intractable for classical hardware. However, one must overcome quantum hardware's high sensitivity to noise, presenting a significant challenge to achieving scalable computation. As a consequence, much effort has been put into characterizing and minimizing noise in quantum computation. This thesis contributes to this effort by characterizing how a recovery minimizes the effective noise on logical components.

Quantum error correction (QEC) shows great promise, as fault tolerant computation will eliminate the effective logical noise once the physical noise is below a certain threshold [30]. In stabilizer quantum error correction [10], one encodes a k-qubit state $|\psi\rangle$ into an n-qubit state $|\bar{\psi}\rangle$ on which we can perform a stabilizer measurement to detect errors. The experimentalist can use the information from these syndrome measurements to enact a decoding meant to invert the noise model or post-select to purify the signal. Although it is possible to use physical characterizations of the qubits to analytically optimize the fault tolerance threshold for simplistic noise models [2], determining the optimal decoding using this method is always hard [19]. Consequently, it is advantageous to characterize the logical operations directly.

Randomized benchmarking (RB) is a general class of scalable procedures used to estimate figures of merit for noisy quantum devices. Since RB aims to characterize the severity of noise afflicting quantum devices, it is natural to apply RB in an encoded setting to obtain a figure of merit for the efficacy of quantum codes. We generalize two previous procedures that apply RB to quantum codes [5, 14] by defining *Logical Fidelity Estimation* (LFE), a character benchmarking procedure [17] wherein one may insert a recovery map after each logical operation.

We survey a simplified set of LFE protocols by discussing non-Markovianity in several critical cases. We focus on showing the following:

1. In table 6.1 we summarize our survey by enumerating when non-Markovianity occurs in a simplified set of LFE procedures.

2. In theorem 6.3 we show rejected post-selection always LFE is robust to gate-dependent noise.

3. When noise is gate-independent, one may delay applying a decoding to the end of the LFE sequence by theorem 5.4.

4. Conjecture 6.4 speculates that QEC at fixed time LFE is potentially robust to gate-dependent noise, minimizes syndrome measurements, and evaluates QEC. However, the survival probability decay is non-standard.

5. In fig. B.5 we give numerical evidence that discarding post-selected sequences can produce non-standard survival probability decays.

Chapter 2 of this thesis summarizes the basic concepts in quantum information needed to understand this thesis. chapter 3 discusses stabilizer codes and quantum error correction. Chapter 4 gives an introduction to RB and reviews some recent applications of RB to stabilizer codes. We then combine chapters 3 and 4 in chapter 5 to define logical fidelity estimation, a set of procedures that characterize the decrease in noise in the logical components. Then in chapter 6, we survey a simplified set of LFE procedures, using the presence of non-Markovianity to prune inviable procedures. In chapter 7, we give numerical evidence to support the claims made in our survey. We conclude in chapter 8.

I claim all of the work summarized above as my own original work except for theorem 2.6 and it's following corollaries, which stem from work in the suppolementary material of [9]. Chapters 1-4 review previous work, while chapters 5-7 contain the novel contributions of the thesis.

Chapter 2

Quantum Information Review

In this section, we build a toolbox to describe and analyze quantum processes. We assume some familiarity with group theory. Experienced readers need only to note theorem 2.6 and its corollaries before moving on.

2.1 Group Theory Background

We will assume a basic knowledge of group theory and its axioms. The reader can find a quantum focused approach to group theory in [31] if further background is needed. We begin with some basic definitions that will help build the groups required to describe quantum computation.

Definition 2.1 (Normalizer)**.** The normalizer $\mathcal{N}(\mathbb{G}, \mathbb{U})$ for a subgroup $\mathbb{G}$ of a group $\mathbb{U}$ is

$$\mathcal{N}(\mathbb{G}, \mathbb{U}) = \{N \in \mathbb{U} |\ \forall G \in \mathbb{G}\ N^\dagger G N \in \mathbb{G}\}$$

where $\mathbb{G}$ is a subgroup of $\mathbb{U}$.

Definition 2.2 (Commutant)**.** The commutant $\mathfrak{C}(\mathbb{G}, \mathbb{U})$ for a subgroup $\mathbb{G}$ of a group $\mathbb{U}$ is

$$\mathfrak{C}(\mathbb{G}, \mathbb{U}) = \{C \in \mathbb{U} |\ \forall G \in \mathbb{G}\ C^\dagger G C = G\}$$

The most general group n qubits can represent is $SU(2^n)$, so we define the shorthand $\mathcal{N}(\mathbb{G}, SU(2^n)) = \mathcal{N}(\mathbb{G})$. Similarly for the commutator.

Recall the single qubit Paulis $\mathcal{P}_1 \subset SU(2)$.

$$\mathcal{I} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \quad X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \quad Y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix} \quad Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}. \tag{2.1}$$

Taking tensor products of these single qubit Paulis gives the full Pauli group $\mathcal{P}_n$.

The normalizer of the Pauli group is the Clifford group $\mathcal{N}(\mathcal{P}_n) = \mathcal{C}_n$. Typically when we use RB to assess quantum computers, $\mathcal{C}_n$ is sampled for sequences whose product is identity. The Clifford + T gate design is an increasingly simple method to implement full quantum computation with a limited gate set. So we can expect that systems that strive to achieve full quantum complexity will implement $\mathcal{C}_n$. Thus, $\mathcal{C}_n$ is a natural setting from which to construct circuits to evaluate a quantum computer.

However, randomized benchmarking is not limited to using the Clifford group alone. In chapter 7, we utilized a subgroup of $\mathcal{C}_n$ to perform randomized benchmarking.

Definition 2.3. The Asymmetric Group $\mathbb{A}_1$ is generated by the set $\mathbb{A}_1^{gens} = \{Z, HP\}$ where

$$H = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \quad P = \begin{pmatrix} 1 & 0 \\ 0 & -i \end{pmatrix} \quad HP = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & -i \\ 1 & i \end{pmatrix}. \tag{2.2}$$

One can check the relations $HXH^\dagger = Z$, $HYH^\dagger = -Y$, $PZP^\dagger = Z$, and $PXP^\dagger = Y$ by hand. In section 2.3 we provide appropriate material to check that $\mathbb{A}_1$ is a suitable group for randomized benchmarking.

Another important gate is the controlled single qubit Pauli C_σ where $\sigma \in \mathcal{P}_1$. The 4x4 matrix representation is

$$C_\sigma = \begin{pmatrix} \mathcal{I}_2 & 0 \\ 0 & \sigma \end{pmatrix} \tag{2.3}$$

from which we may construct a general controlled Pauli operator C_σ which "controls" n qubits with a single qubit. Take $\sigma = \sigma_1 \otimes \ldots \sigma_n \in P_n$ and we have

$$C_\sigma = \prod_{i=1}^{n} \begin{pmatrix} \mathcal{I}_{2^n} & 0 \\ 0 & \otimes_{j=1}^{n} \sigma_i^{\delta_{i,j}} \end{pmatrix} = \begin{pmatrix} \mathcal{I}_{2^n} & 0 \\ 0 & \sigma \end{pmatrix} \tag{2.4}$$

where $\otimes_{j=1}^{n} \sigma_i^{\delta_{i,j}}$ simply picks out the Pauli σ_j from σ and leaves the rest as identity. We use the controlled Pauli operators when performing quantum measurement.

Another important property of the Pauli group is that we can write an implementation $U \in SU(2^n)$ as a rotation generated by Paulis

$$U = e^{i \sum_n \theta_n \sigma_n} \tag{2.5}$$

where σ_n is an enumeration of $\mathcal{P}_n$ and θ_n is degree of rotation for each generator. We make use of this model extensively to create examples, but also to see that $U \in \text{span}(i\mathcal{P}_n)$.

2.2 Quantum Processes

Quantum processes correspond to linear maps, which can be naturally represented by matrices, often known as *process matrices* [35]. We will construct our process matrices for n qubits using an orthonormal basis $\mathbb{B} = \{B_j \mid j \in \mathbb{Z}_{d^2}\}$ for $2^n = d$ dimensional matrices $\mathbb{C}^{d\times d}$ under the Hilbert-Schmidt inner product $\langle A|B\rangle = \text{Tr}\big(A^\dagger B\big)$. By orthonormality and linearity, we have

$$\begin{aligned}\mathcal{E}(\rho) &= \sum_{B\in\mathbb{B}} \langle B|\rho\rangle\, \mathcal{E}(B) \\ &= \sum_{B,B'\in\mathbb{B}} \langle B|\rho\rangle\, \langle B'|\, \mathcal{E}\, |B\rangle\, B'.\end{aligned} \tag{2.6}$$

We therefore represent the density matrix $\rho \in \mathcal{D}(\mathbb{C}^{d\times d}) \subset \mathbb{C}^{d\times d}$ with the column vector $|\rho\rangle$ whose jth entry is $\langle B_j|\rho\rangle$, and a generic linear quantum process $\mathcal{E} \in \mathbb{C}^{d^2\times d^2}$ by a matrix with entries $\mathcal{E}_{j,k} = \langle B_j|\mathcal{E}(B_k)\rangle$. As $U \in SU(2^n)$ is a quantum process, it must have a representation of its action in $\mathbb{C}^{d^2\times d^2}$, we write this action as given by the implementation map $\phi : SU(2^n) \to \mathbb{C}^{d^2\times d^2}$ with $\phi(U)\,|\rho\rangle = \left|U\rho U^\dagger\right\rangle\ \forall U \in SU(2^n)$.

To each noise process $\mathcal{E}$ we associate a fidelity $\mathcal{F}[\mathcal{E}]$ which is used as a figure of merit for a quantum computer with noisy implementation $\theta(G) = \phi(G)\mathcal{E}$. Traditional Randomized Benchmarking [7] established a scalable, robust method of estimating the fidelity with few resource requirements. One can use RB theory to rigorously relate the fidelity to the noise process $\mathcal{E}$

$$\mathcal{F}[\mathcal{E}] = \int \text{Tr}\{|\psi\rangle\langle\psi|\, \mathcal{E}(|\psi\rangle\langle\psi|)\}\ \mathrm{d}\mu(\psi) \tag{2.7}$$

where μ is the Fubini-Study measure over states in the complex projective space $\mathcal{CP}^{d-1}$ [3]. In this paper, much effort will be put into establishing a consistent notion of fidelity at the logical level.

To extract information from a quantum computer, performing a measurement is necessary. We will utilize measurements not only at the end of a sequence, but also during the randomized benchmarking procedure.

Definition 2.4 (Projective Measurement (PM))**.** A set of projectors Q with $\sum_{\Pi_i \in Q} \Pi_i = \mathcal{I}$.

The different Π_i correspond to all the possible outcomes of a measurement defined by Q. Throughout this thesis, we will assume that our measurements are perfect, meaning that measurements can be represented as perfect projections onto the outcome i. If the outcome is known, we update the state using Lüder's rule

$$\bar{\Pi}_i \left|\rho\right\rangle = \frac{1}{\mathrm{Tr}\{\rho\Pi_i\}} \left|\Pi_i \rho \Pi_i\right\rangle . \tag{2.8}$$

When we use projective measures of the form $Q = \{\Pi, I - \Pi\}$, such as in randomized benchmarking experiments, we refer to one outcome as "survival". We will refer to the probability of the outcome Q_s for a sequence length m as the *survival probability*, p_{s_m}.

In quantum computation we utilize a *gate set* $\mathbb{G}$ of unitary maps, which is often a group. We start with an initial state ρ, apply a *gate sequence* $G_m ... G_1$ whose measurement outcome is given by $\langle Q| G_m ... G_1 |\rho\rangle$. However in practice all of the above components will be noisy, represented with the map $\theta : \mathbb{G} \to \mathbb{C}^{d\times d}$. We accept a slight abuse of notation in that we apply θ to preparations and measurements as well as gates to represent *state preparation and measurement* (SPAM) errors. So we denote the outcome of an implemented noisy gate sequence as

$$q_{\vec{G}} = \theta(Q)\theta(G_m)...\theta(G_2)\theta(G_1)\theta(\rho). \tag{2.9}$$

2.3 Representation Theory

This section is meant to serve as a quick representation theory reference; for a more complete description, see [22]. The reader can find additional resources related to RB and representation theory in [9, 17, 33] on which we base this section.

Let V be a vector space of dimension d and then $\mathrm{End}(V)$ is set of linear mappings $V \to V$. A *representation* ϕ of a group $\mathbb{G}$ on $\mathrm{End}(V)$ is a group homomorphism $\phi(G*G') = \phi(G) * \phi(G')$ from $\mathbb{G} \to \mathrm{End}(V)$ Group homomorphisms like ϕ preserve some piece of the group action and cannot make the action on the elements of $\mathbb{G}$ more complex. Call this injective image of $\mathbb{G}$ by the shorthand $\phi(\mathbb{G})$, which forms a subgroup of $\mathrm{End}(V)$.

In this thesis, the vector space V will be all of the $|\rho\rangle$ corresponding to density matrices, and End(V) will be $\mathbb{C}^{d^2\times d^2}$ to represent the process matrices of section 2.2. In this way, the full dynamics of a sequence of quantum processes can be decomposed and analyzed using the tools of representation theory. Below, we give an intuition for one of the most potent tools in representation theory, a general method of decomposing a noiseless implementation of the group element $\phi(G) \in \text{End(V)}$ known as the canonical decomposition.

Suppose there is a subspace W of V with the invariance $\phi(G)w \in W \ \forall w \in W$. In that case, the representation is *reducible*. One could "reduce" this representation into simpler representations $\phi_{\mathbb{H}}$ and $\phi_{\mathbb{H}\perp}$ by restricting the range of ϕ to End(W) and End(V/W) respectively and mapping all other elements to identity. Thus obtaining two *subrepresentations* with underlying subgroups $\mathbb{H}$ and $\mathbb{H}_\perp$, each "representing" a smaller piece of the action of G than ϕ. This decomposition naturally leads to the notion of an irreducible representation or *irrep* as a representation with no invariant subspaces. Schur's famous lemma gives a simple test to see if a representation is irreducible. We utilize a version of Schur's lemma given in [31].

Lemma 2.1 (Schur's Lemma)**.** *If a subrepresentation ϕ is irreducible, then a linear map given by the matrix $M \in \mathfrak{C}(\phi(\mathbb{G}), \text{End}(V))$ is of the form $M = \alpha\mathcal{I}$ for a scalar α.*

Following the above process, one may find that a subspace with the same underlying subgroup appears more than once. Isomorphisms between these subgroups give rise to an equivalency between representations.

Definition 2.5 (Equivalent Representation)**.** If T is a linear map between irreducible representations ϕ and ϕ' such that $T\phi(G) = \phi'(G)T \quad \forall G \in \mathbb{G}$ then $T = 0$ or $\phi \cong \phi'$ are equivalent.

We use definition 2.5 to define a set of equivalence classes for irreps. $\mathfrak{I}$. The number of equivalent copies of the j^{th} irrep. class is it's *multiplicity* m_j. Let's enumerate these equivalent representations with the variable $k < m_j$ in ϕ_j^k. If $m_j > 1$ we call the j^{th} irrep is *degenerate*. Randomized benchmarking procedures which are designed using groups with low numbers of non-degenerate irreps are, in general, more robust. Note that the group $\mathcal{N}(\mathbb{G})$ is irreducible for any $\mathbb{G}$, so $\mathcal{N}(\mathcal{P}_n) = \mathcal{C}_n$ is an intuitive choice for a group to perform RB with.

By following the "reducing" process above we can obtain a *cannonical decomposition* for a representation $\phi(G) \cong \bigoplus_{\mathbb{H}\in\mathfrak{I}} \phi_{\mathbb{H}}(G)$ for $\mathfrak{I}$ the set of all equivalency classes irreps. of $\mathbb{G}$. We'll call a group $\mathbb{H}$ irreducible when it has a single irrep.

Let's do an example with $\mathbb{A}_1$, the single qubit Asymmetric Group defined in section 2.1, showing that the typical 2×2 quantum representation of $\mathbb{A}_1$ has a single non-trivial irrep. with $m_\phi = 1$.

Example 2.1. Using $\mathcal{P}_1$ as a basis for for the state $|\rho\rangle$ allows us to write $|\rho\rangle = \sum_{\sigma \in \mathcal{P}_1} \rho_\sigma |\sigma\rangle$. In this basis, one notes that applying σ only alters the sign of the states $|\sigma\rangle$, so we will focus on applying HP. A quick use of the commutation relations given below eq. (2.2) shows that $\phi(HP)|X\rangle = -|Y\rangle$, $\phi(HP)|Y\rangle = |Z\rangle$, and $\phi(HP)|Z\rangle = |X\rangle$. Thus, all of the non-identity Pauli eigenstates are in the same irreducible subspace. Now, only two subspaces remain, the trivial subspace given by the vector space spanned by $|\mathcal{I}\rangle$ and a non-trivial subspace which must include all Pauli eigenstates. From this, we conclude that either the two subspaces are equivalent, or their corresponding representations are irreducible. The two subspaces must be inequivalent because the trivial irrep only has trivial action in its underlying subgroup. So both representations are irreducible.

We define the projector onto a particular subrepresentation given by ϕ as the projector onto the equivalent representations of ϕ which all have the *character* $\chi_\phi = \mathrm{Tr}(\phi(G))$ as

$$\Pi_\phi = \mathbb{E}_{G \in \mathbb{G}} \phi(G) \chi_\phi(G). \tag{2.10}$$

To project onto spaces representing individual equivalent representations, we use a more sophisticated tool from the supplementary material of [9]. In order for the j^{th} irrep to have multiplicity m_j, by definition 2.5 there must exist m_j^2 linear maps between the equivalent irreps. For the j^{th} irrep, label these linear maps $Q_j^{k,l}$ where $1 \leq k, l \leq m_j$. Arrange the $Q_j^{k,l}$ as matrix blocks labeled by j. Then we can diagonalize $Q_j^{k,k}$ in the basis spanned by $|v_{kn}\rangle$. Finally, arrange the Q_j as matrix blocks so that $Q_j^{k,l} = \sum_{n=1}^{m_j} \Big(|j\rangle \otimes |v_{kn}\rangle \Big) \Big(\langle j| \otimes \langle v_{ln}| \Big)$ so that $Q_j^{k,l}$ is a projector onto $\phi^k_{\mathbb{H}_j}$ when $k = l$ and maps between spaces spanned by the the l and kth equivalent rep. of $\mathbb{H}_j \in \mathcal{G}_\mathcal{I}$ when $k \neq l$.

Lemma 2.2 (Properties of $Q_j^{k,l}$ [9])**.** *$Q_j^{k,k} \phi(G) = \phi^k_{\mathbb{H}_j}(G) Q_j^{k,k}$ and $Q_j^{k,l} Q_{j'}^{k',l'} = \delta_{j,j'} Q_j^{k,l} Q_{j'}^{k',l'}$*

We use the above concept to prove the theorem below, which is based on the work in the supplementary material of [9].

