OPTIMIZED EXPERIMENT DESIGN AND ANALYSIS FOR FULLY RANDOMIZED BENCHMARKING

by

Alex Kwiatkowski

B.S., Stanford University, 2017

M.S., Stanford University, 2017

A thesis submitted to the

Faculty of the Graduate School of the

University of Colorado in partial fulfillment

of the requirements for the degree of

Doctor of Philosophy

Department of Physics

2024

Committee Members:

Emanuel Knill, Chair

Scott Glancy

Daniel Slichter

Joshua Combes

Ana Maria Rey

Shuo Sun

Kwiatkowski, Alex (Ph.D., Physics)

Optimized experiment design and analysis for fully randomized benchmarking

Thesis directed by E. Knill and S. Glancy

Randomized benchmarking (RB) is a widely used strategy to assess the quality of available quantum gates in a computational context. RB involves applying known random sequences of gates to an initial state and using a final measurement step to determine 'success' or 'failure' for each trial. The probabilities of success and failure over many trials can be used to determine an effective depolarizing error per step of the sequence, which is a metric of the gate quality. This thesis investigates the advantages of fully randomized benchmarking, where a new random sequence is drawn for each experimental trial. The advantages of full randomization include smaller confidence intervals on the inferred step error, the ability to use maximum likelihood analysis without heuristics, straightforward optimization of the sequence lengths, and the ability to model and measure behaviors that go beyond the typical assumption of time-independent error rates. We discuss concrete models of time-dependent or non-Markovian errors that generalize the basic RB model of a single exponential decay of the success probability. For any of these models, we implement an experiment-design protocol to minimize the uncertainty of the estimated parameters with a fixed constraint on the time for the complete experiment. Furthermore, we consider several previously published experiments and determine the potential for improvements with optimized full randomization. We observe such improvements in Clifford randomized benchmarking experiments run by our collaborators on a single trapped ion qubit at the National Institute of Standards and Technology (NIST). We also provide native-gate decompositions for future one and two qubit RB experiments in trapped ions, including an implementation of a convenient two-qubit two-design that has not been used previously for RB. Finally, we study non-Markovian errors in RB, where error processes involve an interaction with an arbitrary quantum environment. In the case of gate-independent non-Markovian errors, we put a concrete bound on the non-monotonicity of success probabilities.

Acknowledgements

I am incredibly grateful to Manny Knill and Scott Glancy for all of their instruction, advice, and encouragement throughout my graduate school years. I would also like to thank all of my collaborators, both in theory and experiment, including Shawn Geller, Akshay Seshadri, Mohammad Alhejji, May An van de Poll, Ezad Shojaee, Sristy Agrawal, Marcel Mazur, Aiko Kyle, Karl Mayer, Curtis Rau, Ariel Shlosberg, Shlomi Kotler, Raghavendra Srinivas, Daniel Slichter, Didi Liebfried, John Teufel, Kevin Silverman, Konrad Lehnert, Laurent Stephenson, Hannah Knaack, Alejandra Collopy, Stephen Erickson, Christina Bowers, Adam Brandt, Andrew Wilson, Jenny Wu, Pan-yu Hou, Josh Combes, Ryan DeCrescent, Poolad Imany, and Zixuan Wang.

Finally, I would like to give special thanks to my parents, sisters, and all of my other family members for helping me learn and grow throughout my entire life.

Contents

Chapter

Tables

Table

Figures

Figure

Chapter 1

Introduction

Benchmarking quantum gates is an important task for the design, development, and characterization of a quantum processor [20]. Randomized benchmarking is a widely-used method to benchmark gates in a computational context that takes advantage of long sequences of gates to efficiently gain statistical information even when large errors in state preparation and measurement (SPAM) are present [23, 31, 38, 39, 41, 42, 44, 48, 49, 51, 59, 69]. In fact, in recent years randomized benchmarking has become a standard method of demonstrating high-quality control over quantum systems that are intended to be used for computation and information processing purposes [24, 47, 52]. In general, a randomized benchmarking trial consists of state preparation, followed by a random sequence of steps drawn from a carefully chosen distribution, followed by a measurement indicating 'success' or 'failure' for the trial depending on whether or not the nominal outcome is observed. The simplest example is standard randomized benchmarking, where each step is drawn from a unitary two-design [14, 51]. If each step is modeled as a perfect unitary gate followed by an error channel, and if that error channel is independent of time, sequence position, and the gate that is applied, then the probability of the 'success' outcome decays exponentially as a function of sequence length [49]. The rate of this exponential decay is interpreted as the average fidelity of a step in a computational context. In the case where the error channels are depolarizing channels that are time-independent but possibly gate-dependent, the exponential decay rate is indeed the average fidelity of a step. However, in the case of gate-dependent unitary errors, the interpretation of the exponential decay rate is slightly more complicated [59, 69]. In addition to standard randomized

benchmarking, a wide variety of other randomized benchmarking variants have been proposed and studied in recent years. For example, some variants are designed to characterize the fidelity of gates that aren't compatible with two-designs [9], and other variants are designed for inference of system parameters other than the average fidelity of steps [57, 68]. In many such situations the behavior of the success probability as a function of sequence length can deviate from a single exponential decay [9, 32, 37, 40].

This thesis focuses on our recent work related to randomized benchmarking. One topic of particular emphasis is fully randomized benchmarking, which we introduce in detail in the following paragraphs. Due to hardware limitations, current experimental implementations of all variants of randomized benchmarking intentionally repeat each random sequence many times to collect statistics about the probability of error. Here we study fully randomized benchmarking, where a new random sequence is drawn for each experimental trial. When we compare fully randomized benchmarking to randomized benchmarking with intentionally repeated sequences, we find several concrete advantages of fully randomized benchmarking. Broadly, these advantages come from the fact that, for an arbitrary error channel, a fully randomized benchmarking experiment is statistically indistinguishable from one where the error channel is a depolarizing channel with the same fidelity. The same is not true if sequences are intentionally repeated. If sequences are intentionally repeated, there is a non-trivial distribution of success probabilities over the possible random sequences. This distribution is determined by the true error channel of the gates, and properties of this distribution are observable in the statistics of repeated sequences.

A more detailed summary of the advantages of fully randomized benchmarking is as follows. First, a randomized benchmarking experiment that repeats random sequences will generally have a larger uncertainty in the step error when compared to the same experiment where the sequences are fully randomized. The larger uncertainty comes from increased variance in the estimate of success probability at each sequence length due to the distribution of fidelities over all possible random sequences. This has been studied in Ref. [70] where the authors recommend designing experiments with relatively short sequence lengths in order to mitigate the effect of repeating random sequences.

They also argue that the effect from not fully randomizing is small for Pauli error channels, but point out that their arguments do not apply to unitary error channels. In fact, we analyze previously published randomized benchmarking experiments and find evidence in some cases that a significantly smaller uncertainty could have been obtained if the experiment were fully randomized.

A second advantage of fully randomized benchmarking is that the choice of the set of sequence lengths and the choice of the number of trials for each sequence length can be optimized in a straightforward way to maximize the information gained during the experiment. In Chapter 6 we provide a detailed discussion the numerical procedure that we use to optimize the design of a fully randomized benchmarking experiment. This optimization procedure minimize the uncertainty in the step error according to a pre-chosen statistical model that is linearized around a best guess for the step error and other parameters. We also explain how this optimization is equivalent to a particular form of Fisher optimization for multiple parameters. In experiments that do not fully randomize, optimization of the experiment design requires knowledge of least-squares weights that are generally unknown and depend on the true error model. Previous work about optimization strategies for randomized benchmarking can be found in Refs. [34, 36, 43]. Ref. [36] provides a heuristic optimization strategy for the basic exponential decay model of randomized benchmarking and suggests choosing a short sequence length and a long sequence length at half the inverse step error. Ref. [34] addresses optimization for a Bayesian inference procedure, and Ref. [43] discusses optimization strategies for non-fully-randomized benchmarking where a particular model is chosen for the variance of success probabilities over random sequences.

A third advantage of fully randomized benchmarking is that the step error can be inferred by using maximum likelihood in a straightforward way. In contrast, in experiments that intentionally repeat sequences the step error is typically inferred by means of a weighted-least-squares fit with weights that are a priori unknown, which complicates the interpretation of confidence intervals. In Chapter 7 we describe the statistical analysis we use to infer the step error, which consists of maximum likelihood inference and statistical bootstrapping to obtain confidence intervals. We describe an empirical likelihood ratio test that we use to possibly reject the basic model of a single exponential

decay and demonstrate it on simulated data. Also, we analyze the improvement in uncertainty from fully randomized benchmarking in terms of the underlying distribution of success probabilities over random sequences at a fixed sequence length. We give evidence that the improvement in uncertainty can be significant when the underlying errors are unitary. Furthermore, we make comparisons between previously published randomized benchmarking experiments and optimized fully randomized benchmarking experiments and demonstrate that noticeable improvements in uncertainty are possible.

Finally, fully randomized benchmarking allows for a straightforward analysis of the simplest time-dependent or sequence-position-dependent errors. In fully randomized benchmarking, the only effect of these errors, or of any non-Markovian errors, is to modify the behavior of the success probability as a function of sequence length. We introduce a nested sequence of statistical models for fully randomized benchmarking that can be used to detect this modifed behavior in a straightforward way. Furthermore, the reduced uncertainty of an optimized fully randomized experiment allows for the detection of modified behavior with increased statistical significance.

In this thesis we present an experimental demonstration of fully randomized benchmarking on a single trapped ion qubit, performed by our experimental collaborators at NIST. As part of this demonstration we perform a comparison between randomized benchmarking with uniformly chosen sequence lengths and repeated sequences, fully randomized benchmarking with uniformly chosen sequence lengths, and optimized fully randomized benchmarking, under otherwise equal conditions. We find that substantial reductions in uncertainty are possible, as a result of both fully randomizing and optimizing the experiment design. We also ran a statistical analysis to detect deviations from a single exponential decay that would indicate possible time-dependent behavior. As a result of our analysis we find weak evidence of such deviations.

Practical implementations of randomized benchmarking in a particular platform often require specific decompositions of the required gates into sets of natively available gates. Here we address such gate decompositions for one and two qubit randomized benchmarking. These decompositions are designed for an ion-trap experiment, but will apply to systems with similar native gate sets.

The particular decompositions that we study are designed to both facilitate a fully-randomized implementation and to reduce the complexity of the required experimental control to implement the sequences. In these gate decompositions we assume that the native gateset for single qubits consists of $\pi/2$ rotations about major equatorial axes. For two qubits, we assume that the native gateset consists of collective single qubit rotations, a Clifford entangling gate, and a differential Z gate that can perform a Z rotation on the two qubits with different rotation angles. As part of this study, we describe a two-qubit Clifford two-design that uses a fixed pattern of two Clifford entangling gates per step. We also provide a decomposition into native gates where the only differences between elements of the two-design come from the choice of rotation axis of the single-qubit gates that surround the fixed pattern of differential Z gates and Clifford entangling gates. To our knowledge, this two-design has not yet been used to perform two-qubit randomized benchmarking. This work on gate decompositions for single-qubit and two-qubit randomized benchmarking will be useful for future work on implementing fully randomized benchmarking in ion traps.

Furthermore, this thesis also addresses non-Markovian errors in randomized benchmarking and the behavior of the success probability when non-Markovian errors are present. Traditionally, the theoretical study of randomized benchmarking focuses on the case of Markovian errors, meaning that each physical operation is modeled by an error process acting only on the system of interest. However, in any realistic implementation there is a chance that a physical implementation of an ideal operation will include non-Markovian errors, which are errors that must be modeled by an error process acting on both the system of interest and an additional quantum memory system that is preserved throughout the sequence of operations. The question of characterization techniques in the presence of non-Markovianity has been of recent interest, especially because non-Markovian effects are known to be present when error correcting codes are used. In particular, many researchers are currently interested in understanding the behavior of randomized benchmarking when the errors are non-Markovian [10, 13, 28]. One potential effect that non-Markovianity can have on a randomized benchmarking experiment is to cause the success probability to rise and fall as the sequence length increases. This behavior is sometimes referred to as revivals in the success probability [10], and is

impossible if the errors are Markovian. Here we give a concise study of non-Markovianity in Clifford randomized benchmarking in the case where the Clifford gates acting on the system of interest are assumed to have gate-independent errors. For any given set of success probabilities as function of sequence length, we provide a concrete quantification of revivals. In the case of gate-independent errors on Clifford gates we prove a tight upper bound on this quantity, independent of sequence length.

This thesis is organized as follows. In Chapter 2 we introduce the basic notions of Pauli observables, general quantum channels, Pauli channels, depolarizing channels, Clifford gates, single-qubit rotations, and rotations about n-qubit Pauli operators. In Chapter 3 we give an overview of randomized benchmarking, including a discussion of Pauli twirling, Clifford twirling, and unitary twirling. We give a general statistical description of fully randomized benchmarking, and provide notation and conventions for the basic model of of a single exponential decay. In Chapter 4 we describe gate decompositions for single-qubit and two-qubit randomized benchmarking, using a native gate set that is relevant for ion traps. In Chapter 5 we give several statistical models of fully randomized benchmarking that include relevant time-dependent or position-dependent errors. In Chapter 6 we describe the numerical procedure that we use to optimize the design of an experiment to minimize the uncertainty in the step error according to a pre-chosen statistical model and best guess for the step error and other parameters. In Chapter 7 we describe the statistical analysis we use for randomized benchmarking. We also use previously published randomized benchmarking experiments to give evidence that noticeable improvements in uncertainty are possible from fully randomizing. In Chapter 8 we describe our analysis of a demonstration of fully randomized benchmarking performed by our experimental collaborators at NIST. In Chapter 9 we describe the behavior of Clifford randomized benchmarking with gate-independent non-Markovian errors and put a concrete bound on the amount of revivals in the set of success probabilities.

Chapter 2

Preliminaries

We assume that the reader is familiar with quantum information theory at the level of the standard textbook Ref. [55]. Parts of this thesis are formulated for a general D-level quantum system, modeled by a D-dimensional Hilbert space. However, other parts focus on quantum systems consisting of n qubits. It is convenient to formulate the mathematical description of these systems in terms of the Pauli observables for qubits, and the generalized Pauli observables for qudits. On a single qubit system, the Pauli observables are $\{I, X, Y, Z\}$. The Pauli observables $\{X, Y, Z\}$ are sometimes referred to by an indexed list $\{P_1, P_2, P_3\}$ or equivalently as $\{P_i\}_{i=1}^3$. In this notation, the algebraic relations of the Pauli observables can be conveniently expressed in terms of the commutation relations and the anti-commutation relations. The commutation relations are as follows

$$[P_i, P_j] = 2i\epsilon_{ijk}P_k, \tag{2.1}$$

where ϵ_{ijk} is the fully anti-symmetric symbol that evaluates to 1 for an even permutation of $(1, 2, 3)$, evaluates to -1 for an odd permutation, and evaluates to 0 if any index is repeated. The anti-commutation relations are as follows

$$\{P_i, P_j\} = 2\delta_{ij}I, \tag{2.2}$$

where δ_{ij} is the Dirac delta. The Pauli observables are typically represented as the standard Pauli matrices on the one-qubit Hilbert space $\mathbb{C}^2$.

On a system of n qubits, the corresponding Pauli observables can be obtained by taking all possible tensor products of the single-qubit Pauli observables. One way to express this is with the notation

$$\{I, X, Y, Z\}^{\otimes n}. \tag{2.3}$$

On a system of n qubits there are 4^n Pauli observables, each consisting of a tensor product of n single-qubit Pauli observables taken from the list above. One common way to represent an n-qubit Pauli observable is in terms of Pauli strings, which consist of a string of length n where each character is an element of the set $\{\mathrm{I}, \mathrm{X}, \mathrm{Y}, \mathrm{Z}\}$.

For a D-level quantum system, called a qudit, there is an appropriate generalization of the Pauli operators. In this case, the qudit Pauli operators are generated by $\{Z, X\}$, which satisfy the algebraic relations

$$Z^D = I \tag{2.4}$$

$$X^D = I \tag{2.5}$$

$$(X^k)^\dagger = X^{D-k} \tag{2.6}$$

$$(Z^k)^\dagger = Z^{D-k} \tag{2.7}$$

$$ZX = \omega XZ, \tag{2.8}$$

where k is an arbitary integer satisfying $0 \leq k \leq D$ and ω is a Dth root of unity. For futher information about qudit Pauli operators we refer to Refs. [33, 60]. An element of the set of qudit Pauli operators can be expressed as $\exp[i\phi] X^r Z^s$ for some angle ϕ and some integers $0 \leq r, s \leq D$. When expressed in this form, the algebraic relationship between two elements of the qudit Pauli group can be summarized as follows,

$$X^r Z^s \cdot X^t Z^u = \omega^{st-ru} X^t Z^u \cdot X^r Z^s. \tag{2.9}$$

Here we have used the notation and conventions from Ref. [33]. The standard matrix representation of the qudit Pauli operators on the Hilbert space $\mathbb{C}^D$ can be found in Refs. [33, 60], although we do not use it here. An important fact about the qudit Pauli operators is that they span the linear operators $L(H)$. Furthermore, the identity operator I is the only Pauli operator that commutes with all other Pauli operators, which can be checked with the algebraic relations in Eq. 2.9.