Theorem 2.6. *If a matrix $M \in \mathfrak{C}(\phi(\mathbb{G}), \mathrm{End}(V))$ then $\exists M_j$ with rank m_j s.t. $M = \sum_{j=1}^{|G_\mathcal{I}|} M_j$ and $M_j M_{j'} = \delta_{j,j'} M_j^2$.*

Proof.

$$\phi(G)M = M\phi(G) \qquad ([\phi(G), M] = 0) \tag{2.11}$$

$$Q_j^{k,k}\phi(G)MQ_{j'}^{l,l} = Q_j^{k,k}M\phi(G)Q_{j'}^{l,l} \tag{2.12}$$

$$\phi^k_{\mathbb{H}_j}(G)Q_j^{k,k}MQ_{j'}^{l,l} = Q_j^{k,k}MQ_{j'}^{l,l}\phi^l_{\mathbb{H}_{j'}}(G). \qquad \text{(lemma 2.2)} \tag{2.13}$$

Next, we use definition 2.5 where $T = Q_j^{k,k}MQ_{j'}^{l,l}$ showing $\phi^k_{\mathbb{H}_j}$ is equivalent to $\phi^k_{\mathbb{H}_{j'}}$

$$\phi^k_{\mathbb{H}_j}(G)Q_j^{k,k}MQ_{j'}^{l,l} = \delta_{j,j'}Q_j^{k,k}MQ_{j'}^{l,l}\phi^l_{\mathbb{H}_{j'}}(G). \qquad \text{(definition 2.5)} \tag{2.14}$$

We can use the above to show

$$Q_j^{k,k}MQ_{j'}^{l,l} = \delta_{j,j'}\alpha_{j,k,l}Q_j^{k,l}. \qquad \text{(lemma 2.1)} \tag{2.15}$$

To be clear, lemma 2.1 is used above when $k = l$ otherwise eq. (2.15) follows from the definition of $Q_j^{k,l}$.

We sum over all the projectors onto irreps $Q_j^{k,k}$ in eq. (2.15) to obtain

$$M = \sum_{j=1}^{|\mathfrak{I}|}\sum_{k,l=1}^{m_j}\alpha_{j,k,l}Q_j^{k,l}. \tag{2.16}$$

If $M_j := \sum_{k,l=1}^{m_j}\alpha_{j,k,l}Q_j^{k,l}$ then the desired property for M_j is checked using lemma 2.2. □

Corrolary 2.3. *If* $M \in \mathfrak{C}(\phi(\mathbb{G}), \mathrm{End}(V))$ $\exists M_j$ *with rank* m_j *s.t.* $M^m = \sum_{j=1}^{|\mathfrak{I}|} M_j^m$.

Corrolary 2.4. *If a finite set* $\mathcal{M} \subseteq \mathfrak{C}(\phi(\mathbb{G}), \mathrm{End}(V))$ *then* $\forall M_i \in \mathcal{M}$ $\exists M_{i,j}$ *with rank* m_j *s.t.* $M_i = \sum_{j=1}^{|\mathfrak{I}|} M_{i,j}$ *and* $M_{i,j}M_{i',j'} = \delta_{jj'}M_{i,j}M_{i',j'}$ *by defining* $M_{i,j} := \sum_{k,l=1}^{m_j}\alpha_{i,j,k,l}Q_j^{k,l}$ *similarly to theorem 2.6. One may ignore cross-terms in a product over* i *to obtain*

$$\prod_{i=1}^{|\mathcal{M}|} M_i = \prod_{i=1}^{|\mathcal{M}|}\sum_{j=1}^{|\mathfrak{I}|} M_{i,j} = \sum_{j=1}^{|\mathfrak{I}|}\prod_{i=1}^{|\mathcal{M}|} M_{i,j}. \tag{2.17}$$

2.4 Summary

We reviewed some general groups utilized by quantum computation and pointed out some of their properties.

1. The Pauli group $\mathcal{P}_n$ can be used to generate rotations in $SU(2^n)$.
2. The Clifford group $\mathcal{N}(\mathcal{P}_n) = \mathcal{C}_n$ and the single-qubit asymmetric group $\mathbb{A}_1$ are both usable for randomized benchmarking.

Quantum process were discussed in the context of gate sequences with perfect measurement

1. Quantum processes are maps in $\mathbb{C}^{d^2 \times d^2}$ between density matrices in $\mathcal{D}(\mathbb{C}^{d \times d})$.
2. In quantum computation, we begin in a state $|\rho\rangle$ and enact unitaries $\phi(U)\,|\rho\rangle = \left|U\rho U^\dagger\right\rangle$, perform a gate sequence $\vec{G}$, and measure Q to get the outcome $q_{\vec{G}}$.
3. We use Lüder's rule $\bar{\Pi}\,|\rho\rangle = \frac{1}{\mathrm{Tr}\{\rho\Pi_i\}}\,|\Pi\rho\Pi\rangle$ to describe a measurement with known outcome Π.

We realised that one could evaluate quantum processes in $\mathbb{C}^{d^2 \times d^2}$ using the tools of representation theory

1. An errorless implementation of a group is a representation $\phi(G)$, which we decompose into the cannonical decomposition $\phi(G) \cong \bigoplus_{\mathbb{H} \in \mathfrak{J}} \phi_{\mathbb{H}}(G)$,
2. Corrolary 2.3 and corrolary 2.4 showed that representation theory can be used to simplify products of quantum processes of $\mathcal{M} \subset \mathfrak{C}(\phi(\mathbb{G}), \mathrm{End}(V))$ namely for matrices $M, M_i \in \mathcal{M}$ we have the identities $M^m = \sum_{j=1}^{|\mathfrak{J}|} M_j^m$ and $\prod_{i=1}^{|\mathcal{M}|} M_i = \sum_{j=1}^{|\mathfrak{J}|} \prod_{i=1}^{|\mathcal{M}|} M_{i,j}$.

Chapter 3

Quantum Error Correction

As with classical computers, it is possible to use redundancy to protect computation on a subspace. Over the past 20 years *Quantum Error Correction* (QEC) has proven it has the potential to suppress errors in long computations with the discovery and development of fault tolerant noise thresholds [10, 30, 34]. If the noise rate is below this threshold, errors are suppressed for arbitrarily long computation. We focus on a ubiquitous subfield of quantum error correction, the stabilizer formalism.

3.1 Stabilizer Formalism

The stabilizer formalism is a tool that describes a restricted set of quantum states known as stabilizer states [1, 10] which are extensively applied in the study of quantum error-correcting codes [12]. We study this formalism in the context of randomized benchmarking. We search for possible benchmarks for a diverse ensemble of stabilizer codes.

To specify an $[[n, k]]$ stabilizer code, we define a *stabilizer group* which is an Abelian subgroup $\mathbb{S} \subsetneq \mathcal{P}_n$ with k generators that does not contain $-I$. We enumerate the generators $\mathbb{S}^{gen} = \{S_1^{gen}, \ldots, S_{n-k}^{gen}\}$ and an invariant subspace of dimension 2^k which is referred to as the *code space*. The projector onto the code space is given by

$$\Pi_{\mathbb{S}} = 2^{-n} \sum_{S \in \mathbb{S}} S. \tag{3.1}$$

Equivalently, the code space is the set of all $\left|\bar{\psi}\right\rangle$ such that $S\left|\bar{\psi}\right\rangle = \left|\bar{\psi}\right\rangle$ for all $S \in \mathbb{S}$. We abuse notation a bit in that we also apply $\Pi_{\mathbb{S}}\left|\rho\right\rangle = \left|\Pi_{\mathbb{S}}\rho\Pi_{\mathbb{S}}\right\rangle$ to denote a projection onto the code space. The difference will be clear from context.

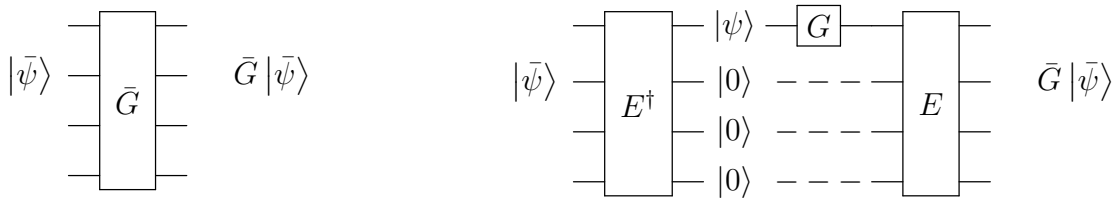


Figure 3.1: Copies of the same circuit implementing the logical gate $\bar{G}$. On the RHS, we can see that the encoder E defines a frame that separates the state's syndrome and logical portions. In this "Unencoded" frame, it is clear that we may, in principle, measure the logical qubits ($|\psi\rangle$ above) and syndrome qubits ($|\vec{0}\rangle$ below) separately. Dashed lines on the RHS remind the reader that the implementation of $\bar{G}$ will be imperfect, so it is likely the noisy implementation of $\bar{G}$ will act to create a non-trivial syndrome.

There exists a Clifford operator E with $EZ_iE^\dagger = S_i^{gen}\ \forall i > k$, commonly called the *encoder* because it maps $E(|\psi\rangle \otimes |\vec{0}\rangle) = |\bar{\psi}\rangle$ [10]. We have illustrated in fig. 3.1 how one may use E to define two "frames" from which one can view the quantum state. We will make use of the unencoded frame defined in fig. 3.1 often.

Secondly, we have the *Logical Group* ($\mathbb{L}$), defined by Pauli operations on the logical qubit in a frame given by E

$$\mathbb{L} = \{E(\sigma \otimes \mathcal{I}_{n-k})E^\dagger \mid \forall \sigma \in \mathcal{P}^k\}.$$

These are the important gates which one uses for computation in stabilizer codes. Our primary interest in performing a benchmark of a stabilizer code will be to characterize the noise afflicting these operators. Last comes the destabilizers.

Definition 3.1 (Destabilizers)**.** The Abelian group generated by

$$\mathbb{T}^{gen} = \{EX_iE^\dagger \mid \forall\ i > k\} \tag{3.2}$$

The general role of destabilizers is to bring us out of the code space without affecting the logical state. When we apply quantum codes in a randomized benchmarking setting, destabilizers cause the syndrome qubits in the RHS of fig. 3.1 to flip.

We can express any Pauli $\sigma \in \mathcal{P}_n$ as

$$\sigma = LST \tag{3.3}$$

for $S \in \mathbb{S}, L \in \mathbb{L}, T \in \mathbb{T}$ [25].

The above definitions suggest there is a particular frame defined by E wherein we can treat logical action as separate from the action on the rest of the system. In fig. 3.1 we make this notion precise by relegating all of the logical action to a single qubit.

Figure 3.1 also implies that action on the syndrome and logical states can commute and that we can, in fact, treat them as separate qubit systems. We call them the *logical and syndrome qubits* respectively. This structure allows us to make measurements without decohering any quantum correlations which might occur in a circuit made from elements of span$\{\mathbb{L}\}$.

Definition 3.2 (Logical and Syndrome Operator Spaces)**.** The operator space defined by the process matrices in span$\{\phi(\mathbb{L})\}$ is the logical space where $\phi(L)$ is a noiseless implementation of L. The syndrome operator space can be defined analogously with $\mathbb{ST}$.

It is worthwhile to note the distinction between the logical, syndrome, and code spaces. The operators in the logical and syndrome spaces specify the evolution of qubits in the unencoded frame. The code space is the set of states such that $S|\psi\rangle = |\psi\rangle \ \forall S \in \mathbb{S}$.

We highlight an important group used commonly to assess logical gates.

Definition 3.3 (Logical Clifford Group)**.** The logical Cliffords can be defined as the group $\mathcal{C}_L = \{E(\mathcal{C} \otimes \mathcal{I}_{n-k})E^\dagger | C \in \mathcal{C}_k\}$ by considering $\mathcal{C}_L$ in the unencoded frame of fig. 3.1.

The $\mathcal{C}_L$ is a group commonly cited [5] as a simple 2-design on the logical space. Generally, we want the logical gate implementation to be transversal to simplify noise in computation. However, the logical Cliffords are not guaranteed to be transversal in more compact codes such as the [[5,1]] code. Therefore in chapter 7 we resort to using another 2-design on the logical operators, the *logical asymmetric group* ($\mathbb{A}_L$) which cannot be expressed so quickly in the unencoded frame. However, we will see the commutation relations with X_L and Z_L are preserved, so $\mathbb{A}_L$ still forms the group on the logical bit in the unencoded frame.

3.2 Stabilizer QEC

It is widely recognized that quantum measurements and operators do not commute in general. Consequently, one cannot measure any operator on the underlying quantum state to check for the presence of errors without destroying the prepared state. Stabilizer codes circumvent this problem by using $[L, ST] = 0 \ \forall L, S, T$ so that one can, in principle, measure $\mathbb{S}$ or $\mathbb{T}$ without disturbing the measurement outcome on the logical space. However, we

choose to measure the eigenbasis of the stabilizers by convention. Stabilizer measurement is done by performing a parity check [30] measurement of the generators of the stabilizer group with a Non-Neumann measurement as in fig. 3.2.

The outcomes of measuring the stabilizer generators S_i^{gen} are denoted by a syndrome $s \in \mathbb{Z}_2^{n-k}$. An encoded state has a trivial syndrome $|\vec{0}\rangle$. We will consider two possible methods of addressing the presence of noise, informed by the syndrome. The first is by simply discarding the state whenever its syndrome is non-trivial. This is called *post-selection* and we typically represent discarding post-selected sequences by a state update inspired by Lüder's rule

$$\bar{\Pi}_{\mathbb{S}} |\rho\rangle = \frac{1}{\mathrm{Tr}\{\rho\Pi_{\mathbb{S}}\}} |\Pi_{\mathbb{S}}\rho\Pi_{\mathbb{S}}\rangle \tag{3.4}$$

Second, by applying *quantum error correction* (QEC), an operation which is chosen to undo the effect of noise. One performs QEC using a Pauli decoding $D_s \in \mathcal{P}_n$ that applies the most likely error pattern consistent with the measurement syndrome.

Note that D_s can be decomposed in the form of eq. (3.3) as $D_s = T_s L_s S_s$, where $T_s \in \mathbb{T}$ triggers the syndrome s, while $L \in \mathbb{L}$ and $S \in \mathbb{S}$. We call a decoding trivial if $L_s = \mathcal{I}\ \forall s$ and denote the resulting operation with $\mathcal{R}^0$. We will encapsulate the syndrome measurement and the decoding step D_s for every syndrome measurement outcome s using a single quantum process referred to as the *QEC map* or *recovery map* $\mathcal{R} = \sum_s R_s$. If we are allowing for either post-selection or QEC, then $\mathcal{R}$ is called an *error reduction map*.

A key model for $\mathcal{R}$ which we utilize extensively is *perfect measurement*, given by

$$\mathcal{R}_{PM} = \sum_{s=0}^{2^{n-k}-1} \mathcal{E}_s \phi(D_s) \Pi_{\mathbb{S}} \phi(T_s) \tag{3.5}$$

where ϕ is the representation for quantum processes used by the computer. Put simply, this model assumes stabilizer measurement always results in an eigenstate of $\mathbb{S}$. A more strict assumption, known as *perfect recovery* [4, 5, 27] consists of a perfect measurement and ensures the state ends up back in the code space

$$\mathcal{R}_{PR} = \sum_{s=0}^{2^{n-k}-1} \mathcal{E}_s \phi(D_s) \bar{\Pi}_{\mathbb{S}}. \tag{3.6}$$

Note $\mathcal{R}_{PR} = \mathcal{R}_{PR}\Pi_{\mathbb{S}}$ acts as a projector from the right. We show that how these two assumptions are applied is crucial to determining when protocols are robust to gate-dependent noise.

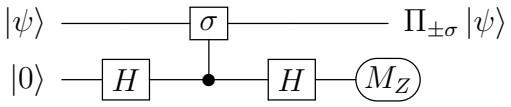


Figure 3.2: A model of Von-Neumann measurement useful for QEC. The Pauli observable σ is extracted from $|\psi\rangle$ using C_σ defined in eq. (2.4), an ancilla qubit, and a Z measurement. This circuit asks "What part of $|\psi\rangle$ lies along an eigenstate of σ?" and stores that in the ancilla to be measured by M_Z. In quantum error correction, we substitute σ for and S^{gen} and attempt to reduce the error using the result of M_z to post-select or apply QEC. The projector $\Pi_{\pm\sigma}$ is the projector onto the plus and minus eigenstates of σ.

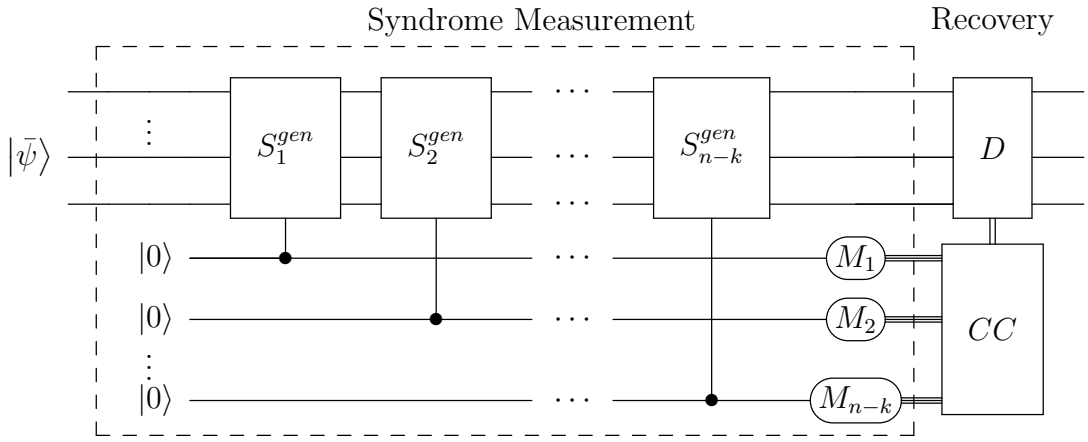


Figure 3.3: We combine a syndrome measurement and decoding step to make a QEC map in the stabilizer formalism. We first perform a Von-Neumann measurement using $C_{S^{gens}}$ and follow it with a classically controlled operator (or a post-selection). The block CC shows the process of "classically computing" a decoding to invert errors associated with the measured syndrome. The block labelled D is the applied decoding.

Definition 3.4 (Error Reduction)**.** We use the symbol $\mathcal{R}$ to denote a syndrome measurement followed by either a post-selection or recovery, i.e., $\mathcal{R}$ is a method of using the code to reduce errors in the system. When $\mathcal{R}$ uses a decoding, $\mathcal{R}$ will be a QEC map that we will specify as 1 of 2 forms: perfect measurement defined in eq. (3.5) and perfect recovery defined in eq. (3.6). We represent syndrome measurement and post-selection with the projector onto the code space $\Pi_{\mathbb{S}}$. However, we may require that Π be normalized. In which case the projector will be written $\bar{\Pi}_{\mathbb{S}}$ as in eq. (3.4). We call $\mathcal{R}$ and error reduction map.

3.3 Fault Tolerance

We give a brief flavor for the diverse and developing field of fault tolerance. For a more complete set of approaches, see [11, 20, 30].

So far, we have only considered scenarios where we have a noisy set of quantum gates. In a realistic scenario, one expects that the post-selection or QEC step $\mathcal{R}$ will be noisy as well. It would be unrealistic to assume that noise on this measurement is small, however [10] showed that for some simplistic noise models $\mathcal{R}$ does not increase the size of errors when the physical noise rate is below a certain threshold. Thus yielding errorless or *fault tolerant* computation.