2.1 Quantum channels

The mathematics of quantum theory is based heavily on the theory of operators acting on a Hilbert space H. In general, an operator A is a linear map on Hilbert space. At times in this section we will want to emphasize the domain and codomain of linear maps, in which case we would denote an operator A as $H \xrightarrow{A} H$. The state of a quantum system is described by a density operator ρ, which is a positive (Hermitian) operator with trace 1 acting on H. It is conventional to refer to the space of operators on H with the notation $L(H)$, where L stands for linear maps on H. It is convenient in quantum information theory to introduce the concept of superoperators, which are linear maps on the space of operators on a Hilbert space H. If Γ is a superoperator, and we wish to emphasize its domain and codomain, we will denote it by $L(H) \xrightarrow{\Gamma} L(H)$. An often-used fact in linear algebra is that the space of superoperators can be identified with the space $L(H) \otimes L(H)$. The isomorphism that underlies this identification is as follows

$$\left(A \xrightarrow{\Gamma} K_1 A K_2\right) \leftrightarrow K_1 \otimes K_2, \tag{2.10}$$

so the superoperator Γ that maps a placeholder operator $A \in L(H)$ to $K_1 A K_2 \in L(H)$ is identified with $K_1 \otimes K_2 \in L(H) \otimes L(H)$. Implicitly here we have used the standard fact from linear algebra that maps of the form $A \to K_1 A K_2$ for arbitrary K_1, K_2 span the space of superoperators. The general evolution of a density operator ρ can be described by a quantum channel Φ, which is a completely-positive, trace-preserving (CPTP) map on the space of states. A CPTP map is often called a quantum channel, and is a special case of a superoperator. An important fact in quantum

information is that any completely-positive (CP) map Φ can be expressed in Kraus form, meaning that there are operators K_i indexed by i such that the superoperator action on an operator ρ is the following

$$\rho \xrightarrow{\Phi} \sum_i K_i \rho K_i^\dagger. \tag{2.11}$$

If Φ is trace-preserving then the corresponding constraint on the operators K_i is that $\sum_i K_i^\dagger K_i = \mathbb{1}$. In quantum information, CPTP maps are typically used to model error processes for physically-realized quantum operations. One important special case of a CPTP map is a unitary gate, which acts as a superoperator via the conjugate action. In what follows we will often consider the group of Pauli gates, treated as superoperators, which are the CPTP maps that correspond to the Pauli unitaries via the conjugate action. We denote the group of Pauli gates (as superoperators) by $\mathcal{P}$, and note that $\mathcal{P}$ is equivalent to the group of Pauli operators modulo phase. If $p \in \mathcal{P}$ is a Pauli gate and P is the corresponding Pauli operator (without phase), then the superoperator action of p is $\rho \xrightarrow{p} P\rho P^\dagger$.

2.2 Pauli channels

One important family of error channels in quantum information theory is the set of Pauli channels, which consists of mixtures of Pauli gates. If the index i runs over Pauli operators and λ_i are positive reals that sum to 1, a general Pauli channel Φ is of the form

$$\rho \xrightarrow{\Phi} \sum_i \lambda_i P_i \rho P_i^\dagger. \tag{2.12}$$

The parameters $\{\lambda_i\}$ form a probability distribution that governs the mixture over Pauli gates. Pauli error channels are widely used to model errors in quantum circuits and to model the performance of error correcting codes [2, 73].

2.3 Depolarizing channels

Depolarizing channels are an important family of error channels and will be used frequently in this thesis. Here we provide the definition of depolarizing channels on arbitrary dimension systems and explain the relationship to mixed-Pauli channels. We use the standard definition that a depolarizing channel Φ on a Hilbert space of dimension D with a depolarizing parameter λ maps an input state ρ to $\Phi(\rho) = (1-\lambda)\rho + \lambda I/D$ where I is the identity operator on Hilbert space and λ satisfies $0 \leq \lambda \leq 1 + 1/(D^2-1)$. A standard fact in quantum information is that a depolarizing channel on a D-dimensional system can be realized as a mixture of Pauli gates. One way to see this is to use representation theory, which is briefly introduced in the following paragraph.

A group G is a collection of elements, including an identity element 1, such that each element $g \in G$ has an inverse element g^{-1}. A representation of G on a vector space H is defined by the vector space H and a map R from $G \to L(H)$ that respects the group structure, meaning

$$R(g_1)R(g_2) = R(g_1 g_2), \tag{2.13}$$

for arbitrary elements $g_1, g_2 \in G$. In particular, $R(g)$ for any $g \in G$ must be an invertible operator because each group element g has an inverse. Here we use a basic fact from representation theory, called the first projection formula in Ref. [30] (Prop. 2.8 on page 15). First, let R be a representation on a vector space H and let H^G denote the subspace of invariant vectors $\{v \in H \mid R(g)v = v \; \forall \; g \in G\}$. In words H^G is the subspace of all vectors in H that are invariant under $R(G)$, meaning invariant under $R(g)$ for all $g \in G$. Then, if P_I denotes the projector onto the subspace H^G, then the first projection formula is as follows

$$\frac{1}{|G|} \sum_{g \in G} R(g) = P_I. \tag{2.14}$$

To use this formula to understand the depolarizing channel, we consider the abstract group of Pauli gates $\mathcal{P}$, which is equivalent to the group of Pauli operators with phases removed, and the

representation R that maps each $p \in \mathcal{P}$ to the corresponding superoperator, treated as an operator on $L(H)$. Concretely, if we fix a particular $p \in \mathcal{P}$ and use P to denote the corresponding Pauli operator (unitary) on H, we have

$$\mathcal{P} \ni p \xrightarrow{R} \left(A \xrightarrow{R(p)} PAP^\dagger \right). \tag{2.15}$$

The fact that R is a representation can be checked as follows

$$R(p_1 p_2) = \left(A \to P_1 P_2 A P_2^\dagger P_1 \right) = R(p_1) \circ R(p_2). \tag{2.16}$$

To determine the invariant subspace $H^{\mathcal{P}}$ of this representation, first observe that every Pauli operator P is an eigenvector of the superoperator $R(p)$ for every Pauli gate p. This follows from the fact that

$$P P_1 P^\dagger = \exp\left[i\phi\right] P_1, \tag{2.17}$$

for all Pauli operators P, P_1, and for some phase ϕ that depends on P, P_1. Therefore, the invariant subspace $H^{\mathcal{P}}$ will be determined by the set of Pauli operators P_1 that commute with all Pauli operators $P \in \mathcal{P}$. A basic algebraic fact of the Pauli operators is that the identity I is the only operator that commutes with all $P \in \mathcal{P}$. As a result, the invariant subspace $H^{\mathcal{P}}$ consists of just the identity I. The connection to the depolarizing channel is as follows. Consider the channel Φ_d that applies a Pauli gate uniformly at random. The action of this channel on an input state ρ is

$$\rho \xrightarrow{\Phi_d} \frac{1}{|\mathcal{P}|} \sum_{i=0}^{|\mathcal{P}|} P_i \rho P_i^\dagger, \tag{2.18}$$

where P_i is the Pauli operator corresponding to the Pauli gate $p_i \in \mathcal{P}$. Up to a constant factor, the action of this channel is equal to the righthand side of the first projection formula in Eq. 2.14. The action is therefore the projection onto the invariant subspace, which is the projection onto the

identity operator I. After keeping track of the constant factors, we have

$$\rho \xrightarrow{\Phi_d} \frac{\mathrm{tr}\,[\rho]}{D} I. \tag{2.19}$$

If ρ is a state it is trace 1, so we find that the channel Φ_d is a depolarizing channel with depolarizing parameter 1.

Another important channel is one where every non-identity Pauli gate is applied uniformly at random. We denote this channel by Φ_e and refer to it as the maximal-parameter depolarizing channel. The action of Φ_e on an input state ρ is as follows

$$\rho \frac{1}{|\mathcal{P}|} \xrightarrow{\Phi_e} \sum_{i=1}^{|\mathcal{P}|} P_i \rho P_i^\dagger. \tag{2.20}$$

This can be equivalently expressed as

$$\rho \xrightarrow{\Phi_e} \frac{1}{|\mathcal{P}|-1}(|\mathcal{P}|\Phi_d(\rho) - \rho). \tag{2.21}$$

Substituting Eq. 2.19 leads to

$$\rho \xrightarrow{\Phi_e} \frac{1}{|\mathcal{P}|-1}\left(\frac{|\mathcal{P}|}{D} I - \rho\right). \tag{2.22}$$

Since $|\mathcal{P}| = D^2$ we have

$$\rho \xrightarrow{\Phi_e} \frac{D}{D^2-1} I - \frac{1}{D^2-1}\rho. \tag{2.23}$$

Comparison to the definition of depolarizing channels given earlier in the section shows that Φ_e is a depolarizing channel with depolarizing parameter $\lambda = 1 + \frac{1}{D^2-1}$. This is the largest possible value of λ, which justifies the choice to call Φ_e the maximal-parameter depolarizing channel.

2.4 Clifford gates

The concept of Clifford gates plays an important role in the theory of randomized benchmarking and is used heavily in this thesis. A unitary C on a system of n qubits is a Clifford gate if it maps

Pauli operators to Pauli operators under the conjugate action

$$CPC^\dagger = \pm P', \tag{2.24}$$

where P, P' are n-qubit Pauli operators. The Clifford gates on a single qubit can be generated by $\pi/2$ rotations about the X and Y axes and are naturally divided into families based on whether the number of required $\pi/2$ rotations is even or odd. Up to sign (which corresponds to a Pauli gate), there are 6 possibilities: the $\pi/2$ rotations about the three major axes $\{X, Y, Z\}$, the identity gate, the Clifford gate R that cyclically permutes the Paulis $\{X, Y, Z\}$, and $R^\dagger$. The gate R is of particular interest in the section about pulse decompositions for randomized benchmarking. Using the notation that $R(P) = RPR^\dagger$, where $R(P)$ denotes the abstract action of the R gate on a Pauli P and $RPR^\dagger$ refers to the product of the corresponding operators, we have

$$R(X) = Y \tag{2.25}$$

$$R(Y) = Z \tag{2.26}$$

$$R(Z) = X. \tag{2.27}$$

The R gate can be realized by concatenating an $X_{\pi/2}$ pulse with a $Y_{\pi/2}$ pulse, and this fact can be verified by direct computation.

2.5 Single qubit rotations

Quantum gates are expressed mathematically as unitaries acting on the Hilbert space for the quantum system of interest. Here we focus on single qubit unitaries, and provide a convenient form for them in the Pauli operator basis. Let Q be a normalized linear combination of Paulis of the form $\sum_i p_i P_i$ where P_i are Pauli operators and p_i are real coefficients satisfying $\sum_i p_i^2 = 1$. Any single-qubit gate can be described by a unitary U of the form $\exp[-iQ\phi/2]$ for some normalized linear combination of Paulis Q and some real angle ϕ. We now show that such a U can be expressed

as

$$U = \cos(\phi/2)I - i\sin(\phi/2)Q \tag{2.28}$$

To check this, observe that $U = \exp[-iQ\phi/2]$ can be written

$$U = \sum_k \frac{(-iQ\phi/2)^k}{k!}. \tag{2.29}$$

Any normalized Pauli vectors satisfies $Q^2 = I$, so this series decomposes into odd and even terms as follows

$$U = I \sum_{k,\text{even}} \frac{(-i\phi/2)^k}{k!} + Q \sum_{k,\text{odd}} \frac{(-i\phi/2)^k}{k!}. \tag{2.30}$$

We can recognize the series expansions of sin and cos to get

$$U = I\cos(\phi/2) - iQ\sin(\phi/2). \tag{2.31}$$

To confirm our intuition that this unitary corresponds to a rotation of the Bloch sphere [55] by an angle ϕ around the axis set by Q, observe that the conjugate action of U on another Pauli vector S is

$$USU^\dagger = (I\cos(\phi/2) - iQ\sin(\phi/2))S(I\cos(\phi/2) + iQ\sin(\phi/2)) \tag{2.32}$$

$$= S\cos^2(\phi/2) + QSQ\sin^2(\phi/2) + i(SQ - QS)\sin(\phi/2)\cos(\phi/2). \tag{2.33}$$

If Q and S commute we have $USU = S$, and if Q and S anti-commute we have $USU = Q\cos\phi - iQS\sin\phi$.

As a concrete example we consider $\pi/2$ and π rotations about major axes on the Bloch sphere. To represent an $X_{\pi/2}$ pulse denoted $U_{X_{\pi/2}}$, set $P = X$ and set $\phi = \pi/2$. Substitution into Eq. 2.31 shows that

$$U_{X_{\pi/2}} = I\cos(\pi/4) - iX\sin(\pi/4). \tag{2.34}$$

It is convenient to compute the conjugate action on the other Pauli operators. The conjugate action on X is trivial, and the conjugate action on Y is as follows

$$U_{X_{\pi/2}} Y U_{X_{\pi/2}}^{\dagger} = (I\cos(\pi/4) - iX\sin(\pi/4))Y(I\cos(\pi/4) + iX\sin(\pi/4)) = -i\sin(\pi/2)(iZ) = Z. \tag{2.35}$$

We see directly that Y is mapped to Z under the action of this unitary, which confirms the intution that this is a $\pi/2$ rotation about the X-axis of the Bloch sphere. Computing for the conjugate action on Z shows that Z is mapped to $-Y$. The $\pi/2$ pulses about other major axes function similarly, and the conjugate action on Paulis can also be computed in this way.

2.6 Clifford rotations on n-qubit systems

The previous treatment applies for rotations on a single-qubit system, but certain unitaries on an n-qubit system can be treated similarly. As one class of convenient unitaries, we consider rotations about a single n-qubit Pauli. More generally, we show that a rotation by $\pi/2$ about any n-qubit Pauli P is a Clifford gate. To see this, use Eq. 2.31 to show that a $\pi/2$ rotation about P can be expressed as

$$U_{P_{\pi/2}} = \cos(\pi/4)I - i\sin(\pi/4)P. \tag{2.36}$$

Now, using Eq. 2.32 we can see that $UQU^{\dagger} = Q$ if P and Q commute, and if they anticommute we have $UQU^{\dagger} = iQS$, where we have dropped the first term because $\cos(\pi/2) = 0$. Because Q and S anticommute, we have $QS = \pm i(QS)$, where (QS) is the Pauli (without sign) corresponding to the Pauli product of Q and S. As a result, the factors of i cancel and we see that the result of the conjugate action is $\pm(QS)$, where the sign is determined by the particular Q and S. Regardless of sign, the output is a Pauli, so we can conclude that a $\pi/2$ rotation about any P is a Clifford gate. In fact, a simpler way to show this is to use the fact that there exists some Clifford C that maps $P \to XIII\ldots$, where $XIII\ldots$ denotes the Pauli that is X on the first qubit and I on the remaining qubits. Then, applying conjugation by C and using the fact that a $\pi/2$ rotation on

a single qubit is a Clifford proves the result immediately. An example of such a unitary that is addressed frequently in this thesis is a $\pi/2$ rotation about the Pauli ZZ on two qubits. This is sometimes referred to as the $\sqrt{ZZ}$ gate, since applying it twice is equivalent to the ZZ Pauli gate.

Chapter 3

Overview of randomized benchmarking

Many variants of randomized benchmarking involve applying random sequences of gates in order to infer certain properties of the associated error processes. The purpose of randomization is to simplify the form of effective error channels through 'twirling'. Twirling an error channel involves randomly selecting a unitary from a particular ensemble, prepending the gate to the error channel, and appending the inverse gate to the error channel. Different choices of gate ensembles lead to different randomization methods in RB-style experiments. Two of the most common randomization methods are Pauli randomization and Clifford randomization. Standard randomized benchmarking typically involves Clifford randomization, which has the mathematical effect of converting all error channels to depolarizing channels (assuming gate-independent errors). In fact, many ensembles of gates can have this effect, including the uniform ensemble over all unitary gates. Any such ensemble is called a 'two-design' and we provide further discussion in Section 3.3. In contrast, Pauli randomization has the effect of converting all error channels into Pauli channels (again assuming gate-independent errors). Pauli randomization is therefore a key theoretical component of characterization methods that assume Pauli error channels, for example Ref. [29]. Another common use case of Pauli randomization is cycle benchmarking [23], which is a variant of randomized benchmarking that uses Pauli randomization to infer the fidelity of a Clifford gate. Furthermore, characterization techniques for error correcting codes typically assume error channels are either depolarizing or Pauli channels.