Traditionally, the diamond distance has been touted as the proper figure of merit to look at if one wishes to guarantee fault tolerant computing [18]. As such, there has been much interest in measuring the diamond distance using robust experimental protocols such as RB [28, 33]. The ultimate goal of this pursuit would be to design a protocol characterizing the physical qubits which indicates fault tolerance is achieved in the ensemble. A recent breakthrough to bound the diamond norm shows that these fault tolerance certifications for physical qubits are perhaps on the horizon [6].

However, this approach suffers from 3 major technical issues, which can be attributed to the large size of the physical system and it's many parameters [18]. First, the diamond norm is only rigorously shown to give fault tolerance guarantees under simplistic noise models [2]. Second, the diamond norm is a worst-case measure of error rates, which may greatly overestimate the presence of errors in the system to ensure fault tolerance is achieved. Finally, optimizing a decoding analytically is hard in general [19].

Optimizing at the logical level does away with the need to characterize the many parameters required to describe noise at the physical level. Thus, we choose to forgo a physical characterization of the noise and design experiments that measure error rates directly at the logical level [18].

3.4 Summary

An $[[n,k]]$ stabilizer code is a 2^k dimensional eigenspace with eigenvalue $+1$ for a set of $n-k$ commuting Pauli operators: $\mathbb{S}^{gen} = \{S_1^{gen}, \ldots, S_{n-k}^{gen}\}$.

1. The code-space is the set $\{|\overline{\psi}\rangle \in \mathcal{H}_2^n : S_i|\psi\rangle = |\psi\rangle\}$ with projector $\Pi_{\mathbb{S}} = \sum_{S\in\mathbb{S}} S$.
2. Each of the encoded states can be prepared using a unitary circuit E called the encoder: $E|\psi\rangle \otimes |0\rangle^{n-k} = |\overline{\psi}\rangle$. We can use E to define the following groups:
   (a) $\mathbb{L}$ the logical group given by $\mathbb{L} = \{E(\sigma \otimes \mathcal{I}_{n-k})E^\dagger \mid \forall \sigma \in \mathcal{P}^k\}$.
   (b) $\mathbb{S}$ the stabilizer group generated by $\mathbb{S}^{gen} = \{EZ_iE^\dagger \mid \forall\ i > k\}$ and $S\left|\bar{\psi}\right\rangle = \left|\bar{\psi}\right\rangle$.
   (c) $\mathbb{T}$ the destabilizer group generated by $\mathbb{T}^{gen} = \{EX_iE^\dagger \mid \forall\ i > k\}$.
   (d) $\mathcal{C}_L$ the logical Clifford group $\mathcal{C}_L = \{E(C \otimes \mathcal{I}_{n-k})E^\dagger | C \in \mathcal{C}_k\}$.
   (e) $\mathbb{A}_L$ the single-qubit logical asymmetric group.
3. We can decompose a n-quit Pauli $\sigma \in \mathcal{P}_n$ as $\sigma = LST$ with $L \in \mathbb{L}$, $S \in \mathbb{S}$, and $T \in \mathbb{T}$,

The error reduction operator $\mathcal{R}$ is a 2 fold process

1. Measure in the eigenbasis of $\mathbb{S}$ to obtain a syndrome.
2. Next we pick one of the two following responses to the syndrome
   (a) Post-selection allows one to discard the outcome when the syndrome is non-trivial. In this case, we may write $\mathcal{R}$ as $\Pi_{\mathbb{S}} = 2^{-n}\sum_{S\in\mathbb{S}} S$ or $\bar{\Pi}_{\mathbb{S}} = \frac{\Pi_{\mathbb{S}}}{\mathrm{Tr}\{\rho\Pi_{\mathbb{S}}\}}$ as in eq. (3.4).
   (b) Qunatum error correction uses the information from the syndrome to enact a Pauli decoding D_s. We use 2 forms for a noisy $\mathcal{R}$:
      i. Perfect Recovery assumes $|\rho\rangle$ always ends up back in the code space given by $\mathcal{R}_{PR} = \sum_{s=0}^{2^{n-k}-1} \mathcal{E}_s\phi(D_s)\bar{\Pi}_{\mathbb{S}}$
      ii. Perfect measurement assumes we end up in an eigenstate of $\mathbb{S}$ given by $\mathcal{R}_{PM} = \sum_{s=0}^{2^{n-k}-1} \mathcal{E}_s\phi(D_s)\Pi_{\mathbb{S}}\phi(T_s)$.

Fault tolerance guarantees that if the combined error rates of the QEC subroutine and gate implementation are below a certain threshold, computation is effectively errorless.

1. Although fault tolerance thresholds can be shown analytically for simplistic noise models, for a general noise model, implementation is needed to characterize fault tolerance.

2. We advocate for measuring logical error rates directly to optimize implementations of QEC.

Chapter 4

Randomized Benchmarking

Informally, Randomized Benchmarking (RB) is an efficient class of protocols for generating figures of merit for noisy computation. Extensions of RB [5, 8, 9, 15, 23, 26] demonstrate that RB is a versatile tool for quantifying the performance of quantum computers.

In this thesis, we survey options for expanding randomized benchmarking to measure error rates on logical operators directly. Our survey will review two previous protocols that used RB to characterize logical error rates. We later use character benchmarking to generalize these protocols.

4.1 Character Benchmarking

Somewhat abstractly, experimental quantum computation aims to develop systems that implement an approximate representation of the unitary group [13, 24]. One can formalize this idea by defining a matrix-valued function θ that maps ideal operations, which form a subrepresentation of the unitary group, to their noisy implementation $\theta(G)$ for $G \in \mathbb{G}$. From this perspective, Randomized Benchmarking (RB) algorithms are natural benchmarks because they experimentally measure the probability of an error occurring using a subrepresentation of $\mathbb{G}$.

Character benchmarking is a general, robust framework for a wide variety of RB procedures [17]. In section 5.1 we utilize character benchmarking to define a broad class of figures of merit for stabilizer codes. We abuse notation somewhat in that we apply θ to preparations and measurements as well as unitary operations. algorithm 4.1 details the character benchmarking procedure.

Algorithm 4.1: Character Benchmarking Procedure

Input: $\mathbb{G}$ - implemented group, ϕ - irrep. with multiplicity 1,
ρ - input state, Q - $\{Q_s, Q_d\}$ a projective measurement,
M - list of sequence lengths

Output: $\mathcal{F} \in [0, 1]$ - Average Fidelity

for $m \in M$ **do**
  repeat
    repeat
      Prepare the input state $\rho = Q_s$.
      for $1 \leq i \leq m$ **do**
        Sample $G_i \in \mathbb{G}$ uniformly.
        Classically compute $G_i G_{i-1}^{\dagger}$ where $G_0 = \mathcal{I}$.
        Apply noisy implementation $\theta(G_i G_{i-1}^{\dagger})$ to the state.
      Sample $G_{\Pi} \in \mathbb{G}$ uniformly.
      Apply the noisy gate $\theta(G_{\Pi} G_m^{\dagger})$ to the state.
      Measure the projective measurement Q.
      Record outcome $q_{\vec{G}}^{RB} \in \{Q_s, Q_d\}$.
    until *desired number of* $q_{\vec{G}}^{RB}$ *are obtained*;
    Record survival probability $p_{s_m}^{RB}$ as the average number of Q_s.
    Weight $p_{s_m}^{RB}$ using the character $\chi_{\phi}(G_{\Pi})$.
  until *desired number of* $p_{s_m}^{RB}$ *are obtained*;
Fit the decay $A\lambda^m + B$ to the many estimates of $p_{s_m}^{RB}$.
return $\mathcal{F} = \mathrm{Tr}[Q_d]\lambda + \mathrm{Tr}[Q_s]$.

The *character benchmarking sequence* $q^{RB}_{\vec{G}}$ for a sequence of gates $\vec{G} = (G_\Pi, G_m, \ldots, G_1)$ is given by

$$q^{RB}_{\vec{G}} = \theta(Q)\theta(G_\Pi G_m^\dagger)\prod_{i=2}^{m}\theta(G_i G_{i-1}^\dagger)\theta(G_1)\theta(\rho). \tag{4.1}$$

For simplicity, we assume $\theta(G) = \phi(G)\mathcal{E}$ for some representation ϕ of $\mathbb{G}$ and noise process $\mathcal{E}$. We expand the sequence using the definition of a representation to derive

$$q^{RB}_{\vec{G}} = \theta(Q)\phi(G_\Pi)\phi(G_m^\dagger)\mathcal{E}\prod_{i=2}^{m}\phi(G_i)\phi(G_{i-1}^\dagger)\mathcal{E}\phi(G_1)\mathcal{E}\theta(\rho). \tag{4.2}$$

Definition 4.1 (Twirled Noise)**.**

$$\mathcal{T}_\mathbb{G}[\mathcal{E}] = \sum_{G\in\mathbb{G}}\phi(G^\dagger)\mathcal{E}\phi(G). \tag{4.3}$$

Lemma 4.1 (Rotating the Twirl)**.** $\mathcal{T}_\mathbb{G}[\mathcal{E}] \in \mathfrak{C}(\phi(\mathbb{G}), \mathbb{C}^{d^2\times d^2})$.

Proof.

$$\begin{aligned}
\phi(G)\mathcal{T}_\mathbb{G}[\mathcal{E}] &= \sum_{G_1\in\mathbb{G}}\phi(GG_1^\dagger)\mathcal{E}\phi(G_1) && \text{(definition of a rep.)} && (4.4)\\
&= \sum_{G_2\in\mathbb{G}}\phi(G_2^\dagger)\mathcal{E}\phi(G_2 G) && \text{(define } G_2 := GG_1^\dagger) && (4.5)\\
&= \mathcal{T}_\mathbb{G}[\mathcal{E}]\phi(G). && && (4.6)
\end{aligned}$$

And thus we have

$$[\mathcal{T}_\mathbb{G}[\mathcal{E}], \phi(G)] = 0. \tag{4.7}$$

□

The above form for the character benchmarking sequence then naturally breaks up into a series of twirled noise models when weighted expectation is taken

$$p^{RB}_{s_m} = \mathbb{E}_{\vec{G}}\left[q^{RB}_{\vec{G}}\chi_\phi(G_\Pi)\right] = \theta(Q)\Pi_\phi\mathcal{T}_\mathbb{G}[\mathcal{E}]^m\mathcal{E}\theta(\rho). \tag{4.8}$$

Where $p^{RB}_{s_m}$ is the survival probability for an RB sequence and Π_ϕ is from eq. (2.10).

Together eq. (4.8) and lemma 4.1 suggest that one can interpret randomized benchmarking with gate-independent noise as a method of simulating the twirled noise map. A natural projection of the noise process onto $\mathfrak{C}(\phi(\mathbb{G}), \mathbb{C}^{d^2\times d^2})$. These symmetrized noise maps are desirable because they compose predictably, making them easier to study than general quantum processes, which may cohere in the degrees of freedom omitted from the twirled noise.

We can simplify eq. (4.8) slightly by re-defining $\mathcal{E}\theta(\rho) \to \theta(\rho)$. Using corrolary 2.3 and lemma 4.1 we can re-write eq. (4.8) as

$$p_{s_m}^{RB} = \sum_{j=1}^{|\mathbb{G}_{\mathcal{I}}|} \theta(Q)\Pi_\phi \mathcal{T}_{\mathbb{G}}[\mathcal{E}]_j^m \theta(\rho) \tag{4.9}$$

where the $\mathcal{T}_{\mathbb{G}}[\mathcal{E}]_j^m$ is the restriction of $\mathcal{T}_{\mathbb{G}}[\mathcal{E}]^m$ onto the span of the jth irrep. The property $\mathcal{T}_{\mathbb{G}}[\mathcal{E}]_j \mathcal{T}_{\mathbb{G}}[\mathcal{E}]_{j'} = \delta_{i,j}\mathcal{T}_{\mathbb{G}}[\mathcal{E}]_j^2$. Π_ϕ picks out the particular $\mathcal{T}_{\mathbb{G}}[\mathcal{E}]_j^m$ corresponding to ϕ which we write as

$$p_{s_m}^{RB} = \theta(Q)\mathcal{T}_{\mathbb{G}}[\mathcal{E}]_\phi^m \theta(\rho). \tag{4.10}$$

Recall from section 2.3 the rank of the matrix $\mathcal{T}_{\mathbb{G}}[\mathcal{E}]_\phi$ is determined by m_ϕ. So if we take $m_\phi = 1$ then $\mathcal{T}_{\mathbb{G}}[\mathcal{E}]_\phi = \lambda\Pi_\phi$ where λ must be in $[0, 1]$ because if λ were complex or negative, then the physical probability $p_{s_m}^{RB}$ would be complex or negative and if $\lambda > 1$ then $p_{s_m}^{RB} \to \infty$ as $m \to \infty$. Because $\lambda \in [0, 1]$ we can guarantee that $p_{s_m}^{RB}$ will decrease monotonically as m increases. We will see in the next section that this property is exceedingly important when evaluating figures of merit are meaningful.

4.2 When does RB give Useful Figures of Merit?

If one sees $\mathbb{E}_{\vec{G}} q_{\vec{G}}^{RB}$ increase or *uptick* one can ascertain $m_\phi > 1$ so that ϕ has multiple copies or $\theta \neq \mathcal{E}\phi(G)$. In sections 6.1 and 6.2 we give examples which suggest the former can happen in LRB-Like sequences and that the latter can be observed in a protocol which minimizes the number of syndrome measurements.

Upticks indicate that the noise $\mathcal{E}$ is non-Markovian [33]. Loosely, non-Markovian noise can be thought of as introducing temporal correlations to the state [29]. So if survival probability increases at step i, the state is more correlated to the initial state than at step $i-1$. Thus the composition of all noisy gates up to that point must be non-Markovian, so at least one of the $\mathcal{E}_i$ is non-Markovian.

Traditionally, upticks are mitigated by guaranteeing that $\mathbb{G}$ is a 2-design over $SU(2^n)$ [7]. For the sake of completeness, we note that a unitary t-design is a set of unitary operators $\mathbb{G}$ satisfying

$$\frac{1}{|\mathbb{G}|}\sum_{G\in\mathbb{G}} P_t(G) = \int_{SU(2^n)} \mathrm{d}\eta(U) P_t(U) \tag{4.11}$$

where P_t is a polynomial in U and $\bar{U}$ of order t, η is the Haar measure, and the integral goes over all unitary operators [3]. However, character benchmarking makes the condition in eq. (4.11) more flexible by directly considering representations like above. Moreover, we can use the ideas in [33] to make an even stronger statement.

Theorem 4.2. *The survival probability of a character benchmarking sequence $p_{s_m}^{RB}$ is monotone decreasing when $\theta(G) = \phi(G)\mathcal{E}_i$, the multiplicity $m_\phi = 1$, and $\mathcal{E}_i$ is a time-dependent linear map acting identically for each instance of $q_{\vec{G}}^{RB}$ in $p_{s_m}^{RB}$.*

Proof. Repeat the derivation in section 4.1 with $\mathcal{E} \to \mathcal{E}_i$. In this case eq. (4.8) becomes

$$p_{s_m}^{RB} = \theta(Q)\Pi_\phi \prod_{i=1}^{m} \mathcal{T}_{\mathbb{G}}[\mathcal{E}_i]\mathcal{E}_0\theta(\rho). \tag{4.12}$$

Then we re-define $\mathcal{E}_0\theta(\rho) \to \theta(\rho)$. Apply corrolary 2.4 instead with $M_i = \mathcal{T}_{\mathbb{G}}[\mathcal{E}_i]$ being decomposed into $M_{i,j} := \mathcal{T}_{\mathbb{G}}[\mathcal{E}_i]_j$ with the property $\mathcal{T}_{\mathbb{G}}[\mathcal{E}_i] = \sum_{j=1}^{|\mathbb{G}_\mathcal{I}|} \mathcal{T}_{\mathbb{G}}[\mathcal{E}_i]_j$ and we have

$$p_{s_m}^{RB} = \sum_{j=1}^{|\mathbb{G}_\mathcal{I}|} \theta(Q)\Pi_\phi \prod_{i=1}^{m} \mathcal{T}_{\mathbb{G}}[\mathcal{E}_i]_j\theta(\rho). \tag{4.13}$$

Again, Π_ϕ picks out the subspace corresponding to ϕ from the twirl

$$p_{s_m}^{RB} = \theta(Q) \prod_{i=1}^{m} \mathcal{T}_{\mathbb{G}}[\mathcal{E}_i]_\phi\theta(\rho). \tag{4.14}$$

Now the argument that $\mathcal{T}_{\mathbb{G}}[\mathcal{E}]_\phi \in [0,1]$ can be applied to each $\mathcal{T}_{\mathbb{G}}[\mathcal{E}_i]_\phi$ as they all must be physically measurable quantities. □

With care, one can show $p_{s_m}^{RB}$ decreases monotonically. Even with heavy gate dependence on θ [17]. However, the gate-dependent noise robustness proof of [17] cannot be applied when the noise model also depends on the current quantum state, as the decoding

does in a QEC map. If one has gate-dependent noise, we show in section 6.1 the current syndrome can encode information about previous gates, which noisy QEC can use to create upticks. Thus, one has to treat $\mathcal{E}$ as if it depends on many previous gates $\mathcal{E}_{\vec{G}}$, not just the most recently applied gate $\mathcal{E}_G$. We show that these non-Markovian memory effects in the syndrome become relevant when QEC maps are inserted into the character benchmarking sequence in eq. (4.1).

4.3 Previous Applications of RB to QEC

We review two previous papers that utilize randomized benchmarking to assess stabilizer codes. In section 4.3.1 we review a resource-intense protocol that is meant to be a robust assessment of a code. In section 4.3.2 we review the second protocol, which is implementable on current quantum computers.

We have also chosen here to omit some details regarding reference sequences. Both procedures utilize logical gate sequences without syndrome measurements as a reference for how much the recovery aids the fidelity. We ignore these robust subroutines because the sequence reduces to a character benchmarking sequence.

4.3.1 Logical Randomized Benchmarking

Logical randomized benchmarking (LRB) is an intuitive, recourse-intensive experiment that applies randomized benchmarking to quantum codes. Combes et al. achieved this characterization by including a QEC map with every gate [5]. Algorithm 4.2 gives a concrete description of the protocol. By viewing the system in the unencoded frame defined in fig. 3.1, one can see that the 2-design property carries over nicely to the logical scenario to form a logical 2-design. The usual choice for $\mathbb{G}$ is the logical Cliffords $\mathcal{C}_L$, which we will use as a default group when applying randomized benchmarking to quantum codes.

Combes et al. propose that one should perform a randomized benchmarking procedure with a logical 2-design and measure the syndrome at each step, but do not necessarily apply decoding immediately. Combes et al. argue that applying the decoding operation may be delayed indefinitely, using a Gottesman-Knill algorithm [1] to commute the decoding to the end of the LRB procedure. If this is true, Combes et al. reasoned, we can decode in post-processing, allowing one to optimize the decoding using only classical computation.

One could indeed use an extended Gottesman-Knill algorithm to propagate the decoding through the perfect logical operators when $\mathbb{G} = \mathcal{C}_L$. However, it is unclear how

Combes et al. plan to commute the decodings through the various noise sources afflicting the quantum computer. In section 5.2 we show that this produces identical results for gate-independent noise without SPAM errors, but whether propagation works with more general noise is unclear. So, for now, we characterize Logical Randomized Benchmarking as two distinct procedures.