3.1 Pauli twirling

Pauli twirling is a transformation of a quantum channel Φ that is achieved by selecting a Pauli gate uniformly at random, and then applying that gate before the channel Φ and applying the inverse gate afterward. The effect of Pauli twirling is to transform Φ into a Pauli channel. Mathematically, Pauli twirling corresponds to the mapping of superoperators $\Phi \to \widetilde{\Phi}$, where $\widetilde{\Phi} = \frac{1}{|\mathcal{P}|}\sum_k P_k \circ \Phi \circ P_k$ is the twirled error process. In the previous line, $|\mathcal{P}|$ is the size of the Pauli group, P_k refers to the Pauli gate (as a superoperator) corresponding to the kth Pauli operator, the symbol $\circ$ denotes the composition of channels, and the sum denotes a classical mixture of channels. In terms of the Kraus operators $\{K_i\}$ of Φ, $\widetilde{\Phi}$ can be expressed as

$$\widetilde{\Phi}(\rho) = \sum_k \sum_i P_k K_i P_k \rho P_k K_i^\dagger P_k. \tag{3.1}$$

The twirled process $\widetilde{\Phi}$ can be expressed in terms of effective Kraus operators $\widetilde{K_i}$ by performing the sum over k for each fixed i. Each Kraus operator K can be expressed in terms of the Pauli operators as $K = \sum_j c_j P_j$. In total, the terms in the Kraus decomposition can be collected to give the following form, which is entirely in terms of Pauli operators.

$$\Phi(\rho) = \sum_{ii'} C_{ii'} P_i \rho P_{i'} \tag{3.2}$$

Using the algebraic properties of Pauli operators [33, 71], the twirling can be interpreted as a mapping that sends the coefficients $C_{ii'}$ in Eq. 3.2 to zero unless $i = i'$. Therefore, the twirled process $\widetilde{\Phi}$ can be expressed as

$$\widetilde{\Phi}(\rho) = \sum_i C_{ii} P_i \rho P_i, \tag{3.3}$$

where C_{ii} are positive coefficients and the index i runs over all Pauli operators. This has the form of a Pauli channel, which is a mixture of Pauli gates with mixture coefficients given by C_{ii}. This fact is the basis for the extensive work about Pauli error channels, randomized compiling, and

cycle benchmarking. A proof appears in Ref. [71], and for convenience we provide a self-contained treatment here using representation theory.

The overall idea is that the twirling action as a map $\Phi \to \widetilde{\Phi}$ on the space of superoperators can be interpreted as the action of a representation averaged over the group $\mathcal{P}$, as in Eq. 2.14. Using the fact that the invariant subspace of the representation operators is spanned by the Pauli gates (as superoperators), applying Eq. 2.14 shows that the resulting channel after twirling must be a Pauli channel. The details are as follows. The group G is the abstract group of Pauli gates $\mathcal{P}$ (as superoperators), and the representation R in question is a map from $\mathcal{P}$ to the space of linear maps on superoperators,

$$\mathcal{P} \ni p \xrightarrow{R} \left(\Phi \to P \circ \Phi \circ P^{-1}\right). \tag{3.4}$$

In the previous equation, $p \in \mathcal{P}$ is a Pauli gate and P is the corresponding superoperator, which maps $\rho \to P\rho P^{\dagger}$. On the righthand-side the map $\left(\Phi \to P \circ \Phi \circ P^{-1}\right)$ is a linear map on the space of superoperators that sends the superoperator Φ to the superoperator $P \circ \Phi \circ P^{-1}$. The twirling action $\Phi \to \sum_k P_k \circ \Phi \circ P_k$ is equal to the action of the average over the representation R, as in Eq. 2.14. To apply the projection formula in Eq. 2.14, it is necessary to first determine the invariant subspace under the representation R. To do this, first observe that the set of superoperators $\{\Phi_{i,j}\}_{i,j}$ defined by

$$A \xrightarrow{\Phi_{i,j}} P_i A P_j^{\dagger}, \tag{3.5}$$

spans the space of superoperators. (In the previous equation the indices i, j each run independently over the set of Pauli operators.) This fact follows from the isomorphism between the space of superoperators and the space $L(H) \otimes L(H)$, and from the fact that that the Pauli operators span $L(H)$. For an arbitrary Pauli gate $p \in \mathcal{P}$, each $\Phi_{i,j}$ is an 'eigenvector' of $R(p)$,

$$\Phi_{i,j} \xrightarrow{R(p)} \exp\left[i\phi\right] \Phi_{i,j}. \tag{3.6}$$

In the previous equation, the phase factor $\exp\left[i\phi\right]$ is determined by the Pauli commutation relations

of the Pauli P corresponding to $p \in \mathcal{P}$ and the P_i, P_j that correspond to $\Phi_{i,j}$ according to Eq. 3.5. An important point is that if $P_i = P_j$ then the phase $\phi = 0$ for any P. If $P_i \neq P_j$ then there will necessarily exist some $p \in \mathcal{P}$ such that $\phi \neq 0$. To see this, consider $P_i = Z(1)$, which is without loss of generality because any Pauli can be transformed to $Z(1)$ by a Clifford unitary. Then, assuming P_j is written in general form as $P_j = Z(z)X(x)$, if $x = 0$ set $P = X(1)$, and if $x \neq 0$ set $P = Z(1)$. Therefore, the superoperators $\Phi_{i,j}$ for $i \neq j$ are not in the invariant subspace of $R(p)$. Thus, the invariant subspace of $R(p)$ is spanned by the elements $\Phi_{i,j}$ for $i = j$. The projection formula in Eq. 2.14 therefore implies that after Pauli twirling, the resulting superoperator $\widetilde{\Phi}$ is a linear combination of the superoperators $\{\Phi_{i,i}\}$. Because the original channel Φ is a CPTP map, then the coefficients in the linear combination must be positive and sum to 1, meaning that the linear combination must in fact be a classical mixture of $\{\Phi_{i,i}\}$. This completes the proof that applying Pauli twirling an arbitrary channel Φ results in an output channel $\widetilde{\Phi}$ that is a Pauli channel.

3.2 Unitary twirling

In addition to Pauli twirling, another twirling possibility is unitary twirling, which involves randomizing over all unitaries uniformly according to the Haar measure. The concept of unitary twirling has been used substantially in the randomized benchmarking literature, and has the effect of converting any channel into a depolarizing channel [51].

Here we provide a treatment of unitary twirling using Schur-Weyl duality, in a way that applies equally to qubit and qudit systems. We first introduce Schur-Weyl duality in a convenient form following Ref. [17]. According to Theorem 3.5 in Ref. [17], if $M \in L(H \otimes H)$, then the following equivalence holds

$$\int dU (U \otimes U) M \left(U^\dagger \otimes U^\dagger\right) = \alpha \mathbb{1} + \beta F, \tag{3.7}$$

where dU is the Haar measure over unitary operators on H, F is the swap operator on $H \otimes H$ and α, β are constants.

To apply Schur-Weyl duality to the twirling of superoperators, we use the isomorphism

between superoperators and elements of $L(H) \otimes L(H)$ introduced in Section 2.1. In particular, we identify the superoperator $\rho \to K_1 \rho K_2$ with the element $K_1 \otimes K_2 \in L(H) \otimes L(H)$. If K_1 and K_2 are each basis elements of $L(H)$, then $K_1 \otimes K_2$ represents a basis element of the space of superoperators. After unitary twirling of a superoperator corresponding to $K_1 \otimes K_2$, the resulting superoperator has the action

$$\rho \to \int dU U K_1 U^\dagger \rho U K_2 U^\dagger. \tag{3.8}$$

Unitary twirling can therefore be expressed as a transformation on $L(H) \otimes L(H)$ as follows

$$K_1 \otimes K_2 \to \int dU U K_1 U^\dagger \otimes U K_2 U^\dagger. \tag{3.9}$$

To match with Theorem 3.5 in Ref. [17], as shown in Eq. 3.7, we express this as

$$K_1 \otimes K_2 \to \int dU (U \otimes U) K_1 \otimes K_2 \left(U^\dagger \otimes U^\dagger\right). \tag{3.10}$$

The theorem can be directly applied to Eq. 3.10, showing that the superoperator after twirling, when expressed as an element of $L(H \otimes H)$, is a linear combination of the identity operator and the the swap operator. To understand the interpretation of the swap operator, we first write it in coordinates as

$$\text{swap} = \sum_{ij} |i\rangle\langle j| \otimes |j\rangle\langle i| . \tag{3.11}$$

Re-identifying this element of $L(H \otimes H)$ with a superoperator, we have the following action on an input operator A

$$A \to \sum_{ij} |i\rangle\langle j| A |j\rangle\langle i| . \tag{3.12}$$

Evaluating the sum shows that this is equivalent to

$$A \to \text{tr}[A] I. \tag{3.13}$$

The resulting superoperator is therefore the completely depolarizing channel. Therefore, a convex linear combination of the identity operator and the swap operator (as an element of $L(H \otimes H)$) is identified with a depolarizing channel as a superoperator. We conclude that any channel becomes a depolarizing channel after unitary twirling.

3.3 Two-designs and Clifford twirling

Unitary twirling is often not experimentally convenient due to the difficulty in compiling a Haar random unitary into native gates. One experimental solution is to instead twirl according to a two-design, which is an ensemble of gates that also converts any channel into a depolarizing channel after twirling [14, 51, 71]. For the purposes of randomized benchmarking, twirling by a two-design is sufficient. In practice, the most common two-design is the uniform ensemble over all Clifford gates [51, 71]. However, there exist two-designs consisting of other ensembles of Clifford gates. For example, the two-qubit two-design that we consider in Chapter 4 does not sample uniformly over all two-qubit Cliffords, and instead consists of Clifford gates drawn from a carefully-constructed ensemble. To prove that the ensemble considered in Chapter 4 is a two-design, we use the fact from Ref. [71] that Pauli randomization followed by Pauli mixing forms a two-design. An ensemble of Clifford gates is Pauli mixing if each non-identity Pauli operator is mapped to the uniform distribution over non-identity Pauli operators under the conjugate action of the Clifford ensemble.

One way to understand the fact that Pauli randomization followed by Pauli mixing is as follows. The effect of Pauli mixing on a channel of the form in Eq. 3.3 is to set all the coefficients C_{ii} for $i \geq 1$ to their average $\bar{C}$. The resulting channel is a depolarizing channel with depolarizing parameter determined by $\bar{C}$. Furthermore, a well-known fact about the Clifford group (on qubits or qudits) is that the uniform ensemble over all Clifford gates forms a two-design [71]. Twirling with a two-design is a key part of any randomized benchmarking protocol [51]. The most common implementation of a two-design is to simply sample uniformly over all Cliffords, but other implementations are useful.

3.4 Overview of fully randomized benchmarking

This section is substantially based on work published in Ref. [45]. We describe our conventions and notation for a statistical description of fully randomized benchmarking experiments, with a focus on the basic model that results in a single exponential decay of the success probability with sequence length. For further information and discussion of randomized benchmarking in general, we refer to Refs. [39, 44, 49, 51]. A fully randomized benchmarking experiment consists of many independent trials, where a trial of sequence length n is composed of a state preparation, followed by a random sequence of n steps, followed by measurement indicating 'success' or 'failure'. The content and distribution of the random steps depends on the benchmarking variant in use. For example, in standard randomized benchmarking each step nominally implements a random Clifford gate. The design of a fully randomized experiment consists of the list $(n_j)_{j=1}^{j_{\max}}$ of sequence lengths to be used, and the list $(w_j)_{j=1}^{j_{\max}}$ of the numbers of independently-randomized trials to be performed at each sequence length. In a fully randomized experiment, the order in which the total $\sum_j w_j$ trials are performed should also be randomized. As we discuss at the end of this section, this can help to minimize the effect of potential time-dependent errors. After the experiment is performed, the data consists of a list $(c_j)_{j=1}^{j_{\max}}$ where c_j is number of success counts observed out of w_j total trials at the sequence length n_j. In general, a statistical model for fully randomized benchmarking consists of a list of parameters (θ_i), which we refer to as $\boldsymbol{\theta}$, and a function $P_{\boldsymbol{\theta}}(n)$ that determines the success probabilities at each sequence length n in terms of the parameters. The success counts (c_j) are then binomially distributed for each j with success probability $P_{\boldsymbol{\theta}}(n_j)$. We note that the procedures for experiment design and analysis that we describe in Sections 6.1 and 7.1 hold for a general model $P_{\boldsymbol{\theta}}(n)$. We assume that fully randomized benchmarking on any particular experimental system admits an accurate description in terms of some $P_{\boldsymbol{\theta}}(n)$ and corresponding statistical parameters $\boldsymbol{\theta}$.

The most common statistical model for randomized benchmarking is a single exponential decay with a rate that represents the step error and a proportionality constant that represents the state preparation and measurement error. We refer to this model as the basic model and provide a

concrete definition in Eq. 3.16. The basic model and other models that we consider in Section 5 can be justified under certain assumptions about the behavior of the experimental system in question. For simplicity and completeness, we provide one such set of assumptions.

(i) Each trial consists of state preparation of a nominal computational basis state $|\psi_{\text{in}}\rangle\langle\psi_{\text{in}}|$ followed by a random sequence of steps from a two-design [1, 14], followed by a randomized final step that returns the state to a random computational basis state, followed by measurement in the computational basis. For discussion of randomized final steps and measurements see Ref. [51]. In short, randomizing the final step and measurement allows the combined effect of state preparation and measurement errors to be treated as a single depolarizing error channel.

(ii) In every trial the kth step has an error channel Λ_k that does not depend on the gate that the step nominally implements. The assumption that errors can be modeled by a channel Λ_k is the Markovian assumption, according to the definition in Ref. [58]. The assumption of gate-independent errors is discussed in further detail in Ref. [69].

(iii) The system is completely reset after each trial and no memory effects are present between trials. This assumption disallows, for example, the possibility of step errors that depend on the temperature of a trapped ion motional mode [66] that heats over time and is not reset at the beginning of each shot. However, this assumption still allows for the possibility that step errors can be drawn from a distribution independently for each trial, or can increase throughout a sequence.

When these assumptions are made, the error channels Λ_k are 'twirled' by the random gates from a two-design and become effective depolarizing channels [51]. In general, the parameters of these depolarizing channels can randomly fluctuate trial-to-trial or can depend on the gate index k. The success probability of a sequence of length n is determined by the composition of all the depolarizing channels at gate indices less than n, which can lead to more complicated behavior than a single exponential decay. To justify the single exponential decay in the basic model, we add a final assumption.

(iv) The error channels Λ_k are independent of time and independent of the step index k. This assumption ensures that each effective depolarizing channel has the same depolarizing parameter.

Although these assumptions may seem restrictive, a single exponential decay can still be a good model in many situations where gate-dependent or certain time-dependent errors are present [22, 69]. In the case of gate-dependent errors, the observed rate of exponential decay may differ from the average fidelity of the gates relative to a fixed basis [59]. The observed exponential decay rate is still indicative of gate performance, however [59]. In the case of errors that depend on a classically fluctuating quantity like temperature, randomizing the order of sequence lengths during the experiment leads to a success probability at each sequence length that is averaged over the fluctuating quantity. To good approximation, this behavior can lead to an effective model where the step errors randomly fluctuate trial-to-trial independently. For further information about error models and assumptions in randomized benchmarking, we refer to Refs. [22, 28, 59, 69].

We now provide notation and conventions for the basic model. We use the standard definition that a depolarizing channel Φ on a Hilbert space of dimension D with a depolarizing parameter λ maps an input state ρ to $\Phi(\rho) = (1-\lambda)\rho + \lambda I/D$ where I is the identity operator on Hilbert space and λ satisfies $0 \leq \lambda \leq 1 + 1/(D^2-1)$. If a system is initialized in a pure state $|\psi\rangle\langle\psi|$ and a depolarizing channel with parameter λ is applied, the fidelity f of the output state with the input state is

$$f = \mathrm{tr}\left[\Phi(|\psi\rangle\langle\psi|) \cdot |\psi\rangle\langle\psi|\right] = 1 - \lambda + \lambda/D. \tag{3.14}$$

The fidelity f does not depend on the input state $|\psi\rangle\langle\psi|$, and therefore the average fidelity of the depolarizing channel Φ is equal to f. We refer to $\varepsilon = 1 - f$ as the error of the depolarizing channel Φ. If depolarizing channels with parameters $\{\lambda_i\}$ are concatenated, the resulting channel is a depolarizing channel with parameter $\lambda = 1 - \prod_i (1-\lambda_i)$. When the fidelity of the concatenated channel is expressed in terms of the individual errors it simplifies to the following

$$f = \frac{1}{D} + \frac{1}{\alpha}\prod_i (1 - \alpha\varepsilon_i), \tag{3.15}$$

where $\alpha = \frac{D}{D-1}$. This motivates the following definition of the basic model,

$$P_{\boldsymbol{\theta}}(n) = \frac{1}{D} + \frac{1}{\alpha}(1 - \alpha\theta_0)(1 - \alpha\theta_1)^n, \tag{3.16}$$

where n is the sequence length, θ_0 is the SPAM error, and θ_1 is the step error. Here n is a non-negative integer, and $\theta_0, \theta_1 \in [0, 1]$. In Section 5 we describe several other models of experimental interest that generalize the basic model. An important property of the basic model is that the SPAM parameter θ_0 appears affine linearly in the expression for $P_{\boldsymbol{\theta}}(n)$. As a result, a randomly fluctuating SPAM parameter is indistinguishable from a constant SPAM parameter equal to the mean of the distribution of random fluctuations. Fully randomized benchmarking is therefore insensitive to drifts in the SPAM parameter, as long as the drifts are uncorrelated with the sequence lengths. In contrast, in a randomized benchmarking experiment that is not fully randomized, randomly fluctuating SPAM errors can contribute to additional statistical fluctuations in the inferred step error.