Combes et al. delay the decoding to justify that the recovery is perfect. As applying the correction in post-processing eliminates the possibility of error. However, as previously stated, it is unclear how the figure of merit measured when decoding is delayed compares to when R is applied after each gate. Thus, for the case of gate-dependent noise, we will spend very little energy considering the case where recovery is perfect. We instead focus on a weakened assumption: perfect measurement.

4.3.2 Fault Tolerance in the IBM Quantum Experience

Although the LRB procedure is a natural method of characterizing quantum gates, it is resource-intensive in ancilla qubits and syndrome measurements. To make a notion of RB on logical qubits physically realizable, Harper et al. in [14] only measured the syndrome at the end of the RB procedure. Also, because Harper et al. measured a physical implementation in the [[4,2,2]] parity check code, the authors were limited to post-select rather than decode. We excluded sequences in which an error was detected from the average of $p_{s_m}^{IBM}$. So this procedure gives a measure of the best-case performance of the code when we can only sense errors at the end of the sequence.

The authors of [14] discard sequences in which a non-trivial syndrome is measured, so the procedure can be described by including a $\bar{\Pi}_{\mathbb{S}}$ after the final gate is applied. Recall the definition from section 3.2 $\bar{\Pi}_{\mathbb{S}} = \alpha \Pi_{\mathbb{S}}$ where α depends on the state that $\Pi_{\mathbb{S}}$ acts on. Because α is correlated to the state, it may be possible that α depends on previously applied gates. If this occurs, then α may "bias" the twirl, meaning that certain terms in the sum have a higher weight than others. Biasing the twirl could cause upticks in extreme cases where only a small portion of the group contributes to survival probability.

The authors of [14] acknowledge this biasing issue in their supplementary material and resolve it by asserting that α is constant for their particular experiment. Our survey expands on this by considering noise where α is not constant. Our numerical evidence in fig. B.5 indicates that for gate-independent noise, the noise model can be significantly altered by the bias introduced by α. However, a full derivation of the survival probability when post-selected sequences are discarded remains elusive.

4.4 Summary

We gave a proof of the fit model for RB with gate-independent noise.

1. Algorithm 4.1 defines the character benchmarking procedure and it's sequence outcome $q_{\vec{G}}^{RB}$ is given in eq. (4.1).
2. We can enumerate some crucial steps of theorem 4.2 the decay model for gate-independent character benchmarking.
   (a) RB sequences with gate-independent noise can be written as a series of twirled noise models $\mathcal{T}_{\mathbb{G}}[\mathcal{E}] = \sum_{G\in\mathbb{G}} \phi(G)\mathcal{E}\phi(G)$.
   (b) Twirled noise can be projected down onto a matrix with rank equal to the multiplicity of ϕ given by $\mathcal{T}_{\mathbb{G}}[\mathcal{E}]_{\phi}$.
   (c) If the multiplicity of ϕ is 1 then $\mathcal{T}_{\mathbb{G}}[\mathcal{E}]_{\phi} = \lambda\Pi_{\phi}$ where $\lambda \in [0,1]$ so $p_{s_m}^{RB}$ is monotone decreasing.

Upticks are an increase in $p_{s_m}^{RB}$ and they indicate noise is non-Markovian.

1. As Markovianity is a crucial property for RB procedures; we will use upticks to prune inviable procedures.
2. Character Benchmarking with time-dependent noise $\mathcal{E}_i$ decreases monotonically.

We studied two previous implementations of RB in the context of stabilizer codes: logical randomized benchmarking in algorithm 4.2 and IBMQ Real RB in algorithm 4.3.

1. Logical randomized benchmarking by Combes et al. in [5] applies QEC after every logical gate.
   (a) Combes et al. give a method to commute decodings through perfect logical Clifford gates; however a method of commuting the decoding through the noisy circuit is unclear.
   (b) LRB commutes each decoding to the end of the circuit to justify that recovery is perfect. So in the absence of a proof one can commute through noisy gates, a perfect recovery should not be assumed.

2. Due to physical limitations, Harper et al. designed a procedure that only measures the syndromes and applies post-selection at the end.

   (a) Harper et al. discarded runs which had non-trivial syndromes, excluding the runs from the estimate of $p_{s_m}^{IBM}$.

   (b) Harper et al. demonstrated that post-selection could decrease noise rates in a physical setting.

Algorithm 4.2: Logical Randomized Benchmarking Procedure

Input: $\mathbb{G}$ - unitary 2-design on logical qubits, Q - $\{Q_s, Q_d\}$ projective measurement in span$\{\mathbb{L}\}$, M - list of sequence lengths, $\mathbb{S}^{gens}$ - Stabilizer Generators, D_s - Decoding

Output: $\mathcal{F} \in [0, 1]$ - Average Fidelity

for $m \in M$ **do**
- **repeat**
  - **repeat**
    - Prepare the encoded state $\rho = Q_s$.
    - **for** $1 \leq i \leq m$ **do**
      - Sample $G_i \in \mathbb{G}$ uniformly.
      - Classically compute $G_i G_{i-1}^\dagger$ where $G_0 = \mathcal{I}$.
      - Apply noisy implementation $\theta(G_i G_{i-1}^\dagger)$ to state.
      - Measure the syndrome using a Von-Neumann measurement of $\mathbb{S}^{gens}$.
      - Apply noisy decoding D_s to state. (or delay decoding)
    - Apply the noisy gate $\theta(G_m^\dagger)$ to the system.
    - Measure the syndrome using a Von-Neumann measurement of $\mathbb{S}^{gens}$.
    - Apply noisy decoding D_s to state. (or delay decoding)
    - Measure the projective measurement Q.
    - **if** *decoding was delayed* **then**
      - Use classical simulation [1] to commute each decoding past $\vec{G}$.
      - Apply the decoding perfectly to the outcome of Q in post.
    - Record outcome $q_{\vec{G}}^{LRB} \in \{Q_s, Q_d\}$.
  - **until** *desired number of* $q_{\vec{G}}^{LRB}$ *are obtained*;
  - Record survival probability $p_{s_m}^{LRB}$ as the average number of Q_s.
- **until** *desired number of* $p_{s_m}^{LRB}$ *are obtained*;

Fit the decay $A\lambda^m + B$ to the many estimates of $p_{s_m}^{LRB}$.

return $\mathcal{F} = \mathrm{Tr}[Q_d]\lambda + \mathrm{Tr}[Q_s]$.

Algorithm 4.3: IBM Quantum Experience "Real RB" Procedure

Input: $\mathbb{G}$ - orthogonal 2-design [15] on logical qubits,
Q - $\{Q_s, Q_d\}$ projective measurement in span$\{\mathbb{L}\}$,
M - list of sequence lengths, $\mathbb{S}^{gens}$ - Stabilizer Generators

Output: $\mathcal{F} \in [0, 1]$ - Average Fidelity

for $m \in M$ **do**
  repeat
    repeat
      Prepare the encoded state $\rho = Q_s$
      for $1 \leq i \leq m$ **do**
        Sample $G_i \in \mathbb{G}$ uniformly.
        Classically compute $G_i G_{i-1}^{\dagger}$ where $G_0 = \mathcal{I}$.
        Apply noisy implementation $\theta(G_i G_{i-1}^{\dagger})$ to state.
      Apply the noisy gate $\theta(G_m^{\dagger})$ to the system.
      Measure the projective measurement Q.
      Apply syndrome measurement and post-selection.
      Record outcome $q_{\vec{G}}^{IBM} \in \{Q_s, Q_d\}$.
    until *desired number of* $q_{\vec{G}}^{IBM}$ *are obtained*;
    Record survival probability $p_{s_m}^{IBM}$ as the average number of Q_s.
  until *desired number of* $p_{s_m}^{IBM}$ *are obtained*;
Fit the decay $A\lambda^m + B$ to the many estimates of $p_{s_m}^{IBM}$.
return $\mathcal{F} = \mathrm{Tr}[Q_d]\lambda + \mathrm{Tr}[Q_s]$.

Chapter 5

Logical Fidelity Estimation

We present a generalized class of procedures to estimate the performance of stabilizer codes, *Logical Fidelity Estimation* (LFE). Our flexible formalism will yield insight into LRB and IBMQ Real RB, identify non-Markovianity in various procedures, and propose new uses for post-selection which ensure Markovianity.

5.1 The LFE Sequence

To generalize the protocols suggested in [5, 14], we extend the character benchmarking sequence to include QEC operations $\mathcal{R}_i$ at each time step. We want the survival probability to be predominantly affected by the logical gates in $\mathbb{G}$, instead of those in $\mathbb{S}$ or $\mathbb{T}$ which are not computationally relevant.

Definition 5.1 (LFE Sequence)**.** A Logical Fidelity Estimation sequence $q^{\mathbb{L}}_{\vec{G}}$ is described by a character benchmarking sequence with an error reduction step $\mathcal{R}$ after each gate, see definition 3.4 for details on error reduction. We write $q^{\mathbb{L}}_{\vec{G}}$ for a gate sequence $\vec{G} = (G_1, \ldots, G_m, L)$ chosen from $\mathbb{G}$ by interleaving a *time-dependent error reduction scheme* $\mathcal{R}_i$ into the character benchmarking sequence

$$q^{\mathbb{L}}_{\vec{G}} = \theta(Q)\mathcal{R}_{m+1}\theta(G_\Pi G_m^\dagger)\prod_{i=2}^{m}\mathcal{R}_i\theta(G_i G_{i-1}^\dagger)\mathcal{R}_1\theta(G_1)\theta(\rho) \tag{5.1}$$

where $G_i \in \mathbb{G}$ are often logical operators and $\mathcal{R}_i = \sum_{s_i} R_{s_i,m}$. In this thesis $\mathcal{R}_i$ is a QEC map, a post-selection, or the identity and $\mathcal{R}_i$ implicitly depends only on sequence length m

and the ith syndrome measurement outcome s_i. However our formalism is general enough to allow LFE extensions, the most common is to delay decoding as in LRB.

We can calculate the LFE survival probability $p^{\mathbb{L}}_{s_m}$ of the LFE sequence in exact analogy to the character benchmarking sequence using algorithm 5.1

$$p^{\mathbb{L}}_{s_m} = \mathbb{E}_{\vec{G}} q^{\mathbb{L}}_{\vec{G}} \chi_{\phi_{\mathbb{L}}}(G_\Pi). \tag{5.2}$$

Algorithm 5.1: Logical Fidelity Estimation Procedure

Input: $\mathbb{G}$ - implemented group, Q - projective measurement $\{Q_s, Q_d\}$ in span$\{\mathbb{L}\}$, M - list of sequence lengths, $\mathcal{R}_i$ - time-dependent error reduction scheme $\phi_{\mathbb{L}}$ - irrep. with codomain in span$\{\mathbb{L}\}$ and multiplicity 1

Output: $\mathcal{F} \in [0, 1]$ - Average Fidelity

for $m \in M$ **do**
- **repeat**
  - **repeat**
    - Prepare the encoded state $\rho = Q_s$
    - **for** $1 \leq i \leq m$ **do**
      - Sample $G_i \in \mathbb{G}$ uniformly.
      - Classically compute $G_i G_{i-1}^\dagger$ where $G_0 = \mathcal{I}$.
      - Apply noisy implementation $\theta(G_i G_{i-1}^\dagger)$ to state.
      - Apply noisy QEC map $\mathcal{R}_i$ to state
    - Sample $G_\Pi \in \mathbb{G}$ uniformly.
    - Apply the noisy gate $\theta(G_\Pi G_m^\dagger)$ to the system.
    - Apply noisy QEC map $\mathcal{R}_{m+1}$ to state
    - Measure the binary PVM Q.
    - Record outcome $q^{\mathbb{L}}_{\vec{G}} \in \{Q_s, Q_d\}$.
  - **until** *desired number of* $q^{\mathbb{L}}_{\vec{G}}$ *are obtained*;
  - Record survival probability $p^{RB}_{s_m}$ as the average number of Q_s.
  - Weight $p^{RB}_{s_m}$ using the character $\chi_\phi(G_\Pi)$
- **until** *desired number of* $p^{\mathbb{L}}_{s_m}$ *are obtained*;

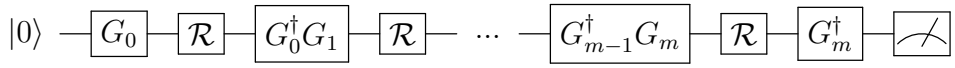
Fit the many estimates of $p^{\mathbb{L}}_{s_m}$ to an appropriate decay model, most likely $A\lambda^m + B$.

return $\mathcal{F} = \mathrm{Tr}[Q_d]\lambda + \mathrm{Tr}[Q_s]$.

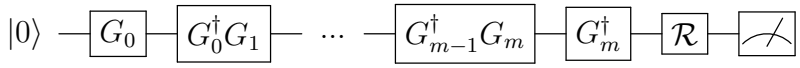
This thesis exclusively concerns LFE procedures that employ a fixed error reduction operator $\mathcal{R}$ and leave more exotic schemes to future work. We enumerate the error reduction schemes in fig. 5.1 based on where they interleave $\mathcal{R}$.

Time-Dependent Error Reduction Schemes

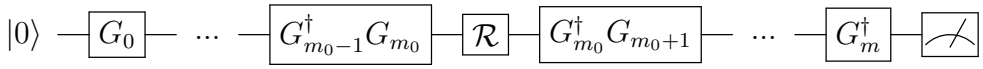
Always - Apply $R_{s_i,m} = R_{s_i} \ \ \forall i \in \mathbb{Z}^m$ for all gates except the final one.



At the End - Apply R with final gate $R_{s_i,m} = R_{s_i}^{\delta_{i,m+1}}$.



Fixed Time - Apply R at a fixed sequence length m_0, $R_{s_i,m} = R_{s_i}^{\delta_{i,m_0}}$ independent from m.



Measure Always, Recovery at the End - Measure the syndrome at each R_i and apply $R_i = R_{s_1 \ldots, s_{m+1}}^{\delta_{i,m+1}}$ which depends on every previous syndrome. (LFE Extension)

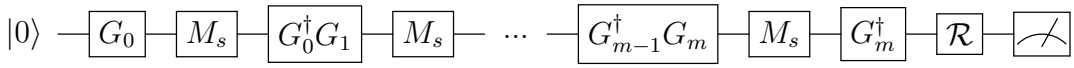


Figure 5.1: An enumeration of all the time-dependent error reduction schemes used in this thesis to perform Logical Fidelity Estimation. One uses these schemes to specify how $\mathcal{R} = \sum_s R_{s,m}$ will be interleaved in the LFE protocol. We include with each scheme a circuit diagram depicting the corresponding LFE procedure. Picking the form of $\mathcal{R}$ specifies the Logical Fidelity Estimation sequence. The gates labeled with M_s are syndrome measurements.

We may choose to specify the implemented form of error reduction in the name of a procedure. For example, QEC always LFE defines all the $\mathcal{R}_i$ for $i \leq m$ to be the same QEC map. Another example is rejected post-selection at the end LFE. This protocol will set the outcome of $q_{\vec{G}}^{\mathbb{L}}$ to Q_d if the final syndrome measured at $\mathcal{R}_{m+1}$ is non-trivial.

Now that we have defined the data collection method, it is natural to talk about manipulating the data. Such is the main topic of our next section.

5.2 Post-processing LFE Data

We distinguish LFE from previous procedures not only by the implemented procedure, but also by how one is allowed to post-process data. In LFE, we allow two methods of post-processing post-selected sequences which contribute to the survival probability $p^{\mathbb{L}}_{s_m}$ in distinct ways. Discarded post-selection attempts to replicate the performance of the code when post-selection is applied. In comparison, rejected post-selection has more easily derivable robustness properties and measures the probability of post-selection indirectly. A full description of both methods of post-selection can be found below.

Definition 5.2 (Discarded Post-Selection)**.** Post-selected sequences are not used to estimate $p^{\mathbb{L}}_{s_m}$. Write $\mathcal{R}_i = \bar{\Pi}_{\mathbb{S}}$ as in eq. (3.4).

Definition 5.3 (Rejected Post-Selection)**.** Post-selected sequences are used to estimate $p^{\mathbb{L}}_{s_m}$, but the outcome is treated as if Q_d was measured. So we write $\mathcal{R}_i = \Pi_{\mathbb{S}}$ from eq. (3.1).

Note that performing discarded post-selection may require the experimentalist to adjust how much precision is needed. This adjustment is required because observing a high number of rejections could make obtaining many estimates of $q^{\mathbb{L}}_{\vec{G}}$ difficult. So in algorithm 5.1 we have purposely kept the number of sequences ambiguous so that the experimentalist can adjust precision as required. For this thesis, we assume that enough points can always be obtained and leave arguments about precision to future work.

The other primary post-processing technique we cover is delaying the decoding, where one can adjust the Pauli frame in post-processing to optimize the fidelity. In [5] the authors assert that one can use classical computation to commute decodings to the end of the gate sequence using classical processing.

Delaying the decoding seemed feasible at first because $\mathbb{G}$ can be a subset of the Cliffords. The decodings are often Paulis, both of which are efficiently simulatable using a Gottesman-Knill algorithm [1]. However, the authors of [5] never mention how one would commute decodings through the error models. Without full noise tomography, it is unclear how to commute the decoding through the noise model to the end of the LFE sequence. In theorem 5.4, we use the fact that one can commute the recoveries through the twirled noise to show a particular case without measurement errors. Theorem 5.4 will require writing the LFE sequence as if we can gather all logical parts of the decoding on the RHS which we make explict in lemma 5.1.

Lemma 5.1. *Any QEC always LFE sequence $q^{\mathbb{L}A}_{\vec{G}}$ with perfect measurement $\mathcal{R} = \mathcal{R}_{PM} = \sum_s \mathcal{E}_s \phi(D_s) \Pi_{\mathbb{S}} \phi(T_s)$ can be written*

$$p^{\mathbb{L}}_{s_m} = \mathbb{E}_{\vec{s}} \mathbb{E}_{\vec{G}} \;\; \theta(Q)\theta(G_\Pi G_m^\dagger) \prod_{i=2}^{m} \mathcal{R}^0_{s_1} \theta(G_i G_{i-1}^\dagger) \mathcal{R}^0_{s_1} \phi(G_1) \phi(L_{\vec{G}}) \theta(\rho) \tag{5.3}$$

for $\theta(G) = \phi(G)\mathcal{E}$ gate-independent and trivial $\mathcal{R}^0_{s_i}$ so that D_{s_i} commutes with all $L \in \phi(\mathbb{L})$.