Chapter 4

Pulse decompositions for RB

Here we discuss convenient gate decompositions for two-designs for one and two qubit randomized benchmarking experiments. One goal of these particular gate decompositions is to ensure that the gates that are the dominant sources of error appear in a fixed pattern. One benefit of designing gate sequences in a fixed pattern is that for a fixed sequence length (meaning a fixed number of steps from a two-design) each of the random sequences have the same number of 'hard' gates, and therefore have approximately the same errors under a depolarizing error model dominated by the hard gates. If instead the number of hard gates can fluctuate over the random sequences, then even with a depolarizing error model the success probabilities at a given sequence length will also fluctuate over random sequences. As we describe in Chapter 7, if the experiment is not fully randomized then this fluctuation will lead to an excess variance in the inferred step error. An important point is that requiring a fixed pattern of hard gates can in general lead to a larger error per step because gate decompositions will involve more hard gates than are strictly necessary to implement each step. However, if the goal is to perform interleaved randomized benchmarking, then using a fixed pattern of hard gates can lead to improved signal-to-noise due to the uniformity of the step errors, as long as the step errors aren't large enough to dominate the error of the interleaved gate.

For single-qubit systems, we focus on gate decompositions using native gates consisting of $\pi/2$ rotations about the $\pm X$ or $\pm Y$ axes of the Bloch sphere. This is motivated by the fact that in many experimental realizations of qubits the available operations consist of electromagnetically-driven

rotations about equatorial axes of the Bloch sphere in an appropriate rotating frame. Often, the equatorial axis of the rotation is determined by the phase of the electromagnetic drive relative to the rotating frame of the qubit. In many experimental scenarios this phase can be electronically controlled to extremely good accuracy, so that any phase error is negligible. For the purposes of this chapter, we restrict to the $\pm X$ or $\pm Y$ axes so that the $\pi/2$ rotations correspond to Clifford gates. Furthermore, we assume that any physical measurement made at the end of the circuit is in the Z-basis, and that any measurement in a different Pauli basis is implemented by first performing a Clifford gate and then making a physical measurement in the Z-basis. In this situation, we can safely assume that a $Z_{\pi/2}$ rotation can be performed perfectly at any point, simply by modifying the phases of future pulses. This takes advantage of the fact that any potential Z-axis rotation at the end of the circuit can be absorbed into a Z-basis measurement.

For two-qubit systems, we treat the case where the native gateset consists of collective equatorial rotations, a differential Z gate that allows for independent control of the two qubits, and a Clifford entangling gate like CNOT. Concretely, the collective equatorial rotations are of the form $X_{\pi/2} \otimes X_{\pi/2}$ (or $Y_{\pi/2} \otimes Y_{\pi/2}$, etc...), and implement the same rotation about some equatorial axis on the two qubits. The differential-Z gate applies a $Z_{\pi/2}$ rotation to one qubit and a Z rotation to to other and without loss of generality can be expressed as $Z_{\pi/2} \otimes Z$. In certain trapped-ion platforms the differential-Z gate is a natural method of implementing non-collective single-qubit operations.

4.1 Pauli randomization for one qubit

Pauli randomization consists of the uniformly-weighted ensemble of Pauli gates. Therefore, to address gate decompositions for single-qubit Pauli randomization it suffices to simply implement the Pauli gates using the native gates. The Pauli gates on one qubit consist of the list $\{I, X, Y, Z\}$. The goal is to find a fixed pattern of equatorial $\pi/2$ pulses where each of the Pauli gates can be realized by an appropriate choice the axes of the $\pi/2$ pulses in the fixed pattern. We find it convenient to allow the Pauli Z gate to possibly appear at the end of the fixed pattern of $\pi/2$ pulses, where it can often be absorbed into the rotation angle of future pulses. Furthermore, if a Z-rotation appears just

prior to measurement in the Z basis, then the Z-rotation can be ignored. Allowing Z-rotations to appear at the end of a fixed pulse pattern is especially helpful in the case of a sequence of random Pauli gates. As we will show, in this scenario the Z gate at the end of each step can be ignored directly without worrying about absorbing it into the phase of future pulses. As we describe here, we find that a fixed pattern of two equatorial $\pi/2$ pulses is sufficient to decompose each of the Pauli gates $\{I, X, Y, Z\}$. A convenient decomposition into the native gates is shown in Table 4.1.

Table 4.1: Gate decomposition of single-qubit Pauli gates into equatorial $\pi/2$ pulses. Multiple equivalent possibilities are separated by commas in the second column. The entry $I \circ Z$ in the fourth row is shorthand for any entry in the first row (corresponding to an I gate) followed by a Z gate. The Z gate can generally be compiled into future gates.

Pauli gate	Pulse decompositions of Pauli gates
I	$X_{-\pi/2}X_{+\pi/2}$, $X_{+\pi/2}X_{-\pi/2}$, $Y_{+\pi/2}Y_{-\pi/2}$, $Y_{-\pi/2}Y_{+\pi/2}$
X	$X_{+\pi/2}X_{+\pi/2}$, $X_{-\pi/2}X_{-\pi/2}$
Y	$X_{+\pi/2}X_{+\pi/2}$, $X_{-\pi/2}X_{-\pi/2}$
Z	$I \circ Z$

Each of the entries in the table can be directly verified by using the fact that the composition of two $\pi/2$ pulses about the same axis is equivalent to a π pulse about that same axis. Similarly, the composition of a $\pi/2$ pulse about an axis with a π pulse about the negative of that axis is the identity. We note that the ensemble of native gates shown in each line of the table is invariant under conjugation by a Pauli Z gate. As a result, to construct a sequence of random single-qubit Pauli gates followed by a measurement in the Z-basis, it suffices to drop the Z-gate at the end of the fourth line and treat the Pauli Z gate exactly the same as an I gate. Concretely, to construct a single step of a Pauli-randomized sequence, first draw a gate from the set $\{I, X, Y, Z\}$ uniformly at random. Then, decompose the particular gate into the native gates by consulting the relevant line of the table. If the relevant line has multiple possibilities for the decomposition (separated by commas), draw one of the possibilities uniformly at random. For the case of the Pauli Z gate, use the exact same ensemble as for the identity gate. This pulse pattern can be used for Pauli

benchmarking on a single qubit, which gives a measure of the average quality of the implementation of the four Pauli gates $\{I, X, Y, Z\}$.

4.2 Single qubit two-design

Here we focus on single-qubit randomized benchmarking where the native gate are $\pi/2$ pulses around the x or y axes of the Bloch sphere. For convenience, we imagine that any z-axis rotations can be implemented perfectly by changing the axis of all later pulses. As we will show, this means that all z-axis rotatations can be eliminated from the pulse sequence. One way to understand the single-qubit two-design that we use is in terms of Pauli randomization followed by Pauli mixing, where the Pauli mixing is implemented by drawing from the Clifford gates $I, R, R^\dagger$ uniformly at random. Altogether, this two-design can be implemented by drawing uniformly at random from the following list of 12 Clifford gates

$$\{I, X, Y, Z\} \circ \{I, R, R^\dagger\} = \{I, X, Y, Z, R, XR, YR, ZR, R^\dagger, XR^\dagger, YR^\dagger, ZR^\dagger\}. \tag{4.1}$$

Each of these gates can be implemented by an appropriate selection of two $\pi/2$ pulses, where each one is a rotation about one of the following axes $\{+X, -X, +Y, -Y\}$. To do this, we first allow Z rotations to occur after the two $\pi/2$ pulses, and later we show that they can be eliminated. Including Z gates, the pulse decompositions are shown in Table 4.2. Again, the uniformly random ensemble of Clifford gates from this list is invariant to conjugation by Pauli Z, so the Pauli Z in the fourth line of the table can be eliminated in a sequence of uniformly random gates.

4.3 Two qubit Pauli randomization with differential Z

We consider pulse schemes for performing Pauli randomization where the native gates are collective $\pi/2$ pulses about the $\{+X, -X, +Y, -Y\}$ axes and the only non-collective gate is a differential-Z gate of the form $Z_{\pi/2} \otimes Z$. In what follows we refer to the differential-Z gate as the DZ gate, as shown in Fig. 4.1. We still allow 'free' collective $Z \otimes Z$ rotations because these commute

Table 4.2: Gate decomposition of relevant single-qubit Clifford gates into equatorial $\pi/2$ pulses. Multiple equivalent possibilities are separated by commas in the second column. The entry $I \circ I \circ Z$ in the fourth row is shorthand for any entry in the first row (corresponding to an I gate) followed by a Z gate.

Clifford gate	Pulse decompositions of relevant Cliffords
$I \circ I$	$X_{-\pi/2}X_{+\pi/2}$, $X_{+\pi/2}X_{-\pi/2}$, $Y_{+\pi/2}Y_{-\pi/2}$, $Y_{-\pi/2}Y_{+\pi/2}$
$X \circ I$	$X_{+\pi/2}X_{+\pi/2}$, $X_{-\pi/2}X_{-\pi/2}$
$Y \circ I$	$X_{+\pi/2}X_{+\pi/2}$, $X_{-\pi/2}X_{-\pi/2}$
$Z \circ I$	$I \circ I \circ Z$
$I \circ R$	$Y_{+\pi/2}X_{+\pi/2}$
$X \circ R$	$Y_{-\pi/2}X_{-\pi/2}$
$Y \circ R$	$Y_{-\pi/2}X_{+\pi/2}$
$Z \circ R$	$Y_{+\pi/2}X_{-\pi/2}$
$I \circ R^\dagger$	$X_{-\pi/2}Y_{-\pi/2}$
$X \circ R^\dagger$	$X_{+\pi/2}Y_{-\pi/2}$
$Y \circ R^\dagger$	$X_{+\pi/2}Y_{+\pi/2}$
$Z \circ R^\dagger$	$X_{-\pi/2}Y_{+\pi/2}$

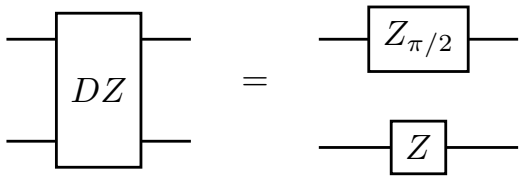


Figure 4.1: Definition of the differential-Z (DZ) gate as a $Z_{\pi/2}$ pulse on the first qubit and a Z gate on the second qubit.

with the differential Z gate and can be absorbed into the phases of future collective pulses. For the purpose of two-qubit Pauli benchmarking, we are interested in decomposing each two-qubit Pauli gate into a fixed pattern of collective $\pi/2$ pulses and differential Z gates. The two-qubit Pauli gates are of the form $\{I, X, Y, Z\} \otimes \{I, X, Y, Z\}$. First of all, we note that any such pattern must have at least two DZ gates. This follows from the fact that any single qubit Pauli gate must have an even

number of $\pi/2$ pulses. Any pattern with an odd number of DZ gates will necessarily have an even number of $\pi/2$ pulses on one qubit and an odd number of $\pi/2$ pulses on the other qubit, which makes it impossible to construct a gate of the form $P_1 \otimes P_2$ where P_1, P_2 are both Pauli gates. The minimal non-collective part of a pulse decomposition is therefore of the form $DZ \circ C \otimes C \circ DZ$, where $C \otimes C$ is a stand-in for an arbitrary collective Clifford gate. As we show, the full pulse decomposition consists of this non-collective part as well as additional collective Clifford gates. It is reasonable to first determine the minimal decomposition of $C \otimes C$ into collective $\pi/2$ pulses. We find that a non-collective part of the form $DZ \circ \pi/2 \circ DZ$ is not sufficient, and that a non-collective part of the form $DZ \circ \pi/2 \circ \pi/2 \circ DZ$ is required. In the previous sentence the element $\pi/2$ in $DZ \circ \pi/2 \circ DZ$ refers to an arbitrary collective major-equatorial-axis $\pi/2$ pulse. The non-collective part of the circuit is shown in Fig. 4.2. To see that a non-collective part of this form is necessary,

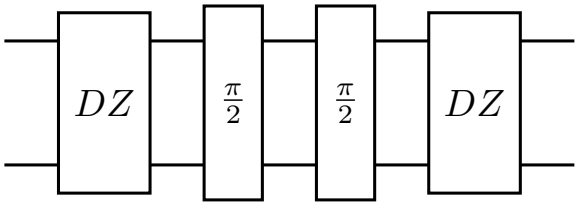


Figure 4.2: Circuit diagram showing the non-collective part of a native gate pattern for an arbitrary two-qubit Pauli gate. The circuit element labeled DZ indicates the differential-Z gate described previously, and the circuit element labeled $\frac{\pi}{2}$ indicates an arbitrary collective major-equatorial-axis $\pi/2$ pulse. The different two-qubit Pauli gates that can be achieved with this pattern are determined by the choice of the axis of the collective $\pi/2$ pulse.

we first introduce the fact that the gate $DZ \circ X_{\pi/2} \otimes X_{\pi/2} \circ DZ$ satisfies the following relations,

$$DZ \circ X_{\pi/2} \otimes X_{\pi/2} \circ DZ = Z \otimes I \circ DZ^{\dagger} \circ X_{\pi/2} \otimes X_{\pi/2} \circ DZ^{\dagger} = Z \otimes I \circ Y_{\pi/2} \otimes X_{\pi/2}^{\dagger}. \tag{4.2}$$

To obtain these identities we used the fact that

$$Z_{\pi/2}^{\dagger} \circ X_{\pi/2} \circ Z_{\pi/2} = Y_{\pi/2}, \tag{4.3}$$

which can be verified by directly checking the conjugate action on Pauli operators. Eq. 4.2 shows that a non-collective part of the form $DZ \circ \pi/2 \circ DZ$ will necessarily be equivalent to $\pi/2$ pulses on the two qubits that are out of phase by $\pi/2$ in the equatorial plane. It is not possible to combine such a gate with only collective gates to obtain a gate of the form $P_1 \otimes P_2$. We therefore conclude that a minimal non-collective pattern of the form $DZ \circ \pi/2 \circ \pi/2 \circ DZ$ is necessary. Since collective $Z_{\pi/2}$ pulses commute with the DZ gate they can always be moved out of the non-collective part of the gate sequence. Therefore, it suffices to consider the options $\{I \otimes I, R \otimes R, R^\dagger \otimes R^\dagger\}$ for the net effect of the two $\pi/2$ pulses between the DZ gates, up to a collective Pauli gate. A direct computation shows that if the net effect is $R \otimes R$ or $R^\dagger \otimes R^\dagger$, then the combined non-collective part of the gate sequence (up to an arbitrary Pauli gate on each qubit) is equivalent to different Clifford gates in the set $\{I, R, R^\dagger\}$ acting on the two qubits. This fact will be useful for constructing independent single-qubit two-designs on the two qubits, and we will return to this concept in Section 4.4. However, for the purpose of constructing two-qubit Pauli gates the only possible option is to require that the two $\pi/2$ pulses between the DZ gates have a net effect of a collective Pauli gate, which is equivalent to $I \otimes I$ up to a collective Pauli. This is true because, if the non-collective part of the gate sequence consists of different Cliffords on the two qubits (up to Pauli gate equivalence), then it is impossible to construct a gate that implements a Pauli on both qubits by adding only collective Cliffords before and after. It therefore suffices to consider a non-collective part of the gate decomposition of the form in Fig 4.3. Direct computation of all of the possibilities of the collective

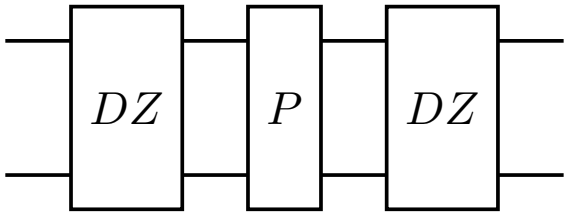


Figure 4.3: Circuit diagram indicating that, for the purposes of constructing a two-qubit Pauli gate using the DZ gate as the only non-collective gate, the two $\pi/2$ pulses in the previous figure must have the net effect of some collective Pauli gate P. Concretely, the two-qubit circuit element labeled P in the diagram is of the form $P \otimes P$ where $P \in \{I, X, Y, Z\}$.

Pauli gate achieved by the two $\pi/2$ pulses between the DZ gates shows that only the following gates can be produced by a decomposition of this form, $\{Z \otimes I, I \otimes Z, X \otimes X, Y \otimes Y\}$. Note that to achieve $I \otimes Z$ we assume that we can perform a collective Z gate for 'free'. It is unhelpful to include a third $\pi/2$ pulse between the two DZ gates, because any combination of three major-axis $\pi/2$ pulses is equivalent to a single $\pi/2$ pulse about some major axis.