Theorem 5.4. *If a QEC always LFE sequence has $\theta(Q) = Q$ and $L_s \in \mathbb{G}\ \forall s$ then we can use lemma 5.1 and simply re-label the G's (rotate the twirl as in lemma 4.1) to commute $L_{\vec{G}}$ next to Q. Allowing one to delay application of $L_{\vec{G}}$ to post-processing.*

Proof of lemma 5.1. Begin by repeating the proof for character benchmarking's fit model in section 4.1 up to eq. (4.9)

$$p^{\mathbb{L}}_{s_m} = \theta(Q) \mathcal{T}_{\mathbb{G}} [\mathcal{E}\mathcal{R}]^m_{\mathbb{G}^{\mathbb{L}}} \mathcal{E} \theta(\rho) \tag{5.4}$$

Let's write out the explicitly and using eq. (3.3) to decompose $D_s = L_s S_s T_s$ yields

$$p^{\mathbb{L}}_{s_m} = \sum_{s=0}^{2^{n-k}-1} \theta(Q) \mathcal{T}_{\mathbb{G}} [\mathcal{E}\mathcal{E}_s \phi(L_s) \phi(T_s S_s) \Pi_{\mathbb{S}} \phi(T_s)]_{\mathbb{G}^{\mathbb{L}}} \mathcal{T}_{\mathbb{G}} [\mathcal{E}\mathcal{R}]^{m-1}_{\mathbb{G}^{\mathbb{L}}} \mathcal{E} \theta(\rho). \tag{5.5}$$

Next we must note the $\phi(L_s)$ can be commuted past $\phi(T_s S_s) \Pi_{\mathbb{S}} \phi(T_s)$ as the former is in $\text{span}\{\mathbb{L}\}$ and the latter is in $\text{span}\{\mathbb{S}\mathbb{T}\}$ to get

$$p^{\mathbb{L}}_{s_m} = \sum_{s=0}^{2^{n-k}-1} \theta(Q) \mathcal{T}_{\mathbb{G}} [\mathcal{E}\mathcal{E}_s \phi(T_s S_s) \Pi_{\mathbb{S}} \phi(T_s) \phi(L_s)]_{\mathbb{G}^{\mathbb{L}}} \mathcal{T}_{\mathbb{G}} [\mathcal{E}\mathcal{R}]^{m-1}_{\mathbb{G}^{\mathbb{L}}} \mathcal{E} \theta(\rho). \tag{5.6}$$

We define $L_s^{G_i} = G_i^\dagger L_s G_i$ and use the fact that $m_{\mathbb{G}^{\mathbb{L}}} = 1$ to commute the ϕ through

$$p^{\mathbb{L}}_{s_m} = \sum_{s=0}^{2^{n-k}-1} \sum_{G_m \in \mathbb{G}} \theta(Q) \phi(G_m^\dagger) \mathcal{E}\mathcal{E}_s \phi(T_s S_s) \Pi_{\mathbb{S}} \phi(T_s) \phi(G_m) \mathcal{T}_{\mathbb{G}} [\mathcal{E}\mathcal{R}]^{m-1}_{\mathbb{G}^{\mathbb{L}}} \phi(L_s^{G_m}) \mathcal{E} \theta(\rho). \tag{5.7}$$

Because the multiplicity of $\mathbb{G}^{\mathbb{L}}$ is 1 we can commute $\phi(L_s^{G_i})$ through the noise to one end of the circuit. We can repeat this process for all the twirls to obtain

$$p^{\mathbb{L}}_{s_m} = \mathbb{E}_{\vec{s}} \mathbb{E}_{\vec{G}} \;\; \theta(Q) \prod_{i=1}^{m} \phi(G_i^\dagger) \mathcal{E} R^0_{s_i} \phi(G_i) \left(\prod_{i=1}^{m-1} \phi(L_{s_i}^{G_i}) \right) \mathcal{E} \theta(\rho) \tag{5.8}$$

The quantity in parentheses is a character benchmarking sequence with unitary noise L_s, so it can be simulated, inefficiently. Right product is the LFE sequence if $L_s = \mathcal{I}$ to R. $\square$

To test theorem 5.4 we implemented measure always recovery at the end LFE and QEC always LFE. We plotted the resulting survival probabilities in fig. 7.1. We found that the two estimations of fidelity agreed. So delaying the decoding to the end may be viable when noise is gate-independent. However, the need to commute the L_s forward and backward through the circuit makes this proof much harder to replicate with gate-dependent noise. It is possible to delay logical Clifford decodings to the end, but not to post-processing, by satisfying the following caveat. One may have to adjust the frame of Q to ensure the right information is measured.

5.3 Summary

The generalized family of logical fidelity estimation procedures we have defined above will serve as a starting point to search for procedures that characterize $\mathcal{R}$, a syndrome measurement with decoding or a post-selection.

1. LFE schemes can be broadly characterized into four types defined in fig. 5.1:
   (a) Always LFE: inserts post-selection or decoding after each independent logical operator.
   (b) At the end LFE: minimize the number of syndrome measurements by only applying $\mathcal{R}$ at the end of the LFE sequence.
   (c) At Fixed Time LFE: Minimizes the number of syndrome measurements by only applying $\mathcal{R}$ at a fixed time.
   (d) measure always recovery at the end LFE: may simulate a perfect recovery.

We found that one has several options for post-processing LFE data depending on the error reduction scheme $\mathcal{R}_i$ used.

1. When $\mathcal{R}$ is post-selection we have 2 options for post-processing data:
   (a) Post-selection discarded simply excludes post-selected sequences from our estimate of $p^{\mathbb{L}}_{s_m}$.
   (b) Post-selection rejected treats post-selected sequences as if they are failures of the logical circuit.
2. If $\mathcal{R}$ is QEC, $\mathcal{E}$ is gate-independent and there is no measurement error on Q, we can apply decoding in post-processing.

In table 5.1 we summarize why one might study each LFE procedure proposed.

Motivating LFE Procedures

	Always	At the End	Fixed Time
QEC	robust to gate indep. suggested in [5]	near-term tests QEC	near-term tests QEC
Discarded PS	robust to gate indep. maximal performance	near-term implemented in [14]	near-term
Rejected PS	robust to gate dep.	near-term	near-term
	Quantifies error detection rate		

Table 5.1: A table of LFE procedures in this thesis with each box explaining why it might be interesting. Each box in the table defines a time-dependent error reduction scheme, which specifies an LFE procedure as in definition 5.1. Near-term indicates that the procedure minimizes the number of syndrome measurements, which will be crucial as errors from one's measurement can be an order of magnitude larger than errors on gates [21, 32]. Procedures that test QEC are more valuable than post-selection as QEC does not require a loss of trace to eliminate errors. So we have labeled procedures that test QEC as such. Maximal performance indicates that discarded post-selection always LFE only quantifies undetectable errors, so the code's performance is maximized.

Chapter 6

Non-Markovianity in Logical Fidelity Estimation

This chapter contains the technical contributions of this thesis. So require that the reader be familiar with the ideas of chapter 5.

We can use theorem 4.2 to make a general statement about LFE so long as $\mathcal{R}$ is omitted from the last step. However, we only make use of special cases of this theorem in this thesis.

Theorem 6.1. *If $\mathcal{R}_{m+1} = \mathcal{I}$ and each $\mathcal{R}_i$ may be a distinct QEC map or rejected post-selection the survival probability of an LFE sequence $p^{\mathbb{L}}_{s_m}$ is monotone decreasing when $\theta(G) = \phi(G)\mathcal{E}$ is gate-independent and $\mathcal{R}_i$ has perfect measurement.*

Proof. We can redefine $\mathcal{E}_i\mathcal{R}_i \to \mathcal{E}_i$ because $\mathcal{R}_i$ has a matrix representation. Concretely, we found a survival probability for RB $p^{RB}_{s_m}$ which is equivalent to $p^{\mathbb{L}}_{s_m}$. Since noise is independent of the gates we can apply theorem 4.2 to show $p^{\mathbb{L}\mathrm{A}}_{s_m}$ is monotone decreasing. □

6.1 Markovianity in Always LFE

We investigate the presence of non-Markovianity in Always LFE under variations in the noise model and $\mathcal{R}$. We begin with the simple case of always LFE with gate-independent noise. We then consider gate-dependent noise in the case of post-selection before moving on to considering forms of QEC. Two forms of QEC will be considered. Perfect recovery and perfect measurement. We will find that the former is robust to gate-dependent noise while the latter may be vulnerable. We conclude this section by summarizing our results.

6.1.1 QEC Always LFE with Gate-Independent Noise

Several useful corollaries follow directly from theorem 6.1, the first of which is listed below.

Corrolary 6.1. *The survival probability of a QEC always or rejected post-selection always LFE sequence $p_{s_m}^{\mathbb{LA}}$ is monotone decreasing when $\theta(G) = \phi(G)\mathcal{E}$ is gate-independent and $\mathcal{R}$ has perfect measurement.*

Theorem 6.1 is ineffective in the case of discarded post-selection as the normalization in $\bar{\Pi}_{\mathbb{S}}$ cannot be included in $\mathcal{E}$ without making $\mathcal{E}$ dependent on the current state. However, the statement does indeed hold.

Theorem 6.2. *The survival probability of a discarded post-selection always LFE sequence $p_{s_m}^{\mathbb{LA}}$ is monotone decreasing when $\theta(G) = \phi(G)\mathcal{E}$ is gate-independent, $\phi(G_i)$ preserves the code space, and $\mathcal{R}_i$ has perfect measurement.*

Proof. Consider a unitary error $\theta(G) = \phi(G)U$. We will show a unitary noise U cannot induce upticks in discarded post-selection always LFE. The proof for a general quantum process can be inferred from linearity.

We recall the fact from section 2.2 that we can write any unitary as $\text{span}(\mathcal{P}_n)$. Then if $U \in \text{span}(\mathbb{LS})$ no post-selection occurs so the sequence is a character benchmarking sequence. If $U \in \text{span}(\mathbb{TS})$ then post-selection does occur, but the post-selected sequences will have the same logical state as if they had not been post-selected. So the logical state and thus the output is unaffected by U.

We can see this by following the character benchmarking proof up to eq. (4.10) to see

$$q_{\vec{G}}^{\mathbb{LA}} = \theta(Q)\phi(G_\Pi)\prod_{i=1}^{m}\phi(G_i^\dagger)U\bar{\Pi}_{\mathbb{S}}\phi(G_i)\theta(\rho) \tag{6.1}$$

$$= \theta(Q)\phi(G_\Pi)\phi(G_m^\dagger)U\bar{\Pi}_{\mathbb{S}}\phi(G_m)\prod_{i=1}^{m}\phi(G_i^\dagger)\Pi_{\mathbb{S}}U\bar{\Pi}_{\mathbb{S}}\phi(G_i)\theta(\rho) \tag{6.2}$$

where the second step can be seen by noting that $\mathbb{G}$ preserves the code space. From this, one can see that if U has any part in $\text{span}(\mathbb{T})$ then the sequence gets post-selected. Thus, the only remaining part of the unitary are the logical parts in $\text{span}(\mathbb{LS})$ which are twirled in the usual manner. □

6.1.2 Post-Selection Always LFE with Gate-Dependent Noise

We give a temporary solution to avoid upticks in always LFE by suggesting that one should replace the decoding with a rejected post-selection. In this case, the noise model cannot be correlated to past gates since any run with non-trivial information in the syndrome is excluded from the survival probability.

Theorem 6.3. *The survival probability of a rejected post-selection always LFE sequence* $p_{s_m}^{\mathbb{LA}}$ *is monotone decreasing when* $\theta(G) = \mathcal{E}_G\phi(G)$ *is gate-dependent and* $\mathcal{R}$ *has perfect measurement.*

Proof. The above can be seen as a subcase of QEC always LFE with perfect recovery in theorem 6.4. Simply pick the decoding $D_s = \delta_{s,\vec{0}}$ and the proof remains unchanged. □

6.1.3 Perfect Recovery with Gate-Dependent Noise

We will explore essentially the case presented in [5] in our framework. *Perfect recovery* (PR) is a strong assumption on $\mathcal{R}$, which assumes a perfect projection back onto the code space with no loss of trace, $\bar{\Pi}_{\mathbb{S}}$. In [5] Combes et al. claimed the logical randomized benchmarking procedure could achieve perfect recovery by continuously updating the Pauli frame using the syndrome measurement outcomes to delay the decoding to the end of the LRB sequence. It is unclear how one could achieve this scheme with gate-dependent noise. So we substitute the desired outcome of the delayed decoding with a simplified perfect recovery model $\mathcal{R}_{PR} = \sum_s \mathcal{E}_s\phi(D_s)\bar{\Pi}_{\mathbb{S}}$ which replicates the ultimate results of [5].

Theorem 6.4. *The survival probability of an QEC always LFE sequence* $p_{s_m}^{\mathbb{LA}}$ *is monotone decreasing when* $\theta(G) = \phi(G)\mathcal{E}_G$ *is gate-dependent and* $\mathcal{R}$ *is perfect recovery.*

Proof. Recall the definition of perfect recovery as $\mathcal{R}_{PR} = \mathcal{R} = \sum_s \mathcal{R}_s = \sum_s \mathcal{E}_s\phi(D_s)\bar{\Pi}_{\mathbb{S}}$ where s sums over the syndrome outcomes. Then an implemented sequence is

$$q_{\vec{G}}^{\mathbb{LA}} = \theta(Q)\mathcal{R}\theta(G_\Pi G_m^\dagger)\prod_{i=2}^{m}\mathcal{R}\theta(G_iG_{i-1}^\dagger)\mathcal{R}\theta(G_1)\theta(\rho) \tag{6.3}$$

$$= \theta(Q)\mathcal{R}\Pi_{\mathbb{S}}\theta(G_\Pi G_m^\dagger)\mathcal{R}\prod_{i=2}^{m}\Pi_{\mathbb{S}}\theta(G_iG_{i-1}^\dagger)\mathcal{R}\Pi_{\mathbb{S}}\theta(G_1)\theta(\rho). \tag{6.4}$$

Next we can use the above form to define an effective logical representation of the sequence $\theta_{\mathbb{S}}(G_iG_{i-1}^\dagger) = \Pi_{\mathbb{S}}\theta(G_iG_{i-1}^\dagger)\mathcal{R}\Pi_{\mathbb{S}}$. By viewing $\theta_{\mathbb{S}}$ in the unencoded frame of fig. 3.1 we can

see $\theta_{\mathbb{S}}$ effectively limits θ to the $2^k \times 2^k$ matrix block corresponding to when the syndrome is zero, i.e. the code space block. We can re-define $\theta(Q)\mathcal{R} \to \theta(Q)$ as we are guaranteed to be in the code space

$$q_{\vec{G}}^{\mathbb{LA}} = \theta(Q)\theta_{\mathbb{S}}(G_\Pi G_m^\dagger)\prod_{i=2}^{m}\theta_{\mathbb{S}}(G_i G_{i-1}^\dagger)\theta(G_1)\theta(\rho). \tag{6.5}$$

Thus we have reduced the sequence to a character benchmarking sequence on the code space block. Which by theorem two of [17], can be fitted to an exponential decay. □

As its name suggests, PR is an unrealistic assumption on the recovery. Moreover, PR eliminates all parameters which describe how R_i acts on $\mathbb{ST}$. Thus, PR should be strictly "easier" to characterize than a recovery which interacts with $\mathbb{ST}$ in some currently undefined sense of "hardness" of Logical Fidelity Estimation. Considering an error model assumption that is weaker than perfect recovery may have complex implications for robustness.

6.1.4 Perfect Measurement with Gate-Dependent Noise

We explore the weaker assumption of perfect measurement $\mathcal{R}_{PM} = \sum_s \mathcal{E}_s\phi(D_s)\Pi_{\mathbb{S}}\phi(T_s)$. Here, upticks appear under gate and syndrome-dependent noise.

We could not expand the above proofs to gate-dependent noise with perfect measurement. This is because syndrome qubits may store information such that D_s and $\mathcal{E}_s$ can depend on previous gates as presented in example 6.2. In other words, the redefinition $R_{s_i}\mathcal{E}_{G_i} \to \mathcal{E}_{G_i}$ may not work because we found in example 6.2 below a case where $R_{s_i}\mathcal{E}_{G_i}$ can be compiled as $\mathcal{E}_{G_i,G_{i-1}}$, a noise model which depends on the previous 2 gates.

In this subsection we use the logical Clifford group $\mathcal{C}_L = \mathbb{G}$ and notate $\phi(C_L)$ as C_L for clarity and to save space. The distinction between $\phi(C_L)$ and C_L will be clear from the context.

First, present an adversarial gate-dependent noise model that produces upticks by storing the entire gate history in the syndrome bits. It will also serve to make the connection to the syndrome's dependence on previous gates obvious. Recall that the single-qubit Clifford group has 24 elements, implying we may encode a Clifford C with as little as 5 bits ($24 < 2^5 = 32$). Call an example of this encoding $s(C)$.

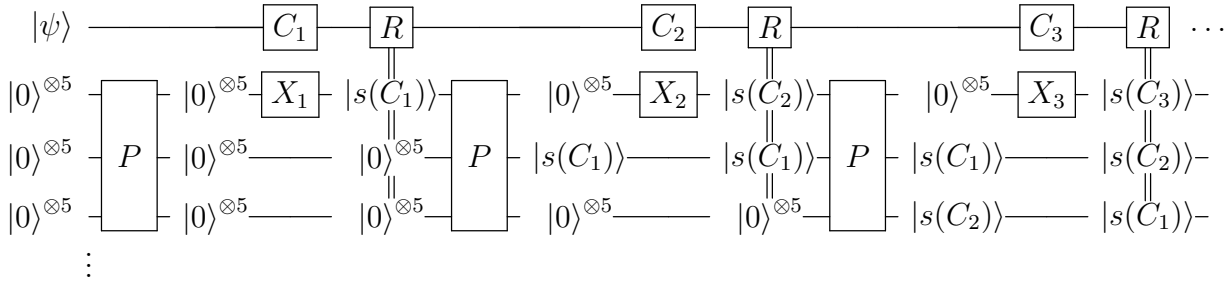


Figure 6.1: A circuit depicting the error model discussed in example 6.1 in an unencoded frame, we have omitted the E and $E^\dagger$ to save space. The double lines denote measurement and classical control. When the lines are absent, the state is written only to show the reader the current syndrome. Note that each time $\mathcal{R}_i$ is applied, it obtains a unique syndrome for each gate history. We can use this information to design a recovery that arbitrarily rejects or corrects the logical state, leading to upticks.

Example 6.1. Consider a $[[n,\, 1]]$ error correcting code where $n > 5m + 1$ and m is the maximum length of the LFE Sequence. An important tool to achieve this is the map

$$P = \prod_{i=n-5}^{2} \text{SWAP}_{i,i+5} \tag{6.6}$$

which swaps qubits down a long line of bits in blocks of 5. The noisy gate $\theta(C_L)$ is then

$$\theta(C_{L_i}) = T_{s(C_i)} C_{Li} \mathcal{E}_P \tag{6.7}$$

so that the gate-dependent destabilizer $T_{s(C_i)}$ encode the gate C_L in the first five syndrome bits of $s(C)$ while the rest are set to 0. The noise $\mathcal{E}_P = EPE^\dagger$ uses P to "pass" information correlated to all the C_{L_i} down the line of syndrome bits. We can view this noise model in the unencoded frame defined in fig. 3.1 to give a concrete definition

$$\theta(C_{L_i}) = EX_i(C_i \otimes \mathcal{I}_{n-k})PE^\dagger \tag{6.8}$$

where $X_i = ET_{s(G_i)}E^\dagger$ as depicted in fig. 6.1.

After the gate comes the recovery with noise model $\mathcal{E}_s = T_s$ so that the measured syndrome is preserved. Thus, the first gate is encoded in the first five syndromes, while the second step transfers the information about the first gate to syndrome bits 6 through 10. The second gate is then encoded into syndrome bits 1 through 5, and so on. As we have assumed $n > 5m + 1$, there will always be enough syndrome bits to encode the gate history at every time-step uniquely.