It remains to determine the minimal number of collective $\pi/2$ pulses that must be added to the non-collective circuit shown in Fig. 4.2 in order to achieve a pulse decomposition for every two-qubit Pauli gate. Direct computation shows that surrounding the non-collective circuit with a single $\pi/2$ pulse before and a single $\pi/2$ pulse afterward is not sufficient to implement every Pauli gate. As a result it is natural to consider a gate pattern shown in Fig. 4.4. We will show in the next

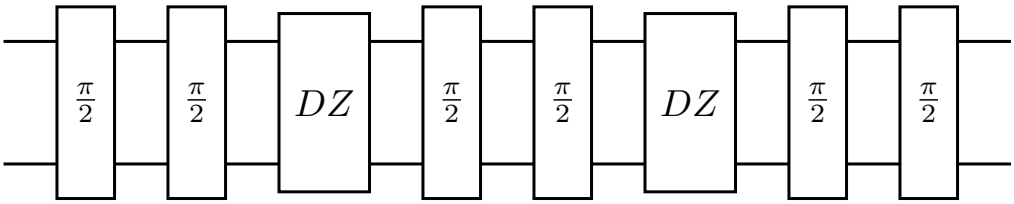


Figure 4.4: Circuit diagram showing the non-collective part of a native gate pattern for an arbitrary two-qubit Pauli gate. The circuit element labeled DZ indicates the differential-Z gate described previously, and the circuit element labeled $\frac{\pi}{2}$ indicates a collective $\pi/2$ pulse about an arbitrary equatorial-major-axis. The different two-qubit Pauli gates that can be achieved with this pattern are determined by the choice of the axis of the collective $\pi/2$ pulses. This pattern is also sufficiently general to produce the necessary Clifford gates for independent single-qubit two designs on two qubits, as discussed in Section 4.4. We assume that collective Z-rotations can be implemented freely, for example by adjusting the rotation axes of future gates as needed.

paragraph that the pattern in Fig. 4.4 is sufficiently general to implement any two-qubit Pauli gate. However, one natural question to address first is whether it is sufficient to consider a simplified form where each pair of collective $\pi/2$ pulses is actually a collective Pauli gate, as shown in Fig 4.5. However, direct computation shows that a circuit decomposition of this form is not sufficiently general to produce all 16 two-qubit Pauli gates. The possibilities for the achievable Pauli gates are shown in Table 4.3.

One natural way to understand the decomposition in Fig. 4.4 is to observe that allowing a

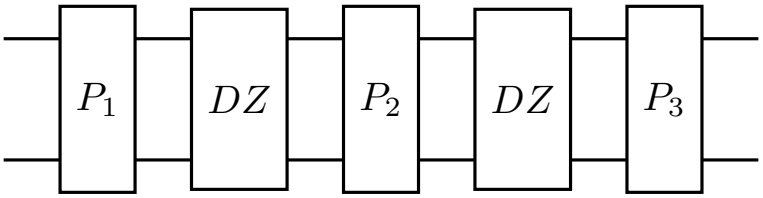


Figure 4.5: Circuit diagram where the pairs of $\pi/2$ pulses in Fig. 4.4 are replaced by Pauli gates. Circuits of this form are not sufficient to implement all two-qubit Pauli gates. The circuit element labeled DZ indicates the differential-Z gate, and the circuit element labeled P_1, P_2, P_3 indicate collective Pauli gates.

Table 4.3: Achievable Pauli gates using the pattern in Fig. 4.5. The entries in the second column indicate the choices for P_1, P_2, P_3 in Fig. 4.5. Choices that lead to the same total Pauli gate are separated by commas.

Total Pauli gate	Pauli gates P_1, P_2, P_3 in Fig. 4.5
$Z \otimes I$	III, XIX, YIY
$Y \otimes X$	IIX, XII
$X \otimes Y$	IIY, YII
$X \otimes X$	IXI, XXX, XYY, YXY, YYX
$I \otimes I$	IXX, IYY, XXI, YYI
$Z \otimes Z$	IXY, IYX, XYI, YXI
$Y \otimes Y$	IYI, XXY, XYX, YXX, YYY
$I \otimes Z$	XIY, YIX

single $\pi/2$ pulse before the non-collective part and a single $\pi/2$ pulse after is equivalent to allowing one of the four following options. First, the non-collective part can be conjugated by an $X_{\pi/2}$ pulse, which sends $Y \leftrightarrow Z$ in the list of non-collective Paulis $\{Z \otimes I, I \otimes Z, X \otimes X, Y \otimes Y\}$. This provides the ability to additionally achieve the Pauli gates $\{Y \otimes I, I \otimes Y, Z \otimes Z\}$. Second, the same conjugation can be performed but it is followed by a $X \otimes X$ gate. This provides the ability to additionally achieve $\{Z \otimes X, X \otimes Z, I \otimes I\}$. The third and fourth options are the same as the first and second options respectively, except with the the conjugation by an $X_{\pi/2}$ pulse replaced by conjugation by an $Y_{\pi/2}$ pulse. This will send $X \leftrightarrow Z$, in the previous lists, allowing for the

ability to additionally achieve $\{X \otimes I, I \otimes X, Z \otimes Y, Y \otimes Z\}$. In sum, with these four options, it is possible to achieve all 16 Pauli gates except $X \otimes Y$ and $Y \otimes X$. Allowing for another $\pi/2$ pulse before and another $\pi/2$ pulse after leads to the circuit pattern shown in Fig. 4.4. This pattern provides the capability to again conjugate by an $X_{\pi/2}$ pulse or a $Y_{\pi/2}$ pulse. With this capability it is now possible to achieve all 16 two-qubit Pauli gates. For example, conjugating the $X \otimes Z$ gate by a $X_{\pi/2}$ pulse is equivalent a $X \otimes Y$ gate, and conjugating the $Z \otimes X$ gate by a $X_{\pi/2}$ pulse is equivalent to a $Y \otimes X$ gate. We summarize this result in the following proposition.

Proposition 1. *All two-qubit Pauli gates can be implemented using the fixed gate pattern shown in Fig. 4.4, which consists of two differential-Z gates and six collective $\pi/2$ pulses about axes each chosen from the list $\{+X, -X, +Y, -Y\}$. The different Pauli gates are determined only by the axes of the $\pi/2$ pulses. It is not possible to implement an arbitrary two-qubit Pauli gate using a fixed gate pattern of differential-Z gates and collective $\pi/2$ where the fixed gate pattern contains fewer than two differential-Z gates or fewer than six collective $\pi/2$ pulses.*

With this decomposition, there is now redundancy in the sense that there are in general multiple ways to achieve a given two-qubit Pauli gate. It is reasonable to randomize over all possible methods of achieving each two-qubit Pauli gate, and in fact additional randomization of this form is generally helpful because it will help negate the effects of gate-dependent unitary errors.

4.4 Independent single-qubit two-designs on two qubits with differential Z

Implementing single-qubit two-designs independently on two qubits is a useful capability for performing single-qubit randomized benchmarking on both qubits simultaneously. This can help diagnose the presence of cross-talk errors between the qubits. For example, if performing a gate on one qubit causes an error on the other qubit, this will show up as an increase in the error rate, as measured by the exponential decay rate of the success probability. Although the fitting and analysis of randomized benchmarking simultaneously on two qubits is largely unchanged, the decomposition of the relevant gate sequences into native gates can present an interesting challenge. Here, we

provide a native-gate-decomposition for simultaneous single-qubit randomized benchmarking on two qubits where the native gates consist of the DZ gate and collective $X_{\pi/2}, Y_{\pi/2}$ pulses on the two qubits. We find that it is sufficient to use the pulse pattern in Fig. 4.4, where there are two DZ gates and six total collective $\pi/2$ pulses in pairs before, between, and after the two DZ gates. Temporarily considering equivalence classes of Clifford gates up to Pauli gates, the pattern of Clifford gates that we implement is shown in Fig. 4.6. In Fig. 4.6, there are three options for Clifford gates on each

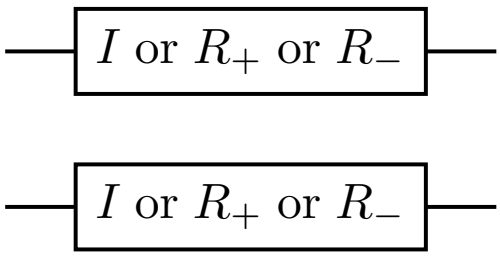


Figure 4.6: Circuit diagram showing the Clifford gates (equivalenced up to Pauli gates) in an implementation of simultaneous single-qubit randomized benchmarking on two qubits.

qubit, and the three options should be chosen independently for each qubit. In total there are then nine equally-weighted options for the total two-qubit (product) Clifford, equivalenced up to Pauli gates. For each of these nine options, two-qubit Pauli randomization must also be performed. For the cases in Fig. 4.6 where the joint Clifford is collective, $\{I \otimes I, R \otimes R, R^\dagger \otimes R^\dagger\}$, Pauli randomization can be achieved by using the method considered in the previous subsection. It remains to show that the gate pattern in Fig. 4.4 is sufficiently general to implement the remaining six two-qubit Cliffords, with Pauli randomization effectively included. A good starting point is to consider the different possibilities for non-collective Clifford gates on the two qubits, while temporarily ignoring Pauli gates. In addition to the non-collective pattern considered in the previous subsection, there are two further options. The first is to apply the collective gate R between the DZ gates and the second is to apply the collective gate $R^\dagger$ between the DZ gates. If the collective gate $R \otimes R$ is applied between the DZ gates, then the non-collective part is equivalent to the gate $R^\dagger \otimes R$. This can be checked directly in terms of the action on Pauli operators, and the circuit equivalence is

shown in Fig. 4.7. Note that appending a collective R or $R^\dagger$ gate to the non-collective gate $R^\dagger \otimes R$

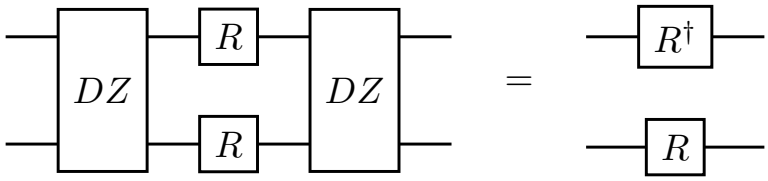


Figure 4.7: Circuit diagram showing the non-collective Clifford pattern obtained by applying the gate R between the DZ gates.

will produce the non-collective gates $I \otimes R^\dagger$ and $R \otimes I$ respectively.

Similarly, if the collective gate $R^\dagger \otimes R^\dagger$ is applied between the DZ gates, then the non-collective part is equivalent to the gate $R \otimes R^\dagger$. This equivalence can also be checked directly in terms of the action on Pauli operators, and is shown in Fig. 4.8. Appending a collective R or $R^\dagger$ gate to

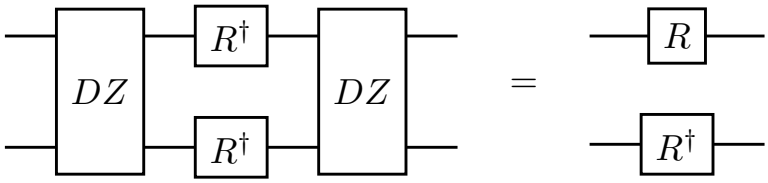


Figure 4.8: Circuit diagram showing the non-collective Clifford pattern obtained by applying the gate $R^\dagger$ between the DZ gates.

the non-collective gate $R \otimes R^\dagger$ will produce the non-collective gates $R^\dagger \otimes I$ and $I \otimes R$ respectively. In total, it follows that all nine of the relevant two-qubit product Clifford gates in Fig. 4.6 can be produced by either applying a collective gate in the set $\{I, R, R^\dagger\}$ between the DZ gates, and including an appropriate collective gate from the set $\{I, R, R^\dagger\}$ after the non-collective portion of the circuit. The nine choices of collective gates from these two sets are in one-to-one correspondence to the nine relevant two-qubit product Clifford gates in Fig. 4.6, and therefore should be evenly weighted in the overall ensemble. Furthermore, all of these options can be produced by gates of the pattern in Fig. 4.4.

It remains to show that Pauli randomization can also be performed by using gates in the pattern in Fig. 4.4. To address this, note that the implementation of the collective R or $R^\dagger$ rotation

in the collective-post portion of the circuit can be freely chosen to include an abitrary collective Pauli, according to Table 4.2. Furthermore by applying a gate pattern as in Fig. 4.9, it is possible to achieve any two-qubit Pauli, when prefixed to the beginning of the circuit. To see this, note that the

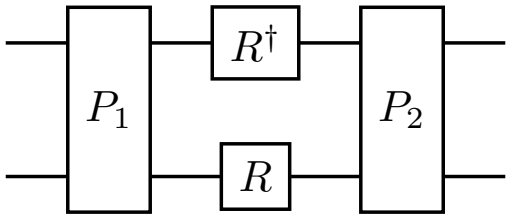


Figure 4.9: Circuit diagram showing the non-collective Clifford pattern obtained by applying the gate R between the DZ gates, along with collective Pauli gates before and after. This pattern is sufficient to perform Pauli randomization, along with the desired non-collective Clifford pattern.

two-qubit Paulis (of the form $P_a \otimes P_b$) can be grouped by their 'Pauli difference', meaning the Pauli Q such that $P_a = QP_b$ modulo phase. Any 'Pauli difference' can be achieved by appropriate choice of the Pauli P_2 that occurs after the non-collective part in Fig. 4.9, and then any two-qubit Pauli can be achieved by appropriate choice of the collective Pauli P_1 that occurs before the non-collective part in Fig. 4.9. Concretely, the four collective Pauli gates $\{I \otimes I, X \otimes X, Y \otimes Y, Z \otimes Z\}$, after propagating backward through the non-collective pattern become the Pauli gates $\{I \otimes I, Y \otimes Z, X \otimes Z, X \otimes Y\}$. Then, the four possibilities for collective Pauli gates applied beforehand provide enough freedom to implement any two-qubit Pauli, prefixed to the beginning of the circuit.

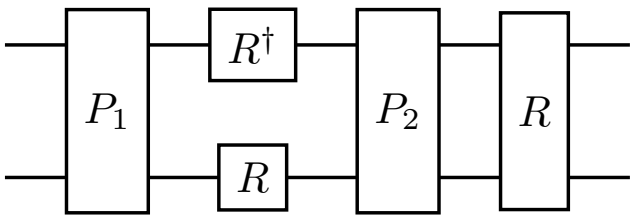


Figure 4.10: Circuit diagram showing the non-collective Clifford pattern obtained by applying the gate R between the DZ gates, along with collective Pauli gates before and after, along with a collective R gate afterward.

A similar technique applies for the circuits in Fig. 4.8 that are followed by a R or $R^\dagger$ gate.

The case of an R gate is shown in Fig. 4.10. By the same argument as before, the collective Pauli gates can be moved to the beginning of the circuit to show that Pauli randomization is still achieved with this pattern, even though the non-collective Clifford gate is different.

To achieve the remaining non-collective Clifford gates out of the nine required possibilities, it suffices to interchange R and $R^\dagger$ in the previous diagrams. This completes the argument that the pulse pattern in Fig. 4.4 is sufficiently general to implement simultaneous single-qubit 2-designs on two qubits. We summarize this result in the following proposition.

Proposition 2. *Independent single-qubit two-designs on two qubits can be implemented using the gate pattern shown in Fig. 4.4, which consists of two differential-Z gates and six collective $\pi/2$ pulses about major Bloch sphere axes. The different gates in the independent single-qubit two-designs are determined only by the axes of the $\pi/2$ pulses. Furthermore, it is not possible to implement independent single-qubit two-designs on two qubits with fewer differential-Z gates or fewer collective $\pi/2$ pulses, such that the different gates are determined only by the axes of the $\pi/2$ pulses.*

4.5 Two qubit two-design with a fixed pattern of entangling gates

We are interested in implementing a Clifford two-design for two qubits where there is a fixed pattern of entangling gates. Specifically, in the ensemble of Clifford's that comprise the two-design, every Clifford will have the same number of entangling gates. This way, the only difference between the different Cliffords in the ensemble comes from the different single-qubit gates. In contrast, when implementing a two-design by uniformly drawing two-qubit Clifford gates, the number of entangling gates will fluctuate in the range $[0, 3]$. For specifics about the number of two-qubit Cliffords that require different numbers of entangling gates, see Ref. [51]. We want a fixed number of entangling gates per step of the two-design because this will reduce fluctations in the success probability over random sequences at each sequence length, assuming that the errors in the entangling gate dominate over other error sources. It is also beneficial experimentally to have a fixed duty cycle of entangling gates, meaning that in each step of the two-design the entangling gates appear at the same time

relative to the start of the step. This implies that the times at which the entangling gates occur is fixed thoughout the benchmarking sequence, and the only difference between benchmarking sequences is the single-qubit gates that appear between the entangling gates. The CNOT gate is a standard choice for the two-qubit Clifford entangling gate, but all choices are equivalent up to single-qubit Clifford gates. Here the gate that we choose is a square-root of the ZZ Pauli gate, and can be expressed as $U = \cos\theta I \otimes I + i \sin\theta Z \otimes Z$ for $\theta = \pi/4$. We choose this gate because it has a particularly simple conjugate action on Pauli operators. Specifically, any Pauli $P_1 \otimes P_2$ that that commutes with $Z \otimes Z$ is unchanged, and if it anticommutes with $Z \otimes Z$ then it is mapped to $iZP_1 \otimes ZP_2$.

Here we describe a Clifford two-design on two qubits where each step of the two-design has two entangling Clifford gates. We note that at least two entangling Clifford gates per step of the two-design are required because a single entangling Clifford is not sufficient to perform Pauli mixing. The basic structure of this two-design is shown in Fig. 4.11. To summarize, this two-design

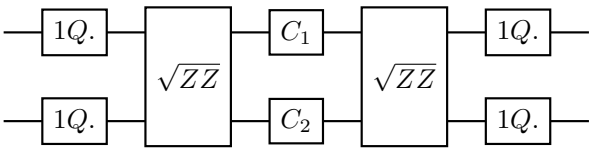


Figure 4.11: Schematic of the Clifford gates in the two-design that we present. The Clifford entangling gate is denoted $\sqrt{ZZ}$, and in each Clifford in the two-design there are two entangling gates in the pattern shown. The circuit element $1Q.$ refers to a single-qubit two-design. The circuit elements C_1 and C_2 indicate certain single-qubit Clifford gates that are drawn from a carefully chosen ensemble to ensure that the overall ensemble is a two-qubit two-design.

consists of the concatentation of three Clifford ensembles: (i) gates drawn independently from single-qubit two-designs independently on each qubit (ii) a pattern of two entangling gates with particular choices of local Clifford gates $C_1 \otimes C_2$ inbetween the entangling gates (iii) gates again drawn independently from single-qubit two-designs independently on each qubit. We note that performing independent two-designs on each qubit is sufficient to perform Pauli randomization on the two qubits. To show that the total ensemble is in fact a two-qubit two design it therefore suffices

to show that the ensemble consisting of the concatenation of (ii, iii) is Pauli-mixing. Furthermore, the fact that another independent single-qubit two-design is used in step (iii) implies that it is sufficient for step (ii) to achieve Pauli mixing up to local Pauli mixing. To explain this concept we find it convenient to first introduce the following notation for ensembles of Pauli operators. Let P denote one of Pauli $\{X, Y, Z\}$, so, for example, a two-qubit Pauli of the form PI means one of $\{XI, YI, ZI\}$. On two-qubits, all non-identity Paulis are of the form PI, IP, or PP. We will use the list notation $\{PI\}$ to denote the uniform ensemble of all Paulis of the form PI, and similarly for $\{IP\}$ and $\{PP\}$. The local Pauli mixing in step (iii) will transform any specific Pauli of the form PI to $\{PI\}$, and similarly Paulis of the form IP and PP will be transformed to $\{IP\}$ and $\{PP\}$ respectively. Furthermore, as a result of the independent single-qubit two-designs in step (i), any specific Pauli of the form PI will already have been mapped to $\{PI\}$, and similarly for Paulis of the form IP and PP. On two qubits, there are 15 non-identity Paulis, consisting of three Paulis of the form PI, three Paulis of the form IP and nine Paulis of the form PP. Therefore, to achieve two-qubit Pauli mixing it suffices for the gate ensemble in step (ii) to map each of $\{IP\}, \{PI\}, \{PP\}$ to a distribution of Paulis consisting of Paulis of the form IP with probability $1/5$, Paulis of the form PI with probability $1/5$, and Paulis of the form PP with probability $3/5$. The terminology 'Pauli mixing up to local Pauli mixing' that was used earlier in this paragraph means exactly this requirement.

To construct gate ensembles for step (ii) that satisfy this requirement, it is useful to first compute the output probability distributions over Paulis that $\{IP\}, \{PI\}, \{PP\}$ get mapped to under various possibilities for the gates C_1 and C_2 that make up the ensemble. It is in fact sufficient to only consider the outputs of $\{IP\}$ and $\{PI\}$ because if the outputs of these have the correct Pauli fractions, then the outputs of $\{PP\}$ must also have the correct Pauli fractions. This follows essentially because each Clifford gate is an automorphism on the Pauli group, which implies that the uniform distribution over non-identity Paulis is preserved when any Clifford ensemble is applied. Therefore, if the uniform distribution over any subset of non-identity Paulis is mapped to the uniform distribution over non-identity Paulis, then the same holds for the complement of that

subset. Since Z-rotations commute with the root-ZZ gate, it is sufficient to consider the following four possibilities for the gates C_1 and C_2: $\{I, R, R^\dagger, X_{\pi/2}\}$. There are then 16 total possibilities for selecting both C_1 and C_2 independently from this list. The Pauli fractions for these possibilities are shown in the following table.

Table 4.4: Output Pauli distributions obtained from various possibilities for the gates C_1, C_2 in the ensemble. For each row, the first column contains a list of choices for C_1 and C_2 that all have the same Pauli output distribution. The next three columns contain the probabilities in the Pauli output distribution when the input is $\{PI\}$, and the final three columns contain the probabilities in the Pauli output distribution when the input is $\{IP\}$.

Choice of $C_1 \otimes C_2$	$PI \to PI$	$PI \to IP$	$PI \to PP$	$IP \to IP$	$IP \to PI$	$IP \to PP$
$I \otimes I$	1	0	0	1	0	0
$I \otimes X_{\pi/2}$, $I \otimes R$, $I \otimes R^\dagger$, $X_{\pi/2} \otimes I$, $R \otimes I$, $R^\dagger \otimes I$	$\frac{1}{3}$	0	$\frac{2}{3}$	$\frac{1}{3}$	0	$\frac{2}{3}$
$R \otimes R$, $R \otimes R^\dagger$, $R \otimes X_{\pi/2}$ $R^\dagger \otimes R$, $R^\dagger \otimes R^\dagger$, $R^\dagger \otimes X_{\pi/2}$, $X_{\pi/2} \otimes R$, $X_{\pi/2} \otimes R^\dagger$, $X_{\pi/2} \otimes X_{\pi/2}$	0	$\frac{1}{3}$	$\frac{2}{3}$	0	$\frac{1}{3}$	$\frac{2}{3}$

Based on this table, an appropriate gate ensemble for step (ii) can be constructed by selecting C_1 and C_2 from the first row with probability 1/10, from the second row with probability 3/10, and from the third row with probability 6/10. With this gate ensemble, $\{IP\}$ and $listbPI$ are both mapped to a distribution of Paulis consisting of Paulis of the form IP with probability 1/5, Paulis of the form PI with probability 1/5, and Paulis of the form PP with probability 3/5. These are the correct output probabilities to perform Pauli mixing up to local Pauli mixing. We summarize this result in the following proposition.

Proposition 3. *The gate pattern shown in Fig. 4.11 implements a two-qubit two-design when in Fig. 4.11, $C_1 \otimes C_2$ is drawn from the list of gates in the first row of Table 4.4 with probability $\frac{1}{10}$, the second row with probability $\frac{3}{10}$, and the third row with probability $\frac{6}{10}$. Any distribution over the individual gates within each comma-separated list in Table 4.4 is sufficient for this purpose.*

4.6 Two qubit two-design with differential Z and root-ZZ

We are interested in implementing a two qubit two-design for randomized benchmarking where the native gates are collective $\pi/2$ pulses, the DZ gate, and a two-qubit Clifford entangling gate. To implement the two-qubit two-design in the previous subsection, it remains only to implement the three possibilities for the gates C_1 and C_2 using the DZ gate and collective rotations. One possibility for achieving this uses a pattern of a collective rotation, followed by DZ, followed by another collective rotation. The analysis can be simplified by using the fact that Z-rotations commute with both the DZ gate and the root-ZZ gate. First, to achieve $C_1 \otimes C_2 = I \otimes I$, it suffices to implement the $I \otimes I \circ DZ \circ I \otimes I$. Second, to achieve $C_1 \otimes C_2 = X_{\pi/2} \otimes I$, it suffices to implement $R \otimes R \circ DZ \circ R^\dagger \otimes R^\dagger$. Third, to achieve $C_1 \otimes C_2 = R \otimes R^\dagger$ it suffices to implement $R \otimes R \circ DZ \circ R^\dagger \otimes R^\dagger$. Combined with all the other decompositions used in the previous subsections, this completes the implementation of a two-qubit two-design using root-ZZ, DZ, and collective rotations as the native gates. This implementation has a fixed pattern of root-ZZ and DZ gates, and the only differences between the gates in the ensemble are controlled by the phases of the collective rotations.

Chapter 5

Models of fully randomized benchmarking

This chapter is substantially based on work published in Ref. [45]. Here we consider several models of fully randomized benchmarking that generalize the basic model. First, we consider a model where the step error is constant throughout the random sequence of an individual trial, but is drawn from a probability distribution $\tilde{\sigma}(\varepsilon)$ independently for each trial. Accordingly, the success probability $P(n)$ is

$$P(n) = \frac{1}{D} + \frac{1}{\alpha}(1 - \alpha\theta_0) \int d\varepsilon \tilde{\sigma}(\varepsilon)(1 - \alpha\varepsilon)^n, \tag{5.1}$$

where $\alpha = \frac{D}{D-1}$ and the parameter θ_0 describes the SPAM error. As written, this model is parametrized by θ_0 and $\tilde{\sigma}$, which is an infinite dimensional parameter. Below we show that only $N+1$ parameters are relevant if the sequence length is bounded by N. The basic model corresponds to the case of $\tilde{\sigma}(\epsilon) = \delta(\epsilon - \theta_1)$, where θ_1 is the step error and δ denotes the Dirac delta distribution. For a general distribution $\tilde{\sigma}(\varepsilon)$, we denote the mean of $\tilde{\sigma}(\varepsilon)$ by θ_1 and interpret it as the parameter analogous to step error. At times it is convenient to shift the probability distribution $\tilde{\sigma}(\varepsilon)$ by its mean θ_1. We define $\sigma(\varepsilon) = \tilde{\sigma}(\varepsilon + \theta_1)$ so that

$$P(n) = \frac{1}{D} + \frac{1}{\alpha}(1 - \alpha\theta_0) \int d\varepsilon \sigma(\varepsilon)(1 - \alpha\theta_1 - \alpha\varepsilon)^n. \tag{5.2}$$

The parameters of the model are now θ_0, θ_1, and σ,where the probability distribution σ is constrained to have mean 0 and support in $[-\theta_1, 1-\theta_1]$. The basic model is recovered with $\sigma(\varepsilon) = \delta(\varepsilon)$. Applying

the binomial expansion to the n'th power in the expression for $P(n)$ gives

$$P(n) = \frac{1}{D} + \frac{1}{\alpha}(1-\alpha\theta_0)\left((1-\alpha\theta_1)^n + \sum_{k=2}^{n}\binom{n}{k}(1-\alpha\theta_1)^{n-k}(-\alpha)^k \int d\varepsilon\sigma(\varepsilon)\varepsilon^k\right), \tag{5.3}$$

where the $k = 1$ term vanishes by the assumption that $\sigma(\varepsilon)$ has mean 0. This motivates the introduction of the moment parameters $\theta_k := \int d\varepsilon\sigma(\varepsilon)\varepsilon^k$ for $k = 2$ to N. In terms of these parameters the success probability can be written

$$P(n) = \frac{1}{D} + \frac{1}{\alpha}(1-\alpha\theta_0)\left((1-\alpha\theta_1)^n + \sum_{k=2}^{n}\binom{n}{k}(1-\alpha\theta_1)^{n-k}(-\alpha)^k\theta_k\right). \tag{5.4}$$

We refer to this model as the 'moments model' and we refer to the parameters θ_k for $k \geq 2$ as the moments parameters. For practical use, the moments parameters are truncated for k larger than some $k_{\max}$, so that $\theta_k = 0$ for $k > k_{\max}$. For example, when we make certain comparisons to published experiments in Section 7.2, we use the moments model with two non-zero moments parameters θ_2, θ_3 for a total of four parameters. When we design the experiments in Chapter 8 we also use the moments model with four total parameters. When we analyze those experiments we use the moments model with three total parameters, where we remove θ_3. In that case, we report $\sqrt{\theta_2}$ because this is on the same scale as θ_1 and is the standard deviation of σ if θ_2 comes from a true probability distribution. We note that if all the moments parameters are zero, the moments model reduces to the basic model with spam error θ_0 and step error θ_1. We note that the parameters $\theta_2, \ldots$ are the mean-subtracted moments of the original distribution $\tilde{\sigma}$. For later use, we denote the moments of $\tilde{\sigma}$ as $\tilde{\theta}_k = \int d\varepsilon\varepsilon^k\tilde{\sigma}(\varepsilon)$. Treating θ_1 as a constant, for $k \geq 2$, θ_k is an affine linear combination of $\tilde{\theta}_2, \ldots \tilde{\theta}_k$, and similarly for $\tilde{\theta}_k$ in terms of the $\theta_2, \ldots, \theta_k$.

The moments model is universal in the following sense. In the absence of trial-dependent step errors, the most general benchmarking model has arbitrary success probabilities $P(n)$ depending on n. We show that any such success probabilities can be modeled by a suitable choice of parameters of the moments model, provided the implicit linear restrictions on the moment parameters due to

positivity and support constraints of the probability distribution σ are lifted. We first write the moments model in terms of the moments of $\tilde{\sigma}$,

$$P(n) = \frac{1}{D} + \frac{1}{\alpha}(1 - \alpha\theta_0)\left(1 + \sum_{k=1}^{n} \binom{n}{k}(-\alpha)^k \tilde{\theta}_k\right). \tag{5.5}$$

The parameter θ_0 linearly determines and is determined by $P(0)$. For the remaining probabilities, we fix θ_0. Eq. 5.5 establishes a linear relationship between the $P'(n) = P(n) - P(0)$ for $n \geq 1$ and the $\tilde{\theta}_k$ for $k \geq 1$ of the form $P'(n) = \sum_{k\geq 1} M_{nk}\tilde{\theta}_k$. The matrix M_{nk} is lower triangular with diagonal entries $M_{nn} = (P(0) - 1/D)(-\alpha)^n$. Here we assume that $P(0) \neq 1/D$. If $P(0) = 1/D$, then the initial state would be completely depolarized and the choice of moment parameters would be irrelevant. With this assumption the diagonal entries M_{nn} are nonzero, and M therefore has a lower triangular inverse. It follows that in the absence of constraints on the $\tilde{\theta}_k$, all possible $P(n)$ can be modeled with a choice of the moment parameters. If the maximum sequence length under consideration is $n_{\max}$, we can truncate the matrix at $n = N$ and model $P(0), \ldots P(n)$ with a choice of $\tilde{\theta}_1, \ldots \tilde{\theta}_N$, or equivalently $\theta_1, \ldots, \theta_N$, for any fixed θ_0.

In addition to the basic model and the moments model, another model of experimental interest is one where the errors in a sequence experience drift as a function of position within the sequence. To motivate this behavior we consider a miscalibrated single-qubit gate where the miscalibration drifts linearly as a function of time but is reset at the beginning of each sequence. Concretely, we consider a nominal gate U that implements a π rotation about the x-axis of the Bloch sphere and can be expressed as $U = \exp[-i(\pi/2)X]$, where X is the Pauli-X operator. We denote the action of the possibly miscalibrated gate by $\tilde{U} = \exp[-i(\pi/2 + \phi)X]$, where ϕ is an error parameter that describes the angle of erroneous rotation. The average fidelity of $\tilde{U}$ with the nominal gate U is equal to $1/3 + 2\cos^2(\phi)/3$. A plausible error model is that the erroneous rotation ϕ depends linearly on time, which could correspond physically to a linear drift of Rabi frequency. If the gate is perfectly calibrated at $t = 0$, expanding to lowest order for short times shows that the error will grow quadratically. If $\phi \neq 0$ at $t = 0$ and the expansion is performed to second order, the error will

in general have both linear and quadratic dependence for short times. Altogether, this motivates consideration of the following approximate model,

$$P(n) = \frac{1}{D} + \frac{1}{\alpha}(1 - \alpha\theta_0)\left(\prod_{k=1}^{n}(1 - \alpha(A + Bk + Ck^2))\right), \tag{5.6}$$

where θ_0 is a SPAM parameter and A, B, C are parameters that govern the linear and quadratic drift. One question of possible experimental relevance is whether this drift model can be distinguished from the moments model when the parameters of the moments model are moments of a true probability distribution $\tilde{\sigma}(\varepsilon)$. Here we show that this is indeed possible for at least one region of the space of parameters A, B, C. In particular, we consider $\theta_0 = 0, C = 0$, and approximate Eq. 5.6 to lowest order in B for $B > 0$. When we determine the moments parameters that match this model we find that $\theta_2 < 0$ for this region of parameter space, which is impossible for a true second moment. To determine the matching moments parameters, we follow the procedure outlined in the previous paragraph. In the approximate linear drift model we have $P(1) = 1/D + (1 - \alpha A - \alpha B)/\alpha$ and in the moments model we have $P(1) = 1/D + (1 - \alpha\theta_1)/\alpha$. We equate these to determine θ_1 and find $\theta_1 = A + B$. Similarly, we then equate $P(2)$ in both models

$$(1 - \alpha\theta_1)(1 - \alpha\theta_1 - \alpha B) = (1 - \alpha\theta_1)^2 + \alpha^2\theta_2. \tag{5.7}$$

Solving for θ_2 we find $\theta_2 = -B(1 - \alpha\theta_1)/\alpha$, which satisfies $\theta_2 < 0$ when $B > 0$. Equating $P(3)$ in both models we find that θ_3 is of order $O(B^2)$, and by induction we find that $P(k)$ is also of order $O(B^2)$ or higher when $k \geq 3$, so the parameters θ_k for $k \geq 3$ can be dropped to good approximation when B is small. In total, we conclude that if the true model is the linear drift model for small positive B, an analysis using the moments model with parameters $\theta_0, \theta_1, \theta_2$ would likely find $\theta_2 < 0$ in the large data limit, which is impossible for a true second moment. Thus, it would be possible to detect such a drift in a randomized benchmarking experiment.