We have defined a system where the syndrome uniquely encodes the history of gates applied until any time step. Thus, we can a priori design a decoder D_s, which introduces Pauli errors and correct them. For example, identify the syndromes corresponding to when the input state $|\psi\rangle$ is in an eigenstate of Z_L and apply a Pauli X_L error with D_s. By the structure of the Clifford group, we can be sure that our recovery scheme will reject this state 1/3 of the time. One can keep track of the commutation relations of X_L and the applied C_L to arbitrarily control when the state is rejected and accepted.

In the above example, $\mathbb{ST}$ acts as a memory for the full history of the applied gates. Our noise model required a significant lack of syndrome reset and highly gate-dependent noise to create an imprint of the gate history.

Next, we present a less adversarial gate-dependent noise model where a QEC always LFE sequence exhibits oscillations in the sequence length. This example attempts to minimize the amount of information about previous gates that the syndrome needs to hold in order to produce upticks. We use only the first three syndrome qubits to store information about the previous logical operator, then adjust the recovery to correct the error. This example shows that although heavy gate-dependence is still required, we can relax the unrealistic assumption that many syndromes do not reset in exchange for a small lack of syndrome reset, namely, on syndromes 100 and 001 as described below.

Example 6.2. Consider a QEC always LFE procedure benchmarking the [[7,1]] stabilizer code with $\mathcal{C}_L$, the logical Cliffords and $\mathcal{R}_{PM} = \sum_s \mathcal{E}_s \phi(D_s) \Pi_{\mathbb{S}} \phi(T_s)$. Choose $\mathcal{E}_{C_L}$ on the last 3 syndromes to encode the action of C_L on X_L. For example: 001 in 001 if C_L commutes with X_L, 010 if C_L maps X_L to Y_L, and and 100 if C_L maps X_L to Z_L. Assume $\mathcal{E}_i = \mathcal{I}$ so $\mathcal{R}$ applies a perfect reset on these qubits. We now ignore the last 3 syndromes until eq. (6.12).

Define the recovery's error model on the first 3 syndromes as the maps $\mathcal{E}_{100} = \phi(T_2)$ and $\mathcal{E}_{001} = \phi(T_3)$, both are meant to store information about previous gates in the syndromes by preventing the syndrome from resetting. Define part of the recovery: $D_{100} = D_{110} = X_L$ and keep D_{011} undefined for now. Otherwise assume $\mathcal{E}_i = \phi(D_i) = \mathcal{I}$ for all i for simplicity. Finally, we define $\mathcal{E}_{C_L}$ on the first 3 syndromes, let $\mathcal{E}_I \to \mathcal{E}_I \phi(T_1)$ and $\mathcal{E}_{C_L} \to \mathcal{E}_{C_L} \phi(T_3)$ otherwise.

To simplify, we only explicitly show the first three syndromes. Perfectly prepare the initial state $|\psi\rangle = |0_L \Pi_{\mathbb{S}}\rangle$. After the first gate we obtain

$$\theta(C_{L_1}) |\psi\rangle = |0_L \Pi_{100}\rangle \text{ or } C_{L_1}^{\neq I} |0_L \Pi_{001}\rangle \tag{6.9}$$

where each term on the RHS corresponds to the different gate-dependent errors on C_L. After the first recovery we obtain

$$\mathcal{R}\theta(C_{L_1}) |\psi\rangle = X_L |0_L \Pi_{010}\rangle \text{ or } C_{L_1}^{\neq I} |0_L \Pi_{001}\rangle . \tag{6.10}$$

After the second gate

$$\theta(C_{L_2})\mathcal{R}\theta(C_{L_1})\left|\psi\right\rangle = X_L\left|0_L\Pi_{110}\right\rangle \text{ or } C_{L_2}^{\neq I}X_L\left|0_L\Pi_{011}\right\rangle \text{ or } C_{L_1}^{\neq I}\left|0_L\Pi_{101}\right\rangle \text{ or } C_{L_2}^{\neq I}C_{L_1}^{\neq I}\left|0_L\Pi_{000}\right\rangle \tag{6.11}$$

and after the second recovery

$$(\mathcal{R}\theta(C_L))^2\left|\psi\right\rangle = \left|0_L\Pi_{000}\right\rangle \text{ or } D_{011}C_{L_2}^{\neq I}X_L\left|0_L\Pi_{000}\right\rangle \text{ or } C_{L_1}^{\neq I}\left|0_L\Pi_{000}\right\rangle \text{ or } C_{L_2}^{\neq I}C_{L_1}^{\neq I}\left|0_L\Pi_{000}\right\rangle. \tag{6.12}$$

Because the last three syndromes give the commutation relations of C_{L_2} and X_L, we can choose D_{011} to correct the X_L error. As there is no R_{m+1}, the gate inversion can be treated as if it were perfect.

We calculate the survival probability after the first $\mathcal{R}\theta(C_{L_1})$

$$q_{G_1}^{\mathbb{L}} = \mathbb{E}_{C_{L_1}}\left\langle\psi_L\right| C_L^\dagger\mathcal{R}\theta(C_L)\left|\psi_L\right\rangle = \frac{|\mathcal{C}_L|-1}{|\mathcal{C}_L|} \tag{6.13}$$

the probability X_L was not applied by the recovery. And after the second round the error is corrected regardless of what happened in the first round

$$q_{G_2,G_1}^{\mathbb{L}} = \mathbb{E}_{C_{L_1},C_{L_2}}\left\langle\psi_L\right| C_{L_{1:2}}^\dagger(\mathcal{R}\theta(C_L))^2\left|\psi_L\right\rangle = 1. \tag{6.14}$$

An uptick preceding oscillations with amplitude $\approx 2\%$ in the survival probability.

Next, we note always LFE with perfect measurement can uptick if the error reduction scheme is a trivial decoding. This example shows that we need not adversarially choose the decoding to induce an uptick, implying that upticks can occur in any QEC always LFE procedure with gate-dependence.

Example 6.3. The above holds even for trivial decodings that do not affect the logical space. Include the logical part of the decoding in $\mathcal{E}_s$ and carry through the above calculation with $D_i = I$ to give a procedure and noise model which upticks.

6.1.5 Summary

Always LFE seems to be generally robust to gate-independent noise. However, extending the results to gate-dependent noise proved to be more difficult. We have shown that

relaxing the assumptions on $\mathcal{R}$ beyond the simplistic setting of perfect recovery allows for unfamiliar upticks if the noise model is heavily gate-dependent. These upticks can be attributed to dependence on previously applied gates and can be guaranteed to disappear when $\mathcal{R}$ is a post-selection. We found a workaround to avoid upticks by showing that rejected post-selection always is robust to gate-dependent noise.

All of the procedures considered so far are resource-intensive, as they require many syndrome measurements. The errors on measurements are likely to be almost an order of magnitude higher than the error on the gates [21, 32], so we will spend the next two sections attempting to design robust procedures that minimize these syndrome measurements. We hope that reducing the number of syndrome measurements will make these procedures feasible in the near-term.

6.2 Markovianity in At the End LFE

One can see a key feature of non-Markovianity in at the end LFE by taking the special case where $\mathcal{R}$ is rejected post-selection. One can imagine this procedure as instead of measuring Q, the experimentalist measures $Q_{Ext} = \{Q_s\Pi_{\mathbb{S}}, I - Q_s\Pi_{\mathbb{S}}\}$. The representation $\phi(G)$ will often have multiple copies in this expanded operator space. For instance, if the $\mathbb{G} = \mathcal{C}_L$ one could swap the logical qubit with any syndrome qubit to obtain another representation.

We present 2 example LFE sequences, which demonstrate that at the end LFE can produce upticks under gate-independent noise. Example 6.4 shows that upticks can occur when the $\mathcal{R}$ is a post-selection, while example 6.5 shows that upticks occur when $\mathcal{R}$ is QEC.

Example 6.4. Consider an at the end LFE procedure with $\theta(\rho) = \Pi_L\Pi_{\mathbb{S}}$, $\theta(Q) = \{\Pi_L, \mathcal{I} - \Pi_L\}$ and $R_{PS} = \Pi_{\mathbb{S}}$ is a rejected post-selection benchmarking the Logical Cliffords $\mathcal{C}_L$. This procedure will benchmark an error model given by the following small unitary overrotation on $\mathbb{ST}$

$$\theta(C_L) = \mathcal{E}_{DL}e^{i\delta T}\phi(C_L) \tag{6.15}$$

where $\mathcal{E}_{DL} = (1-p)\rho + \frac{p}{4}\sum_{L\in\mathbb{L}} L\rho L$ is depolarizing noise on the logical space.

If we repeat this gadget many times one can see that correlations can build up on $\mathbb{ST}$

$$\mathcal{T}[\mathcal{E}]^m = \mathcal{E}_{DL}^m e^{im\delta T}. \tag{6.16}$$

Subsequently if one post-selects based on the syndrome corresponding to T we can write the survival probability $p_{s_m}^{\mathbb{L}\mathrm{E}}$ as.

$$p_{s_m}^{\mathbb{L}\mathrm{E}} = \left\langle \Pi_L^+ \Pi_{\mathbb{S}} \right| \mathcal{T}^m \left| \Pi_L^+ \Pi_{\mathbb{S}} \right\rangle \tag{6.17}$$

$$= \left\langle \Pi_L^+ \right| \mathcal{E}_{DL}^m \left| \Pi_L^+ \right\rangle \left\langle \Pi_{\mathbb{S}} \right| e^{im\delta T} \left| \Pi_{\mathbb{S}} \right\rangle$$

$$= p_{DL}^m \ \cos(\delta m) \tag{6.18}$$

We recognize this damped oscillator behaviour in the simulated data of fig. B.8.

If the error reduction scheme is not a post-selection, but instead uses a QEC map to affect Q, upticks on $Q^{synd} = \{I - \Pi_{\mathbb{S}}, \Pi_{\mathbb{S}}\}$ induce upticks in Q. Explicit examples of this can be found below.

Example 6.5. Repeat the above example, except having $\mathcal{R}$ perfectly measure and apply a logical Pauli, say X_L, in response to the syndrome T produces. Then as oscillations occur in $\mathbb{ST}$, the probability X_L is produced by $\mathcal{R}$ oscillates as well. X_L will register as an error in the logical space, inducing oscillations in the sequence length identical to the above.

Because at the end LFE does not follow standard RB decays, we expand our survey to include another class of near-term procedures. Although these new procedures also produce non-standard decays, more of them are guaranteed to be Markovian while again using few syndrome measurements.

6.3 Markovianity in Fixed Time LFE

As we saw in the previous section, when we put a QEC map at the end of the LFE procedure, upticks are produced. A simple explanation is that $\mathcal{R}$ acts differently for different sequence lengths. However, if $\mathcal{R}$ performed at a fixed sequence length, then the upticks presented in section 6.2 cannot occur as $\mathcal{R}$ applied is independent of the sequence length. The corollary below an immediate consequence of theorem 6.1.

Corrolary 6.2. *The survival probability of a Fixed Time LFE sequence $p_{s_m}^{\mathbb{L}F}$ is monotone decreasing when $\theta(G) = \phi(G)\mathcal{E}$ is gate-independent and $\mathcal{R}$ is a rejected post-selection or QEC.*

In near-term devices, the noise from syndrome measurement is likely to be large. So to keep measurement noise from dominating $p_{s_m}^{\mathbb{L}}$ it is expected that LFE procedures that

minimize the number of syndrome measurements will be useful in near-term applications. Because at the end LFE will certainly not be robust to gate-dependent errors, fixed time LFE is the only known procedure that minimizes syndrome measurements and may be robust to gate-dependence.

Lemma 6.3. *In rejected post-selection at fixed time LFE, $p_{s_m}^{\mathbb{L}A}$ is monotone decreasing when $\theta(G) = \mathcal{E}_G\phi(G)$ is gate-dependent and $\mathcal{R}$ which is not discarded post-selection has perfect measurement.*

Proof. One can view a Fixed Time LFE sequence with rejected post-selection as inserting the code space projector $\Pi_{\mathbb{S}}$ at a fixed time in the LFE sequence. This projector can only decrease the signal at the time it was interleaved. Every other time step of the LFE sequence is just a character benchmarking sequence, and thus, the entire sequence is robust to gate-dependent noise. □

Robustness to gate-dependent noise remains to be shown when $\mathcal{R}_i$ is a QEC map.

Conjecture 6.4. *QEC at fixed time LFE minimizes the number of syndrome measurements and is robust to gate-dependent noise.*

There is currently a mismatch in the development of RB with gate-dependent noise and RB with time-dependent noise. As of right now, the best analysis of RB with time-dependent noise comes from [33]. However, some recent results from Helsen et al. showing further robustness to gate-dependent noise [16] and we anticipate that robustness guarantees for gate and time-dependent noise will be advanced in the near future. These advances will likely bolster the robustness of QEC at fixed time as an error characterization technique.

In fig. B.6 we give examples that show discarded post-selection at fixed time LFE produces upticks under gate-independent noise when the syndrome measurement is perfect. We feel this numerical evidence is sufficient to show that discarded post-selection at fixed time is inviable.

6.4 Summary

We showed Always LFE is robust to gate-independent noise, regardless of the form for the post-selection or recovery map $\mathcal{R}$.

1. In theorem 6.1 we expanded these results LFE procedures which the final $\mathcal{R}_{m+1}$ is $\mathcal{I}$.

We expanded the formalism of [5] by considering gate-dependent noise in Always LFE procedures.

1. Rejected post-selection always LFE is robust to gate-dependent noise by theorem 6.3.
2. QEC always LFE is robust to gate-dependent noise when $\mathcal{R}$ has perfect recovery by theorem 6.4.
3. QEC always LFE produces upticks under gate-dependent noise when $\mathcal{R}$ has perfect measurement by examples 6.1 and 6.2.

We attempted to minimize the number of syndrome measurements to look for procedures that may be implementable in the near-term.

1. QEC at the end LFE is the naive approach to testing QEC and minimizing syndrome measurements.
   (a) QEC at the End LFE produces upticks in gate-dependent noise. by example 6.4.
   (b) Rejected post-selection at the end LFE produces upticks in gate-dependent noise by example 6.4.
2. We suggested fixed time LFE as a near-term procedure that is robust.
   (a) QEC at fixed time LFE and rejected post-selection at the end LFE are robust to gate-independent noise. by theorem 6.1.
   (b) We suspect that the above results may hold under gate-dependent noise as well.

A summary of the proofs of this section can be found in table 6.1.

Non-Markovianity in Logical Fidelity Estimation

Recovery	Noise	Always	End	Fixed
QEC map	Indep.	✓ corrolary 6.1	× example 6.5	✓ corrolary 6.2
	Dep.	✓ if PR see theorem 6.4. × if PM see examples 6.1 and 6.2.	× N/A	? conjecture 6.4
Discarded PS	Indep.	✓ theorem 6.2	? fig. B.5	× fig. B.6
	Dep.	?	?	× N/A
Rejected PS	Indep.	✓ N/A	× example 6.4	✓ N/A
	Dep.	✓ theorem 6.3	× N/A	✓ lemma 6.3

Table 6.1: We summarize the general robustness properties of the LFE schemes in table 5.1 for both gate-independent (indep.) and gate-dependent (dep.) noise. In particular, we look for signatures of non-Markovianity in the form of upticks. ✓ indicates a proof that no upticks occur. In contrast, a × indicates the presence of upticks. Question marks denote an unknown result. N/A tells the reader that the result follows trivially from its gate-dependent or gate-independent counterpart. We note that Rejected Post-Selection always gives reliable results under the most general noise models. Question marks indicate open problems. Numerical evidence for each of the above conclusions can be found in appendix B, but if we only have numerical evidence, we reference the relevant figure in the table above.

Chapter 7

Numerical Evidence

7.1 Methods

We assume some familiarity with chapter 5.

We wrote a Mathematica program to implement all the LFE procedures in table 6.1 with perfect measurement in the [[5,1,3]] stabilizer code. We directly estimated the LFE sequence $q_G^{\mathbb{L}}$ by simulating the full matrix dynamics of eq. (5.1) and perfect syndrome measurement as described in eq. (2.8). We ignore $\Pi_{\phi^{\mathbb{L}}}$ from eq. (2.10) in $q_G^{\mathbb{L}}$ as it can be absorbed into Q when there are no measurement errors. We tested the 9 LFE procedures in table 6.1 under overrotation noise in 2 directions. The plots are grouped in appendix B by the implemented LFE procedure. When quantum error correction was applied, we used the minimum weight decoder [30].

Survival probabilities were estimated using either 1000 or 2000 simulations of $q_G^{\mathbb{L}}$ for each sequence length m and directly extracting the survival probability from the output state. We recognize that this data collection method may ignore issues with precision that may occur in a physical implementation. As such, we only make claims regarding the survival probability and assume we can sample a large number of points. When relevant, we fit the survival probability estimates to an exponential decay with details included in an inset table.

Obtaining enough samples of $p_{s_m}^{\mathbb{L}}$ was more difficult for discarded post-selection LFE. Discarded post-selection dramatically decreases the number of $p_{s_m}^{\mathbb{L}}$ sampled when errors are detected. We also found that discarded post-selection was very effective at correcting errors. So we needed more points to retain accuracy, and a stronger noise model was required to see decays. Thus we used 2000 sequences and doubled the error model's strength.

Generators of Implemented Stabilizer Code	
$\mathbb{S}^{gens}$	$\mathbb{T}^{gens}$
XZZXI	YIIIY
ZZXIX	YIIYI
ZXIXZ	YIYII
XIXZZ	YYIII

Table 7.1: The Paulis which generate $\mathbb{S}$ and $\mathbb{T}$ in the [[5,1,3]] stabilizer code implemented in our numerics. They are arranged so that the Paulis across corresponding columns anti-commute. $\mathbb{L}$ is generated from $X_L = X^{\otimes 5}$ and $Z_L = Z^{\otimes 5}$.

The logical Clifford group is not transversal in the [[5,1]] code, so we choose $\mathbb{G}$ to be the logical asymmetric group $\mathbb{A}_L$ generated by

$$\mathbb{A}_L^{gen} = \{X^{\otimes 5}, Z^{\otimes 5}, (HP)^{\otimes 5}\}. \tag{7.1}$$

The $m_{\phi^{\mathbb{L}}} = 1$ property required for character benchmarking was checked in section 2.3 for a single qubit. We can carry over this property to $\mathbb{A}_L$ because we can view the LFE procedure in the unencoded frame defined in fig. 3.1. In this frame, HP's commutation relations with the logical qubit's X, Y, and Z operators are invariant between the encoded and unencoded frames, so $\mathbb{A}_L$ must be a representation of $\mathbb{A}_1$ on the logical qubit. So if Q is in span($\mathbb{L}$), as required by LFE, there is $\phi^{\mathbb{L}}$ *s.t.* $m_{\phi^{\mathbb{L}}} = 1$.

For each of the forms of LFE in table 6.1, we estimated the survival probability for 2 noise channels, which we give below in the usual representation of $SU(2^n)$. The first is an X-overrotation on each of the physical qubits

$$U_{phys}^{Rot} = \prod_{j=1}^{5} e^{i\frac{(.03)}{2\pi}X_j}. \tag{7.2}$$

The second noise model considered is an overrotation in the direction of a destabilizer generator as in example 6.4

$$U_{\mathbb{T}}^{Rot} = e^{i\frac{(.03)}{2\pi}T^{gen}}. \tag{7.3}$$

Although destabilizer overrotations are somewhat adversarial, they show that QEC at the end LFE produces upticks.

Note the lack of an example with purely logical noise. We have purposely omitted it here as we feel [5] already addressed these noise models. We chose noise models that rotate the same amount, but only the rotation's direction is varied between experiments.

7.2 Numerics for Always LFE

We can break up our numerical evidence for QEC always LFE into two main points. The first is to show that one can commute the recovery to the end of the circuit. The second is to give data that compares QEC always LFE to the other forms of LFE considered in this thesis.

To test that one can commute recoveries through the LFE circuit, we have compared the output of measure always recovery at the end LFE and QEC always LFE under physical overrotations in fig. 7.1. Because the 3% physical rotations appear to be below the noise threshold, we increased the error rate to compare the two LFE schemes. We found that survival probabilities closely mirrored each other.

Unsurprisingly, we verified in fig. B.1 that QEC always LFE does not produce upticks under gate-dependent noise. It appeared that 3% physical overrotations were very close to the fault tolerance threshold. We found that destabilizer overrotations lowered the threshold below a 3% rotation, and the QEC always LFE procedure produced decays.

Rejected post-selection always LFE performed well for the noise models given. Figure B.7 shows that exponential decays can be measured and fitted in the usual way. No upticks appear.

Figure B.4 shows that discarded post-selection always LFE is insensitive to physical noise. We expected this insensitivity, as we determined in theorem 6.2 that for gate-independent noise, discarded post-selection always LFE is only responsive to undetectable errors in span$\{\mathbb{L}\}$.

7.3 Numerics for At the End LFE

In section 6.2 we found multiple examples where at the end LFE can produce upticks with gate-independent noise. To verify these examples, it was essential to focus on noise models that enact destabilizers. A brief flavor for our numerical results is given in fig. 7.2.

In particular, fig. B.2 shows that QEC at the end LFE may have large upticks under a small destabilizer overrotation noise model. However, we note that under physical overrotations, the decay was clear and successfully carried out the exponential fit. So this procedure may still have limited use for characterizing QEC when one has certain guarantees on the nature of the noise being benchmarked.

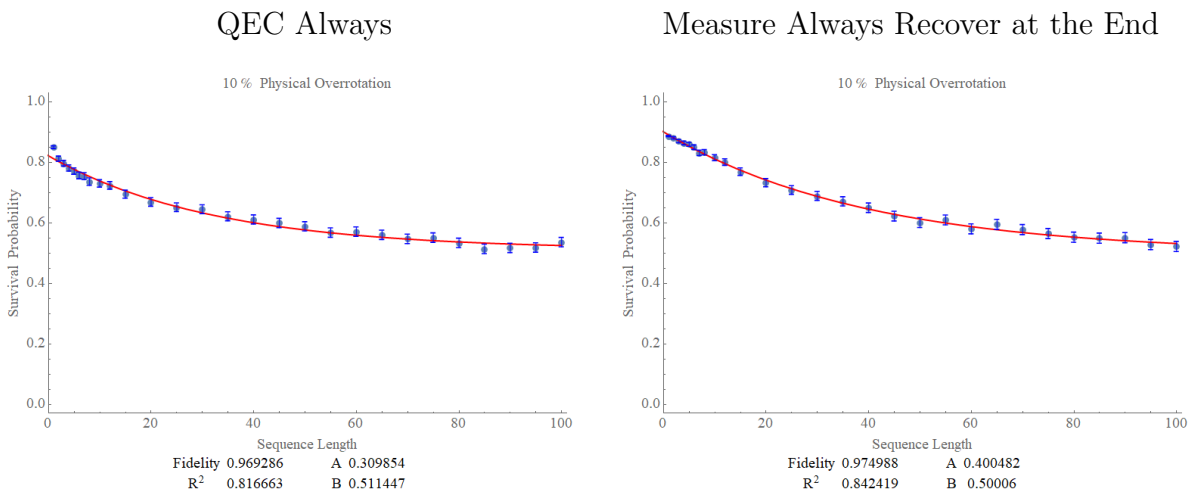


Figure 7.1: We plotted survival probabilities for QEC always LFE and measure always recover at the end LFE to see if commuting recoveries through the circuit changed the infidelity. Note that we have increased the severity of physical overrotations to 10%. We plotted the survival probability for each LFE procedure in blue, and we show the exponential fit model as a solid red line. We find that in each case, the fidelities agree in the first two digits.

Figure B.8 shows rejected post-selection at the end LFE suffers from even worse oscillations than QEC at the end. So it can be more or less ruled out as a viable procedure. Rejected post-selection is much better suited to always LFE and at fixed time LFE.

In sharp contrast to the above procedures, ignored post-selection at the end is unaffected by destabilizer noise. So we omitted this behavior from fig. B.5; however, one can still observe non-standard decays due to physical X-overrotations. This insensitivity to destabilizers could be anticipated by noting that the destabilizers trigger syndromes which are completely uncorrelated to the logical output. Thus the post-selections are uncorrelated to the logical state, and the logical output is unaffected by the post-selections. However, this is not the case for the physical X-overrotation noise, where we observe an intriguing non-standard decay.

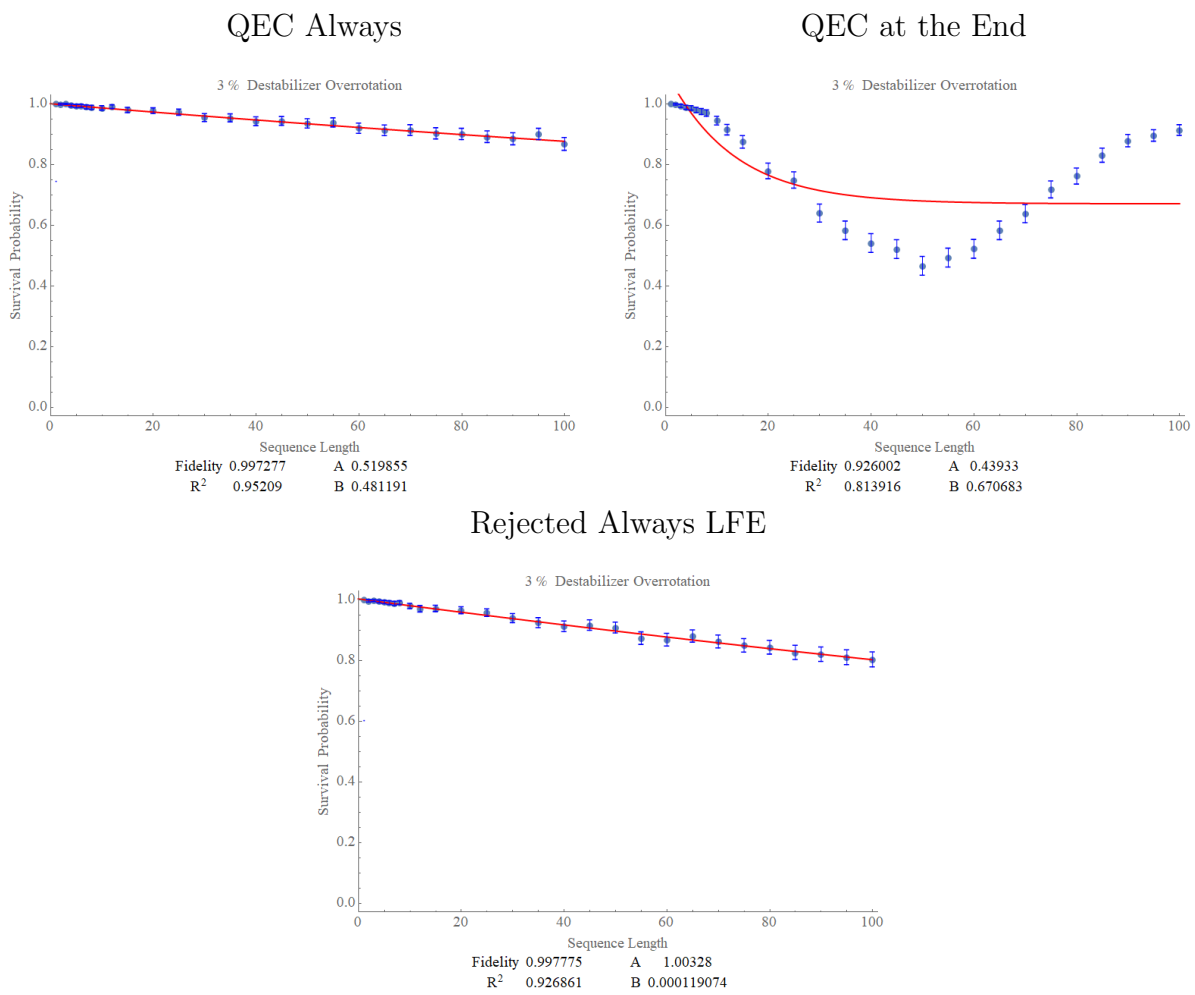


Figure 7.2: 3 LFE procedures were simulated in the [[5,1]] Code with (3%) overrotations about the X direction on each physical qubit and measured with 1000 points per sequence length. We plotted the survival probability for each LFE procedure in blue, and we show the exponential fit model as a solid red line. The minimum weight decoding was utilized to perform error correction. We can see that in the cases of QEC always LFE and rejected post-selection always LFE no upticks are observed as predicted in chapter 6. For the case QEC at the end LFE, we can see clear signs of non-Markovianity in the form of upticks.

7.4 Numerics for Fixed Time LFE

We implemented fixed time LFE with $\mathcal{R}$ applied at the 50^{th} gate. The recovery generally shows up in each plot as a decrease in precision, accompanied by a decrease in survival probability.

QEC at fixed time did not produce upticks in any of the implemented noise models. Suggesting that it may be an effective method of characterizing how much QEC is affecting the implemented state. The dip in survival probability might be interpreted as an indicator of how much QEC affects the size of the error. A large drop in survival probability indicates that QEC significantly increases the size of errors. A fundamental property to characterize fault tolerance.

Rejected post-selection at fixed time is relatively trivial. It merely reports the probability of detecting an error at the fixed point in the sequence length where post-selection is interleaved. Nevertheless, this procedure does not uptick and so makes a viable assessment of a stabilizer code.

From fig. B.6, one quickly notices that upticks occur in discarded post-selection at fixed time LFE. So we can quickly rule out discarded post-selection at fixed time LFE with this numerical evidence. Although we note that these experiments demonstrate that discarded post-selection's restorative powers are remarkable, the survival probability can sometimes jump very close to 1 when the post-selection is performed as observed in fig. B.6.

7.5 Summary

We implemented simulations in Mathematica to test the 9 LFE protocols of table 6.1.

1. One can see the results our numerics for each procedure in table 7.2.
2. We verified in fig. 7.1 that we could delay recovery to the end of the LFE sequence without changing the fidelity.
3. Discarded post-selection at the end LFE appears to produce non-standard decays in fig. B.5.

Summary of Numerics

	Always	At the End	Fixed Time
QEC	fig. B.1 no upticks	fig. B.2 shows upticks	fig. B.3 no upticks
Discarded PS	fig. B.4 no upticks	fig. B.5 no upticks	fig. B.6 shows upticks
Rejected PS	fig. B.7 no upticks	fig. B.8 shows upticks	fig. B.9 no upticks

Table 7.2: Table summarizing the numerical results available in appendix B for each of the LFE protocols considered in tables 5.1 and 6.1. We focus on the presence of upticks, a standard signature of non-Markovianity, in the data. We prune inviable procedures such as QEC at the end LFE or rejected post-selection at the end LFE by requiring that the observed noise be Markovian.

Chapter 8

Conclusions

We defined logical fidelity estimation, a natural class of benchmarks for quantum codes. We found that their outputs may indicate non-Markovian noise, summarizing the results in table 6.1. Upticks in QEC at the End LFE can be well understood using the concept of degenerate irreps. However, the source of upticks in QEC Always LFE remains more enigmatic, owing to dependence on previously applied gates in the sequence. Our analysis shows that rejected post-selection always LFE is the natural benchmark which is immune to both kinds of non-Markovianity discussed above.

We also amended some of the work in [5] by showing that one can indeed apply decodings in post-processing. However, a method to extend these results to gate-dependent noise is unclear. Using these results, we concluded that one should not assume a perfect recovery; instead, it is necessary to consider weakened assumptions such as perfect measurement.

We found that if one wishes to reduce the number of syndrome measurements and test QEC, QEC at fixed time LFE has distinguished itself as an intuitive, robust option. This result is in contrast to QEC at the end, which produces upticks in the presence of destabilizer noise. However, the fit model for QEC at fixed time LFE is non-standard. Thus, finding a method of interpreting the results of QEC at fixed time LFE is an intriguing open problem.

Discarded post-selection seemed to be harder to characterize. We can think of discarded post-selection always as maximizing the survival probability $p_{s_m}^{\mathbb{L}A}$ because the scheme only lets through undetectable errors. In discarded post-selection at the end LFE, detectable errors are allowed to build up into undetectable errors. Characterizing how detectable errors transition to undetectable errors will be crucial to preserving the encoded state, so identifying the fit model when $\mathcal{R}$ is a discarded post-selection is a significant open problem.

References

[1] Scott Aaronson and Daniel Gottesman. Improved simulation of stabilizer circuits. *Physical Review A - Atomic, Molecular, and Optical Physics*, 2004.

[2] Panos Aliferis, Daniel Gottesman, and John Preskill. Quantum accuracy threshold for concatenated distance-3 codes. *Quantum Information and Computation*, 2006.

[3] Barnhill, Marie. Extensions of Randomized Benchmarking. 2015.

[4] Christopher Chamberland, Joel Wallman, Stefanie Beale, and Raymond Laflamme. Hard decoding algorithm for optimizing thresholds under general Markovian noise. *Physical Review A*, 2017.

[5] Joshua Combes, Christopher Granade, Christopher Ferrie, and Steven T. Flammia. Logical Randomized Benchmarking. 2017.

[6] Hillary Dawkins, Joel Wallman, and Joseph Emerson. Combining T1 and T2 estimation with randomized benchmarking and bounding the diamond distance. *Physical Review A*, 2020.

[7] Joseph Emerson, Robert Alicki, and Karol Zyczkowski. Scalable noise estimation with random unitary operators, 2005.

[8] Alexander Erhard, Joel J. Wallman, Lukas Postler, Michael Meth, Roman Stricker, Esteban A. Martinez, Philipp Schindler, Thomas Monz, Joseph Emerson, and Rainer Blatt. Characterizing large-scale quantum computers via cycle benchmarking. *Nature Communications*, 2019.

[9] Jay M. Gambetta, A. D. Córcoles, S. T. Merkel, B. R. Johnson, John A. Smolin, Jerry M. Chow, Colm A. Ryan, Chad Rigetti, S. Poletto, Thomas A. Ohki, Mark B. Ketchen, and M. Steffen. Characterization of addressability by simultaneous randomized benchmarking. *Physical Review Letters*, 2012.

[10] Daniel Gottesman. Stabilizer Codes and Quantum Error Correction. 5 1997.

[11] Daniel Gottesman. Theory of fault-tolerant quantum computation. *Physical Review A - Atomic, Molecular, and Optical Physics*, 1998.

[12] Daniel Gottesman. *An introduction to quantum error correction and fault-tolerant quantum computation*. 2010.

[13] W T Gowers and O Hatami. Inverse and stability theorems for approximate representations of finite groups. *Sbornik: Mathematics*, 2017.

[14] Robin Harper and Steven T. Flammia. Fault-Tolerant Logical Gates in the IBM Quantum Experience. *Physical Review Letters*, 122(8):1–8, 2019.

[15] A. K. Hashagen, S. T. Flammia, D. Gross, and J. J. Wallman. Real Randomized Benchmarking. *Quantum*, 2:85, 2018.

[16] J Helsen, I Roth, E Onorati, A Werner, and J Eisert. A general framework for randomized benchmarking. *arXiv: Quantum Physics*, 2020.

[17] Jonas Helsen, Xiao Xue, Lieven M.K. Vandersypen, and Stephanie Wehner. A new class of efficient randomized benchmarking protocols. *npj Quantum Information*, 2019.

[18] Pavithran Iyer. *A critical analysis of quantum error correction methods for realistic noise processes*. PhD thesis, 2018.

[19] Pavithran Iyer and David Poulin. Hardness of decoding quantum stabilizer codes. *IEEE Transactions on Information Theory*, 2015.

[20] H. G. Katzgraber and R. S. Andrist. Stability of topologically-protected quantum computing proposals as seen through spin glasses. In *Journal of Physics: Conference Series*, 2013.

[21] Morten Kjaergaard, Mollie E. Schwartz, Jochen Braumüller, Philip Krantz, Joel I.J. Wang, Simon Gustavsson, and William D. Oliver. Superconducting Qubits: Current State of Play, 2020.

[22] Nick Lord, W. Fulton, and J. Harris. Representation Theory: A First Course. *The Mathematical Gazette*, 1995.

[23] Easwar Magesan, Jay M. Gambetta, B. R. Johnson, Colm A. Ryan, Jerry M. Chow, Seth T. Merkel, Marcus P. Da Silva, George A. Keefe, Mary B. Rothwell, Thomas A. Ohki, Mark B. Ketchen, and M. Steffen. Efficient measurement of quantum gate error by interleaved randomized benchmarking. *Physical Review Letters*, 2012.

[24] Seth T. Merkel, Emily J. Pritchett, and Bryan H. Fong. Randomized Benchmarking as Convolution: Fourier Analysis of Gate Dependent Errors. 4 2018.

[25] David Poulin. Optimal and efficient decoding of concatenated quantum block codes. *Physical Review A - Atomic, Molecular, and Optical Physics*, 2006.

[26] Timothy J. Proctor, Arnaud Carignan-Dugas, Kenneth Rudinger, Erik Nielsen, Robin Blume-Kohout, and Kevin Young. Direct Randomized Benchmarking for Multiqubit Devices. *Physical Review Letters*, 2019.

[27] Benjamin Rahn, Andrew C. Doherty, and Hideo Mabuchi. Exact performance of concatenated quantum codes. *Physical Review A - Atomic, Molecular, and Optical Physics*, 2002.

[28] Yuval R. Sanders, Joel J. Wallman, and Barry C. Sanders. Bounding quantum gate error rate based on reported average fidelity. *New Journal of Physics*, 2016.

[29] Mohan Sarovar, Timothy Proctor, Kenneth Rudinger, Kevin Young, Erik Nielsen, and Robin Blume-Kohout. Detecting crosstalk errors in quantum information processors. *Quantum*, 4:321, 9 2020.

[30] Barbara M. Terhal. Quantum error correction for quantum memories, 2015.

[31] M Tinkham. *Group Theory and Quantum Mechanics*. Dover Books on Chemistry and Earth Sciences. Dover Publications, 2003.

[32] M. Veldhorst, J. C.C. Hwang, C. H. Yang, A. W. Leenstra, B. De Ronde, J. P. Dehollain, J. T. Muhonen, F. E. Hudson, K. M. Itoh, A. Morello, and A. S. Dzurak. An addressable quantum dot qubit with fault-tolerant control-fidelity. *Nature Nanotechnology*, 2014.

[33] Joel J. Wallman and Steven T. Flammia. Randomized benchmarking with confidence. *New Journal of Physics*, 2014.