Chapter 6

Optimized experiment design for RB

This chapter is substantially based on work published in Ref. [45].

6.1 FRB experiment design

We describe a procedure to optimize the design of a fully randomized benchmarking experiment for statistical performance according to an arbitrary pre-chosen statistical model $P_{\boldsymbol{\theta}}(n)$. The goal of the optimization is to minimize the anticipated uncertainty of the inference of the parameter of interest, θ_{i_0}. In many cases the parameter of interest is the step error θ_1. The optimization is performed by linearizing the model around a reference point $\boldsymbol{\theta}^{(0)}$ and constructing a linear estimator for θ_{i_0} that has minimum variance and is insensitive to the other parameters at the reference point. The standard deviation of the optimal linear estimator is the uncertainty of inference of θ_{i_0} in the linearized model, and is therefore a 'first-order' approximation of the anticipated uncertainty of inference of θ_{i_0} in the actual model. The accuracy of this approximation depends on the 'closeness' of the reference point to the true point and on the nearby 'curvature' of the statistical model. For more details we refer to Refs. [53, 61]. We assume that the models of fully randomized benchmarking considered here are reasonably well-behaved in the sense that the uncertainty estimates from a linearized model around a reference point obtained from prior calibration are close to the true uncertainties obtained in an experiment.

The optimized experiment design that we describe here is called C-optimal design, and it can be formulated as a linear program [21, 35]. Other types of optimization with different objectives

are also possible. For example, a general formulation of C-optimal design minimizes the variance of an arbitrary linear combination of model parameters. Similarly, another objective could be to jointly minimize a weighted sum of variances of several parameters. All of these objectives lead to convex optimization problems and have a close connection to Fisher information [35, 53, 61]. In Section 6.2 we provide a description of the relationship between C-optimal design and Fisher information. For more information and details about these types of optimized experiment design, we refer to Refs. [25, 35, 61, 63].

Here we present the optimization procedure to minimize the anticipated uncertainty of a single parameter θ_{i_0}, specifically in the context of designing experiments for fully randomized benchmarking. The optimization is performed over the sequence lengths n_j and numbers of trials w_j at each sequence length, which together comprise the experiment design. The optimization is further subjected to a constraint on the total experimental time T. Altogether, the inputs to the optimization are: the statistical model $P_{\boldsymbol{\theta}}(n)$, the reference point $\boldsymbol{\theta}^{(0)}$, the pre-chosen parameter θ_{i_0}, the maximum sequence length $n_{\max}$ that is available in an experiment, and a list t_n of the amount of experiment time that it takes to experimentally perform a sequence of length n. The details of the optimization procedure are as follows. Let $P_{\boldsymbol{\theta}^{(0)}}(n)$ denote the success probabilities at the reference point $\boldsymbol{\theta}^{(0)}$ as a function of the sequence length n, and let δp_n denote small changes in $P_{\boldsymbol{\theta}}(n)$ around $P_{\boldsymbol{\theta}^{(0)}}(n)$, so $P_{\boldsymbol{\theta}}(n) = P_{\boldsymbol{\theta}^{(0)}}(n) + \delta p_n$. Any differentiable model can be linearized around the reference point $P_{\boldsymbol{\theta}^{(0)}}(n)$. Let $L_{ni} = \frac{\partial P(n)}{\partial \theta_i}|_{\boldsymbol{\theta}^{(0)}}$ be the gradient of the model at the reference point. Then we can write

$$\delta p_n = \sum_i L_{ni} \delta\theta_i, \tag{6.1}$$

to first order in the $\delta\theta_i$. For the purpose of optimization we now assume the linearized model.

Let $\delta\hat{p}_n$ denote the empirical estimator of δp_n obtained from the observed frequency of successes after subtracting the probability of success at the reference point. If we denote the

observed number of success counts by $\hat{c}_n$, we have

$$\delta\hat{p}_n = \frac{\hat{c}_n}{w_n} - P_{\boldsymbol{\theta}^{(0)}}(n). \tag{6.2}$$

We consider linear estimators $\hat{A}$ of the form

$$\hat{A} = \sum_n C_n \delta\hat{p}_n, \tag{6.3}$$

where we choose the coefficients C_n so that $\hat{A}$ estimates $\delta\theta_{i_0}$ with minimum variance at the reference point. Concretely, $\hat{A}$ estimates $\delta\theta_{i_0}$ if the coefficients satisfy $\sum_n C_n L_{ni} = \delta_{ii_0}$, which implies that $\langle\hat{A}\rangle = \delta\theta_{i_0}$ and that $\hat{A}$ is insensitive to the other parameters $\theta_{i\neq i_0}$. Of the many linear estimators that satisfy these constraints, we wish to construct one with the minimum variance at the reference point, subject to the additional constraint that the experiment takes a total time T. If a trial with a sequence of length n takes a time t_n, then this constraint can be expressed as $\sum_n w_n t_n = T$. At the reference point the variance v_n of $\delta\hat{p}_n$ is determined by the number of trials w_n and the binomial statistics of a single trial according to

$$v_n = \operatorname{var}\delta\hat{p}_n = \frac{P_{\boldsymbol{\theta}^{(0)}}(n)(1 - P_{\boldsymbol{\theta}^{(0)}}(n))}{w_n}. \tag{6.4}$$

It follows that the variance V of $\hat{A}$ satisfies

$$V = \operatorname{var}\hat{A} = \sum_n C_n^2 \frac{v_n}{w_n}, \tag{6.5}$$

where we have used the independence of $\delta\hat{p}_n$ for different n. In total, to construct the optimal linear estimator we minimize V jointly over the C_n and the w_n, subject to the constraints $\sum_n C_n L_{ni} = \delta_{ii_0}$ and $\sum_n w_n t_n = T$. We are free to optimize over the C_n and the w_n in either order. The optimization of the w_n at fixed C_n yields a closed form solution, which can then be optimized over choices of the C_n by a linear program as explained in the following paragraph.

We now fix the C_n and minimize V over choices of the w_n subject to the constraint $\sum_n w_n t_n = T$. For this we introduce the Lagrange multiplier λ and find the critical points with respect to w_n of

$$V_\lambda = \sum_n C_n^2 \frac{v_n}{w_n} + \lambda \left(\sum_n w_n t_n - T \right). \tag{6.6}$$

Differentiating by w_n and solving for w_n gives the critical point equations

$$w_n = \frac{|C_n|\sqrt{v_n}}{\sqrt{\lambda}\sqrt{t_n}}, \tag{6.7}$$

which we substitute back into the expression for V_λ to obtain

$$\begin{aligned} V_{\lambda,\text{opt}} &= \sum_n |C_n| \sqrt{\lambda} \sqrt{v_n t_n} + \lambda \left(\sum_n \frac{|C_n|\sqrt{v_n t_n}}{\sqrt{\lambda}} - T \right) \\ &= 2\sqrt{\lambda} \sum_n |C_n| \sqrt{v_n t_n} - \lambda T. \end{aligned} \tag{6.8}$$

Substituting the solution for w_n into the constraint and rearranging terms constrains λ according to $\lambda = \left(\sum_n |C_n|\sqrt{v_n t_n}/T\right)^2$. Substituting this value for λ into the expression for $V_{\lambda,\text{opt}}$ gives the minimum variance for fixed C_n

$$V_{\text{opt}} = \frac{1}{T} \left(\sum_n |C_n| \sqrt{v_n t_n} \right)^2. \tag{6.9}$$

To minimize V_{opt} over the constrained values of C_n, it suffices to minimize the quantity $F = \sum_n |C_n|\sqrt{v_n t_n}$ with the linear constraints $\sum_n C_n L_{ni} = \delta_{ii_0}$. This can be done by means of a linear program using a standard method for handling the absolute values [6]. The resulting linear program

is

$$\begin{aligned} \text{Minimize: } & F = \sum_n \tilde{C}_n \sqrt{v_n t_n} \\ \text{Variables: } & (C_n)_{n=1}^{n_{\max}}, \left(\tilde{C}_n\right)_{n=1}^{n_{\max}} \\ \text{Subject to: } & \text{for all } n, \tilde{C}_n \geq 0, \\ & \text{for all } n, -\tilde{C}_n \leq C_n \leq \tilde{C}_n, \\ & \sum_n C_n L_{ni} = \delta_{ii_0}. \end{aligned} \tag{6.10}$$

Once the optimal C_n are determined, the optimal w_n can be determined by substitution into Eq. 6.7. We summarize these results in Proposition 4.

Proposition 4. *Let $P_{\boldsymbol{\theta}}(n)$ be a statistical model for fully randomized benchmarking in terms of parameters $\boldsymbol{\theta}$ and let $\boldsymbol{\theta}^{(0)}$ be a reference point. Let $P_{\boldsymbol{\theta}}(n)$ be differentiable at $\boldsymbol{\theta}^{(0)}$ with respect to each parameter in $\boldsymbol{\theta}$ for each n. An experiment design for a fully randomized benchmarking experiment that minimizes the uncertainty of a particular model parameter θ_{i_0}, to 'first-order' accuracy around the reference point, can be chosen by first performing the optimization in Eq. 6.10 to determine the coefficients C_n and then substituting those coefficients into Eq. 6.7 to determine the optimal weights w_n. The optimal weights w_n can be rounded to integer values to design a realistic experiment.*

After this substitution the optimal w_n will be non-negative real numbers, and must be rounded to integer values to design a real experiment. In practice we find that the rounding has only a small effect on the statistical power of the experiment. In total, this optimization method determines the experiment design that has the minimum variance of the best linear estimator of the parameter $\boldsymbol{\theta}^{(0)}$ in the linearized model at the reference point. This variance can be computed in terms of the optimal C_n according to Eq. 6.9. As we describe in Section 7.1, for analysis of randomized benchmarking data we use the maximum likelihood-estimator in the full model. In the limit of a large amount of collected data we expect the variance of the maximum likelihood-estimator in the full model to match the variance of the best linear estimator of $\boldsymbol{\theta}^{(0)}$ in the linearized model. In any

realistic scenario discrepancies can arise between the two variances as a result of the finite amount of collected data, or because the reference point used for the optimization differs from the true point. In this sense, the anticipated variance of the optimal experiment design in Eq. 6.9 should be regarded as approximate, although we expect good agreement in well-behaved cases. For example, in the randomized benchmarking experiments run at NIST that we describe in Chapter 8, we find that the anticipated variance closely matches the observed variance.

6.2 Evaluate anticipated uncertainty for a given experiment design

If the experiment design is fixed, the coefficients (C_n) of the optimal linear estimator for a parameter θ_{i_0} at the reference point $\boldsymbol{\theta}^{(0)}$ can be computed as follows. In the notation of Section 6.1, the goal is to minimize the variance in Eq. 6.5, which is $v = \sum_n \frac{C_n^2 v_n}{w_n}$, subject to the linear constraints $\sum_n C_n L_{ni} = \delta_{ii_0}$. This is a quadratic program with linear constraints and can be written in matrix notation as a minimization of $\mathbf{c}^\top Q \mathbf{c}$ subject to $E\mathbf{c} = \mathbf{d}$, where $\mathbf{c}$ is the list of coefficients (C_n) in vector form, Q is a diagonal matrix with diagonal elements $Q_{nn} = \frac{v_n}{w_n}$, the constraint matrix E has elements $E_{in} = L_{ni}$, and $d_i = \delta_{ii_0}$. The solution for $\mathbf{c}$ can be obtained by solving

$$\begin{bmatrix} Q & E^\top \\ E & 0 \end{bmatrix} \begin{bmatrix} \mathbf{c} \\ \lambda \end{bmatrix} = \begin{bmatrix} 0 \\ \mathbf{d} \end{bmatrix} \tag{6.11}$$

where λ is a vector of Lagrange multipliers [16]. This can be achieved by using the standard formula for the inverse of a block matrix [16, 46], and the solution for $\mathbf{c}$ is

$$\mathbf{c} = Q^{-1}E^\top (EQ^{-1}E^\top)^{-1}\mathbf{d}. \tag{6.12}$$

As a result, the ith column of the matrix $M = Q^{-1}E^\top (EQ^{-1}E^\top)^{-1}$ has the coefficients of the optimal linear estimator for the ith parameter θ_i.

6.3 Relationship to Fisher information

For a given experiment design and reference point $\boldsymbol{\theta}^{(0)}$, we show that the covariance matrix V of the optimal linear estimators obtained in Section 6.2 is equal to the inverse of the Fisher information matrix. This is well established in the literature on experiment design and Fisher information [25, 53, 54, 61], and for convenience we provide a derivation here. As we show in Section 6.2, the ith column of the matrix $M = Q^{-1}E^\top(EQ^{-1}E^\top)^{-1}$ has the coefficients of the optimal linear estimator for the ith parameter θ_i. Therefore, the covariance matrix V of these linear estimators satisfies $V = M^\top QM$, which evaluates to

$$V = M^\top QM = (EQ^{-1}E^\top)^{-1\top}EQ^{-1}QQ^{-1}E^\top(EQ^{-1}E^\top)^{-1}. \tag{6.13}$$

The matrix Q is diagonal, so we have $(EQ^{-1}E^\top)^{-1\top} = (EQ^{-1}E^\top)^{-1}$, and this simplifies to

$$V = (EQ^{-1}E^\top)^{-1}. \tag{6.14}$$

The Fisher information matrix for a single trial of sequence length n can be obtained according to the standard formula [53]

$$F_{ii'}(n) = \left\langle \frac{\partial}{\partial\theta_i}\log p \frac{\partial}{\partial\theta_{i'}}\log p \right\rangle_{\boldsymbol{\theta}^{(0)}}, \tag{6.15}$$

where the expectation value is taken over the two measurement outcomes 'success' and 'failure', and p is the likelihood of getting a particular outcome. The subscript $\boldsymbol{\theta}^{(0)}$ indicates that the formula is evaluated at the reference point $\boldsymbol{\theta}^{(0)}$. Evaluating this for the two-outcome measurement for a single trial of sequence length n gives

$$\begin{aligned} F_{ii'}(n) = P_{\boldsymbol{\theta}}(n)\frac{\partial}{\partial\theta_i}\log\left[P_{\boldsymbol{\theta}}(n)\right]\frac{\partial}{\partial\theta_{i'}}\log\left[P_{\boldsymbol{\theta}}(n)\right]\Big|_{\boldsymbol{\theta}^{(0)}} + \\ \left[1-P_{\boldsymbol{\theta}}(n)\right]\frac{\partial}{\partial\theta_i}\log\left[1-P_{\boldsymbol{\theta}}(n)\right]\frac{\partial}{\partial\theta_{i'}}\log\left[1-P_{\boldsymbol{\theta}}(n)\right]\Big|_{\boldsymbol{\theta}^{(0)}}. \end{aligned} \tag{6.16}$$

Using the fact that $E_{ni} = \frac{\partial}{\partial \theta_i} P_{\boldsymbol{\theta}}(n)\Big|_{\boldsymbol{\theta}^{(0)}}$, this simplifies to

$$F_{ii'}(n) = \frac{E_{ni} E_{ni'}}{P_{\boldsymbol{\theta}^{(0)}}(n)(1 - P_{\boldsymbol{\theta}^{(0)}}(n))}. \tag{6.17}$$

Weighting by the number of trials w_n at sequence length n and summing over n gives a total Fisher information matrix of

$$F_{ii'} = \sum_n \frac{E_{ni} w_n E_{i'n}}{P_{\boldsymbol{\theta}^{(0)}}(n)(1 - P_{\boldsymbol{\theta}^{(0)}}(n))}. \tag{6.18}$$

Using the fact that, as in Section 6.1, the matrix Q is diagonal with entries $Q_{nn} = w_n / v_n$ with $v_n = [P_{\boldsymbol{\theta}^{(0)}}(n)(1 - P_{\boldsymbol{\theta}^{(0)}}(n))]$, this simplifies to $F = EQ^{-1}E^\top$. Therefore, comparison to Eq. 6.14 shows that the covariance matrix of the optimal linear estimators V is equal to the inverse of the Fisher information matrix F. In this sense, the optimization procedure described in Section 6.1 is Fisher-optimal.