[34] Chenyang Wang, Jim Harrington, and John Preskill. Confinement-Higgs transition in a disordered gauge theory and the accuracy threshold for quantum memory. *Annals of Physics*, 2003.

[35] Christopher J. Wood, Jacob D. Biamonte, and David G. Cory. Tensor networks and graphical calculus for open quantum systems. *Quantum Information and Computation*, 2015.

APPENDICES

Appendix A

Usage of Provided Code

This appendix is meant to be a simple guide for replicating the results contained in this thesis. Downloads are available from GitHub https://github.com/mathmeetsmusic/Thesis_Numerics. Although full documentation is not currently available, questions about further usage can be directed to mathmeetsmusic@gmail.com.

Begin by downloading and opening the "Thesis_Numerics.nb" file. Once open, scroll down to the section labeled "benchmark," and within that section, one should find four subsections, each containing code for a different set of procedures. The first three concern this thesis directly. To produce the results of appendix B below, run each of the cells in the subsection labeled "Plot Sequences for Appendix B" in the order provided. To plot the results of fig. 7.1, one can run the subsection labeled "Test to see if Recovery can be Delayed" in the order provided. Finally, the discarded post-selection plots with heightened error rates can be re-created by running all the cells in the subsection labeled "Discarded Post-Selection with Non-standard Decay."

The length of run-time will vary dramatically based on the error model and sequence type. Running on 8 cores, each of the above procedures will likely finish in under 4 hours of computation. Note that the default precision of benchmarks is much lower than the precision provided in this thesis. To recreate the data here, one will have to adjust the *ncircuit* variable to the desired precision. We have included real-time estimates of total run-time for each of the types of LFE. However, the accuracy and frequency of updates may vary. The resulting plots should be automatically placed in the current directory of the .nb file.

Appendix B

Figures for Numerics

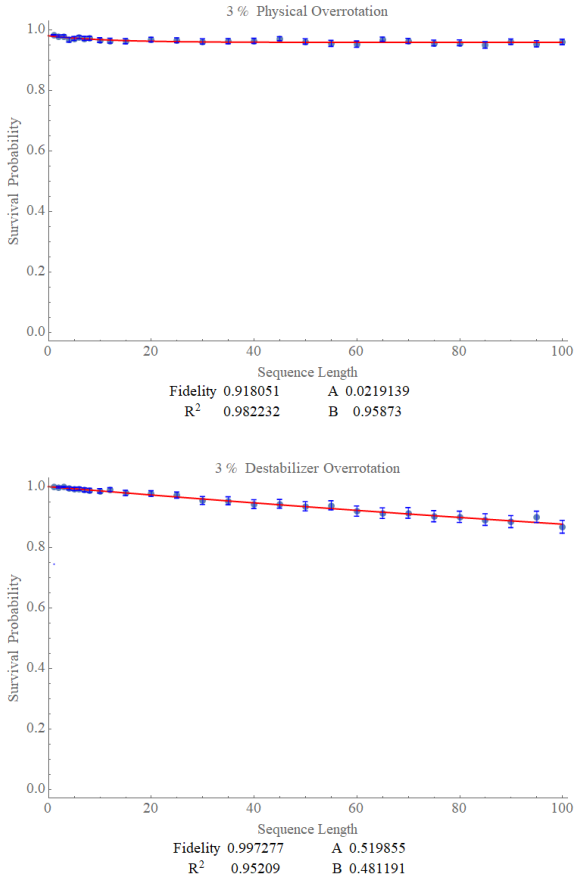


Figure B.1: QEC Always LFE was simulated in the [[5,1]] Code with (3%) overrotations about the 2 separate axis given by physical overrotation $U_{phys}^{Rot} = \prod_{j=1}^{5} e^{i\frac{(.03)}{2\pi}X_j}$ (above) and a destabilizer overrotation $U_{\mathbb{T}}^{Rot} = e^{i\frac{(.03)}{2\pi}T^{gen}}$ (below) with 1000 points per sequence length. We plotted the survival probability for each LFE procedure in blue, and we show the exponential fit model as a solid red line. The minimum weight decoding was utilized to perform error correction. One can see that we are apparently below the threshold for physical overrotations, so the estimation of fidelity is inaccurate. For more meaningful fits physical overrotation noise, see fig. 7.1. No upticks were observed in either case as predicted in chapter 6

QEC at the End LFE

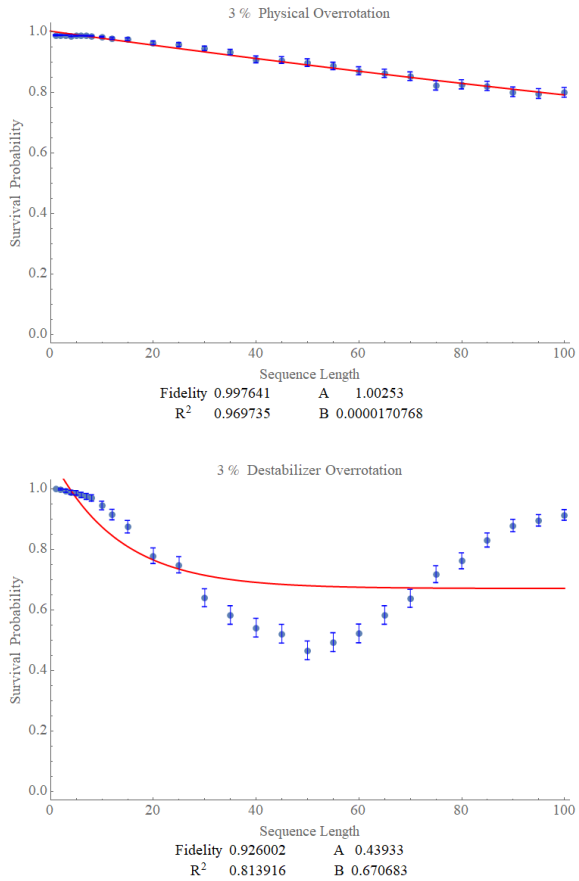


Figure B.2: QEC at the End LFE was simulated in the [[5,1]] Code with (3%) overrotations about the 2 separate axis given by physical overrotation $U_{phys}^{Rot} = \prod_{j=1}^{5} e^{i\frac{(.03)}{2\pi}X_j}$ (above) and a destabilizer overrotation $U_{\mathbb{T}}^{Rot} = e^{i\frac{(.03)}{2\pi}T^{gen}}$ (below) with 1000 points per sequence length. We plotted the survival probability for each LFE procedure in blue, and we show the exponential fit model as a solid red line. The minimum weight decoding was utilized to perform error correction. In the case of destabilizer overrotations, large oscillations are present in the survival probability. So QEC should not be inserted at the end of the sequence to have robustness to gate-independent noise.

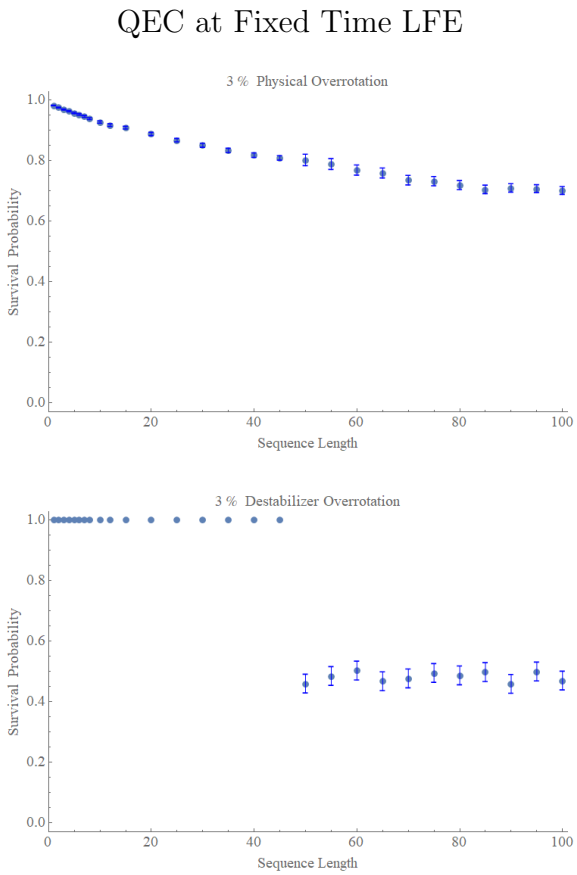


Figure B.3: QEC at Fixed Time LFE was simulated in the [[5,1]] Code with (3%) over-rotations about the 2 separate axis given by physical overrotation $U_{phys}^{Rot} = \prod_{j=1}^{5} e^{i\frac{(.03)}{2\pi}X_j}$ (above) and a destabilizer overrotation $U_{\mathbb{T}}^{Rot} = e^{i\frac{(.03)}{2\pi}T^{gen}}$ (below) with 1000 points per sequence length. We plotted the survival probability for each LFE procedure in blue, and we show the exponential fit model as a solid red line. The minimum weight decoding was utilized to perform error correction. We inserted QEC after the 50th gate. The large jump in the destabilizer noise model below an artifact of the destabilizer noise model not affecting the logical state. When QEC acts at gate 50, it can only harm the logical qubit, so we see a large decrease in fidelity. As expected, we observe no upticks in either noise model.

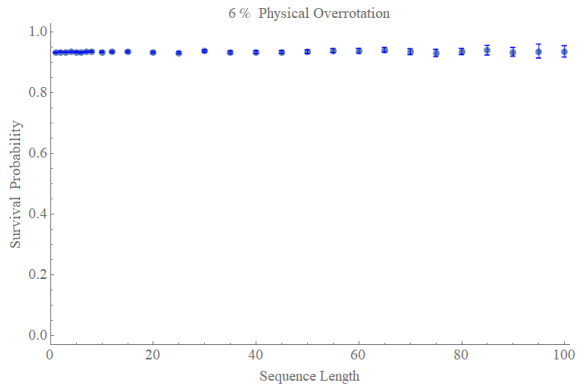


Figure B.4: Discarded post-selection always LFE was simulated in the [[5,1]] Code with (6%) overrotations about the X-axis for each physical qubit given by $U_{phys}^{Rot} = \prod_{j=1}^{5} e^{i\frac{(.03)}{2\pi}X_j}$ with 2000 points per sequence length. We plotted the survival probability for each LFE procedure in blue. Note the general insensitivity to the noise models presented and supports the conclusions of theorem 6.2. Where only uncorrectable noise in span$\mathbb{L}$ will explicitly induce decays. In a nutshell, this procedure seems to maximize the code's sensitivity and maximally purifies the signal by excluding any sequence that may contain errors.

Discarded Post-Selection at the End LFE

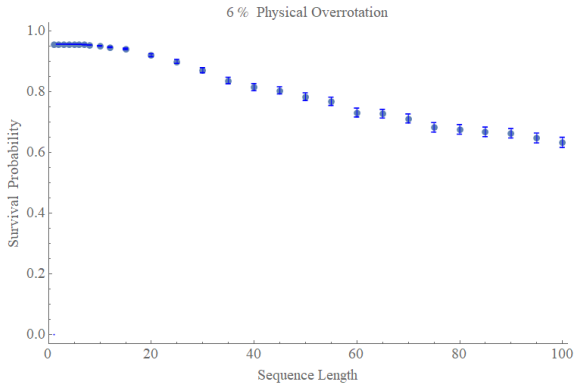


Figure B.5: Discarded post-selection at the end LFE was simulated in the [[5,1]] Code with (6%) overrotations about the X-axis for each physical qubit given by $U_{phys}^{Rot} = \prod_{j=1}^{5} e^{i\frac{(.03)}{2\pi}X_j}$ with 2000 points per sequence length. We plotted the survival probability for each LFE procedure in blue. One can observe the small deviations from the exponential decay, with the survival probability appearing as an ostensibly concave function of sequence length. Confirming the intuition of [14] but still showing no signs of upticks. Note that decays appear when the noise is a destabilizer-logical overrotations.

Discarded Post-Selection at Fixed Time LFE

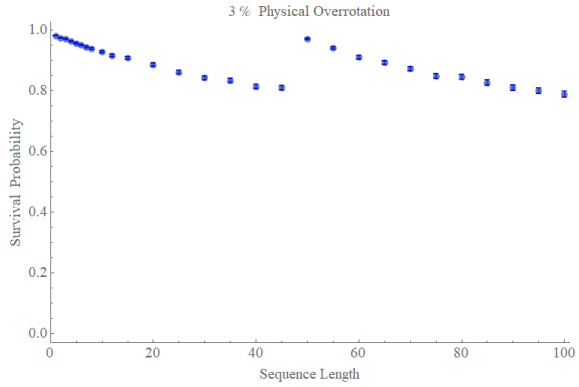


Figure B.6: Discarded post-selection at fixed time LFE was simulated in the [[5,1]] Code with (6%) overrotations about the x-axis for each physical qubit given by $U_{phys}^{Rot} = \prod_{j=1}^{5} e^{i\frac{(.03)}{2\pi}X_j}$ with 2000 points per sequence length. We plotted the survival probability for each LFE procedure in blue. We inserted discarded post-selection after the 50th gate. Clear upticks can be seen at gate 50 when $\mathcal{R}$ is interleaved due to the exclusion of sequences with errors.

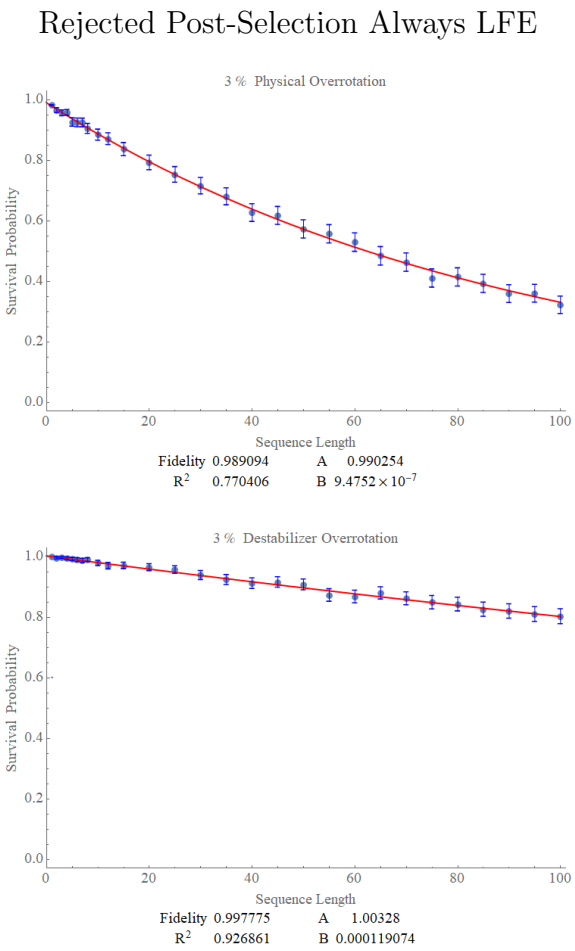


Figure B.7: Rejected post-selection always LFE was simulated in the [[5,1]] Code with (3%) overrotations about the 3 separate axis given in chapter 7 and measured with 1000 points per sequence length. We plotted the survival probability for each LFE procedure in blue, and we show the exponential fit model as a solid red line. Here we have tested theorem 6.3 by defining R_i to be a post-selection. We found that, like the case of QEC, no upticks were observed.

Rejected Post-Selection at the End LFE

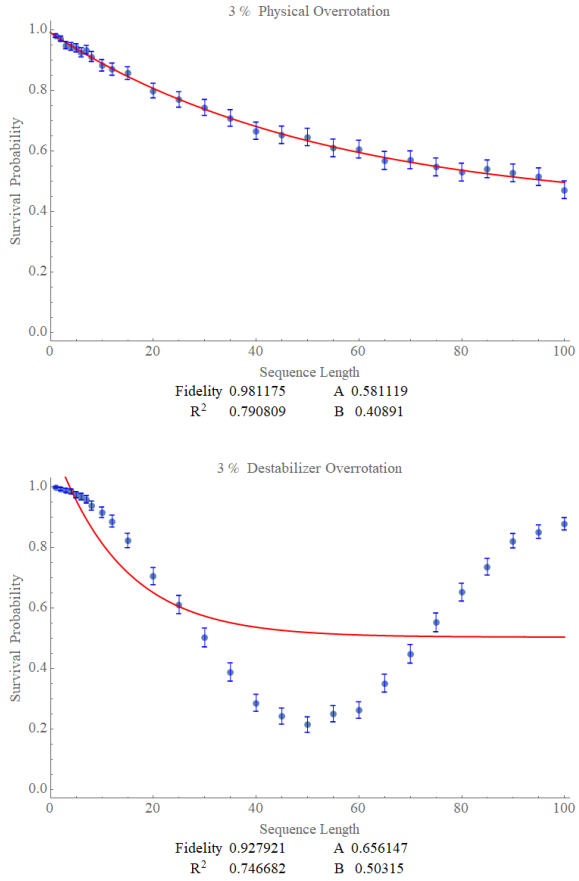


Figure B.8: Rejected post-selection at the End LFE was simulated in the [[5,1]] Code with (3%) overrotations about the X-axis given in chapter 7 and measured with 1000 points per sequence length. We plotted the survival probability for each LFE procedure in blue, and we show the exponential fit model as a solid red line. In the case of destabilizer overrotations, large oscillations are present in the survival probability. So rejected post-selection should not be inserted at the end of the sequence to have robustness to gate-independent noise.

Rejected Post-Selection at Fixed Time LFE

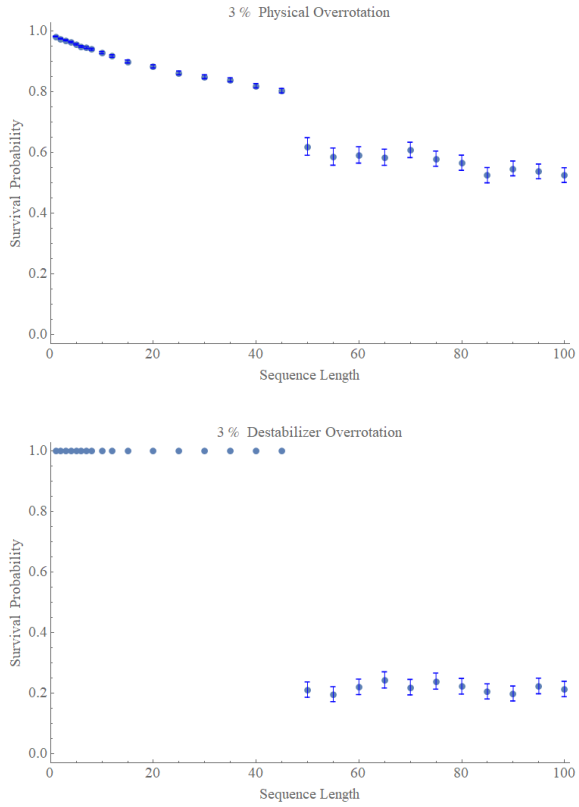


Figure B.9: Rejected post-selection at Fixed Time LFE was simulated in the [[5,1]] code with (3%) overrotations about the X-axis given in chapter 7 and measured with 1000 points per sequence length. We inserted a rejected post-selection after the 50th gate. This procedure would mostly quantify the probability that the code rejects a state by measuring the jump in survival.