6.4 Experiment design optimization focusing on more than 1 parameter

Here we show how to design a fully randomized benchmarking experiment to minimize the sum of the variance of several parameters in the statistical model. Without loss of generality, we assume that we are interested in minimizing the sum of the variances of the first $i_{\max}$ parameters. Instead of constructing a single linear estimator with coefficients $\{C_n\}$ for a particular parameter θ_i, we now construct a family of linear estimators $\{C_n^{(i)}\}_{i-1}^{i_{\max}}$, each of which estimates the ith parameter respectively. The coefficients of these linear estimators can be represented as a matrix C_n^i with $i_{\max}$ rows and N columns. If these coefficients are fixed, the optimal weights $\{w_n\}$ can again be determined analytically by Lagrange multipliers as follows. First, assume we want to minimize the trace of the covariance matrix of this family of linear estimators. The objective function, with a Lagrange multiplier to encode the constraint, is

$$V = \sum_{i,n} (C_n^i)^2 \frac{v_n}{w_n} + \lambda \left(\sum_n w_n t_n - T \right). \tag{6.19}$$

After taking derivatives with respect to w_n and setting them to zero, we have the following,

$$w_n = \sqrt{\sum_i (C_n^i)^2} \frac{\sqrt{v_n}}{\sqrt{t_n}} \frac{1}{\sqrt{\lambda}}. \tag{6.20}$$

The constraint $\sum_n w_n t_n = T$ can then be used to solve for $\sqrt{\lambda}$,

$$\sqrt{\lambda} = \frac{\sum_{n'} \sqrt{\sum_{i'} (C_{n'}^{i'})^2} \sqrt{v_{n'} t_{n'}}}{T}. \tag{6.21}$$

Substitution into the objective function leads to

$$V = \frac{1}{T} \left(\sum_n \sqrt{\sum_i (C_n^i)^2} \sqrt{v_n t_n} \right) \left(\sum_{n'} \sqrt{\sum_{i'} (C_{n'}^{i'})^2} \sqrt{v_{n'} t_{n'}} \right). \tag{6.22}$$

Therefore, the objective function can be minimized by minimizing $V' := \left(\sum_n \sqrt{\sum_i (C_n^i)^2} \sqrt{v_n t_n} \right)$ subject to the linear constraints that the ith linear estimator, with coefficients $\{C_n^i\}_n$, is for the ith parameter, meaning $\sum_n C_n^i L_{nj} = \delta_{ij}$ for each i.

This can be reformulated as a second-order-cone-problem by introducing the N additional variables D_n that are intended to represent $\sqrt{\sum_i (C_n^i)^2}$ respectively. The objective function then becomes $\sum_n D_n \sqrt{v_n t_n}$, and the additional constraints are $\sum_i (C_n^i)^2 \leq D_n$ for each n. At the optimum, this inequality will be tight because if $\sum_i (C_n^i)^2 < D_n$, then D_n can be made smaller while still satisfying the constraints, which means that the objective function can be made yet smaller. This has the requisite structure of a second-order-cone problem in $N(i_{\max} + 1)$ variables. The linear constraints on the original problem can be transferred to the new problem. In total, the resulting

optimization is as follows

$$\begin{aligned}
\text{Minimize: } & \sum_n D_n \sqrt{v_n t_n} \\
\text{Variables: } & (D_n)_{n=1}^{n_{\max}}, \left(C_n^i\right)_{n=1,i=1}^{n_{\max},i_{\max}} \\
\text{Subject to: } & \text{for all } n,\ D_n \geq 0, \\
& \text{for all } n,\ \sum_i (C_n^i)^2 \leq D_n, \\
& \text{for } i \leq i_{\max}, \sum_n C_n^i L_{ni'} = \delta_{ii'}.
\end{aligned} \tag{6.23}$$

Once the optimal coefficients (C_n^i) have been determined, the optimal weights can be obtained by substitution into Eq. 6.20. We summarize these results in Proposition 5

Proposition 5. *Let $P_{\boldsymbol{\theta}}(n)$ be a statistical model for fully randomized benchmarking in terms of parameters $\boldsymbol{\theta}$ and let $\boldsymbol{\theta}^{(0)}$ be a reference point. Let $P_{\boldsymbol{\theta}}(n)$ be differentiable at $\boldsymbol{\theta}^{(0)}$ with respect to each parameter in $\boldsymbol{\theta}$ for each n. An experiment design for a fully randomized benchmarking experiment that minimizes the sum of the variances of the first i_{max} model parameters, to 'first-order' accuracy around the reference point, can be chosen by first performing the optimization in Eq. 6.23 to determine the coefficients C_n and then substituting those coefficients into Eq. 6.20 to determine the optimal weights w_n. The optimal weights w_n can be rounded to integer values to design a realistic experiment.*

6.4.1 Minimizing a weighted linear combination of parameter variances

The previous section describes how to optimize an experiment to minimize $\sum_i V_i$, where V_i is the variance of the linear estimator for the parameter θ_i. There are situations where we are interested in minimizing $\sum_i V_i Y_i^2$ where Y_i^2 is a list of positive weights that is fixed ahead of time. For example, if we care equally about the first i' parameters and not the rest, we can set $Y_i^2 = 1$ for $i \leq i'$ and $Y_i^2 = 0$ for $i > i'$. Also, we can compare to experiment optimizations for a single parameter i' by setting $Y_{i'}^2 = 1$ and all other $Y_i^2 = 0$.

To optimize the experiment design in this case, we have to make a small modification to the analysis of the previous section. For completeness, we include the full calculation here. The goal is to minimize the following

$$V = \sum_{i,n} C_{ni}^2 \frac{v_n}{w_n} Y_i^2, \tag{6.24}$$

where the minimization is over C_{ni} and w_n, subject to the constraints $\sum_n w_n t_n = T$ and $w_n \geq 0$, and the constraint that the linear estimators are 'for' their respective parameters, $\sum_n C_{ni} T_{nj} = \delta_{ij}$. Just as before, if the the family of linear estimators C_{ni} is fixed ahead of time, the optimization over the $\{w_n\}$ can be done in closed form by using Lagrange multipliers. After taking derivatives with respect to the $\{w_n\}$, we find that

$$w_n = \sqrt{\sum_{i''} C_{ni''}^2 Y_{i''}^2} \frac{\sqrt{v_n}}{\sqrt{t_n}} \frac{1}{\sqrt{\lambda}}, \tag{6.25}$$

where λ is the lagrange multiplier for the constraint $\sum_n w_n t_n = T$. We substitute into the constraint to solve for λ, and find

$$\sqrt{\lambda} = \frac{1}{T} \sum_{n'} \sqrt{\sum_{i'} C_{n'i'}^2 Y_{i'}^2} \sqrt{v_{n'} t_{n'}}. \tag{6.26}$$

Substitution into the objective function leads to

$$V = \sum_n \sqrt{\lambda} \sqrt{v_n t_n} \frac{\sum_i C_{ni}^2 Y_i^2}{\sqrt{\sum_{i''} C_{ni''}^2 Y_{i''}^2}} = \sum_n \sqrt{\lambda} \sqrt{v_n t_n} \sqrt{\sum_i C_{ni}^2 Y_i^2}. \tag{6.27}$$

Finally, substitution for $\sqrt{\lambda}$ gives

$$V = \frac{1}{T} \left(\sum_n \sqrt{v_n t_n} \sqrt{\sum_i C_{ni}^2 Y_i^2} \right)^2, \tag{6.28}$$

and this can be minimized by minimizing $\sum_n \sqrt{v_n t_n} \sqrt{\sum_i C_{ni}^2 Y_i^2}$. This can be cast as a second-order-cone problem by introducing the auxiliary variables D_n, which are intended to represent

$\sqrt{\sum_i C_{ni}^2 Y_i^2}$, and introducing the constraint $D_n \leq \sqrt{\sum_i C_{ni}^2 Y_i^2}$. The objective function remains $f = \sum_n D_n \sqrt{v_n t_n}$, and the linear constraints remain the same. In total, the resulting optimization is as follows

$$\begin{aligned} \text{Minimize: } & \sum_n D_n \sqrt{v_n t_n} \\ \text{Variables: } & (D_n)_{n=1}^{n_{\max}}, (C_{ni})_{n=1,i=1}^{n_{\max},i_{\max}} \\ \text{Subject to: } & \text{for all } n,\ D_n \geq 0, \\ & \text{for all } n,\ \sum_i (C_{ni})^2 Y_i^2 \leq D_n, \\ & \text{for } i \leq i_{\max}, \sum_n C_{ni} L_{ni'} = \delta_{ii'}. \end{aligned} \tag{6.29}$$

Once the optimal coefficients (C_n^i) have been determined, the optimal weights can be obtained by substitution into Eq. 6.26. We summarize these results in Proposition 6

Proposition 6. *Let $P_{\boldsymbol{\theta}}(n)$ be a statistical model for fully randomized benchmarking in terms of parameters $\boldsymbol{\theta}$ and let $\boldsymbol{\theta}^{(0)}$ be a reference point. Let $P_{\boldsymbol{\theta}}(n)$ be differentiable at $\boldsymbol{\theta}^{(0)}$ with respect to each parameter in $\boldsymbol{\theta}$ for each n. An experiment design for a fully randomized benchmarking experiment that minimizes the weighted sum of the variance of the model parameters θ_i, with positve weights Y_i^2, to 'first-order' accuracy around the reference point, can be chosen by first performing the optimization in Eq. 6.29 to determine the coefficients C_n and then substituting those coefficients into Eq. 6.26 to determine the optimal weights w_n. The optimal weights w_n can be rounded to integer values to design a realistic experiment.*

6.5 Interleaved randomized benchmarking

Interleaved randomized benchmarking (IRB)[50] is a common variant of randomized benchmarking that analyzes a specific Clifford gate G. In IRB the gate G is interleaved between steps of a two-, and under reasonable assumptions, the error channel of the gate G is effectively depolarized. The depolarizing parameter of this channel can be inferred by comparing the decay rates in success

probability in the case where G is interleaved to the case the case where G is left out and only the two-design steps are performed.

Here we discuss optimized experiment design for interleaved randomized benchmarking. At a high level, IRB can be analyzed with only small modifications to the procedures described in the previous sections. The only modifications to those procedures concern the list of available experiment design choices for the purposes of experiment design optimization, and the question of exactly what statistical model to use when analyzing the experiment. To address these issues we provide a complete discussion of experiment design optimization for interleaved randomized benchmarking. Specifically, we will consider two approaches to interleaved randomized benchmarking, which are addressed in the next two subsections.

6.5.1 Interpretation as two different two-designs

In the first approach we treat the experiments involving basic two-design steps as a standard randomized benchmarking setup, and we treat the experiments involving the interleaved Clifford gates as a separate randomized benchmarking setup where the implementation of the two-design consists of performing a basic two-design step followed by the interleaved gate. Applying any Clifford gate after a gate from a two-design ensemble results in a combined ensemble that is still a two-design. This justifies the interpretation of the combination of a basic two-design-step and the interleaved gate as the implementation of a slightly different two-design, which we refer to as the interleaved two-design. We still assume that the SPAM parameter s is the same for both two-designs. This leads to a statistical model with three parameters: the SPAM parameter s, the decay rate of the basic two-design r_b, and the decay rate of the interleaved two-design r_i. In this approach, we are interested in inferring $r_i - r_b$, which is the difference in decay rates associated to the two different two designs. This difference carries the interpretation of the 'quality' of the interleaved gate. We expect the difference in decay rates to approximately correspond to the fidelity of the interleaved gate, although there may be subtleties in this interpretation due to the cancellation of coherent errors in a gate-dependent error model.

The statistical model for this formulation of interleaved randomized benchmarking is as follows. We assume that the success probability for both two designs follows the basic model of randomized benchmarking. Denoting the success probability for the basic two-design by $P^b_{\boldsymbol{\theta}}(n_b)$ and the success probability for the interleaved two-design by $P^i_{\boldsymbol{\theta}}(n_i)$, we have

$$P^b_{\boldsymbol{\theta}}(n_b) = \frac{1}{D} + \frac{1}{\alpha}(1-\alpha s)(1-\alpha r_b)^n, \tag{6.30}$$

where D is the Hilbert space dimension, α is the constant $D/(D-1)$. Similarly, we have

$$P^i_{\boldsymbol{\theta}}(n_i) = \frac{1}{D} + \frac{1}{\alpha}(1-\alpha s)(1-\alpha r_i)^n. \tag{6.31}$$

In both of these equations, $\boldsymbol{\theta}$ denotes the list of statistical parameters, which in this case is $\{s, r_b, r_i\}$.

In the formulation of experiment design optimization for randomized benchmarking in the previous sections each trial was specified only by its sequence length, but here each trial must also specify either the basic two-design or the interleaved two-design. Designing the experiment now requires choosing the sequence lengths and trial numbers for both the basic and interleaved two-designs. As a result, the full list of possible experiment design choices can be obtained by concatenating the list of available choices for n_b to the list of available choices for n_i. Each of these choices has a time cost that is determined by the SPAM time cost t_s (assumed to be the same for both two-designs) and the time costs t_b, t_i for the basic and interleaved two-design steps respectively. The full statisical model in question consists of Eq. 6.30 and Eq. 6.31, and the full specification of an interleaved randomized benchmarking experiment consists of the number of trials to be run at each available choice of n_b and n_i. Now, the goal is to minimize the variance of inferring the quantity $r_i - r_b$. To accomplish this, it is convenient to relabel statistical parameters by introducing $r_- = r_i - r_b$ and $r_+ = r_i + r_b$. After this relabeling the statistical parameters are the list $\{s, r_-, r_+\}$, and we are interested in an optimal linear estimator that infers r_- while being insensitive to r_+ and s. The numerical procedure to perform the experiment design optimization now follows directly

from the previous section.

6.5.2 Interpretation involving two different depolarizing parameters

In the second approach to IRB, we assume that the error channels associated to the basic two-design step and to the interleaved gate are both depolarizing channels. Now, the specification of an IRB trial is determined by n_b and n_i, which are the number of basic two-design steps and the number of interleaved gates, respectively. In principle, the only restriction on n_b and n_i is that $n_i < n_b$, so that the interleaved gates can be properly 'interleaved' into the basic two-design steps. Furthermore, we require that the interleaved gates are evenly spaced throughout the sequence. These requirements help to enforce the assumption that the error channels of both the interleaved gate and the basic two-design are indeed twirled to depolarizing channels. In this approach to IRB we assume the experiment is well described by a statistical model involving the SPAM parameter s, the depolarizing parameter of the basic two-design step λ_b, and the depolarizing parameter of the interleaved gate λ_s, of the following form

$$P_{\boldsymbol{\theta}}(n_i, n_b) = \frac{1}{D} + \frac{1}{\alpha}(1 - \alpha s)(1 - \alpha\lambda_b)_b^n(1 - \alpha\lambda_i)_b^n. \tag{6.32}$$

The vector of statistical parameters $\boldsymbol{\theta}$ is $\{s, \lambda_b, \lambda_i\}$. The time cost of each step is determined by n_b and n_i based on the time cost of implementing the basic two-design steps and the interleaved gate respectively. The parameter of interest is the depolarizing paramter of the interleaved gate λ_i, and the numerical procedure to perform the experiment design optimization follows directly from Section 6.1 In traditional interleaved randomized benchmarking experiments, there are either zero interleaved gates in a trial, or the number of basic two-design steps and the number of interleaved gates in a trial are essentially the same. These choices correspond respectively to either having no interleaving or 'fully interleaved' scenarios. One interesting possiblity for designing experiments for this statistical model is that in principle it may be beneficial to have 'partially interleaved' experiments, where the interleaved gate is only performed some intermediate fraction of the time.

To numerically address this possiblity, we perform the design optimization for a specific scenario. Concretely, we use the statistical model in Eq. 6.32 with a reference point SPAM error of 5×10^{-2}, a basic step error of 1×10^{-4} and an interleaved gate error of 2×10^{-4}, a SPAM time cost of 1×10^{-3} s, a basic step time cost of 1×10^{-3} s, and an interleaved gate time cost of 3×10^{-4} s. We run the optimization to minimize the variance of inferring the interleaved gate error with a total available time of 4×10^{3} s. We set the minimum number of basic two-design steps to 5 and the maximum to 2×10^{4}, and we require that the number is divisible by 5. For each possible number of basic steps in the sequence, we allow corresponding fractions of interleaved gates in the list $\{\frac{0}{5}, \frac{1}{5}, \frac{2}{5}, \frac{3}{5}, \frac{4}{5}, \frac{5}{5}\}$. When we run the optimization, we find that the optimal experiment design consists of 10898 trials with 5 basic steps and 0 interleaved gates, 4145 trials with 485 basic steps and 485 interleaved steps, and 1149 trials with 1145 basic steps and 0 interleaved steps. For this specific example, we observe that the optimized experiment design does not have any fraction of interleaved gates other than 0 or 1. In general we have not found any scenario where the optimized design contains any other fractions of interleaved gates. An interesting direction of future work would be to investigate this question further and attempt to prove that it is never optimal to include fractions of interleavd gates other than 0 or 1.

Chapter 7

Statistical Analysis

This chapter is substantially based on work published in Ref. [45].

7.1 Maximum likelihood methods

For inference of the model parameters $\boldsymbol{\theta}$ for any model $P_{\boldsymbol{\theta}}(n)$, we use maximum likelihood whenever it is tractable and well-behaved, which it is for all the examples we consider in this paper. Maximum likelihood has the advantage that it is asymptotically unbiased, meaning that as the amount of collected data grows to infinity, the inferred model parameters match the true point in parameter space. To obtain confidence intervals on one or more parameters one may use statistical bootstrapping, which is a method of resampling the observed data to learn how much the inferred quantities vary as the data varies. For more information about maximum likelihood and statistical bootstrapping we refer to Refs. [5, 19]. Here we provide the log-likelihood function for an arbitrary model and discuss the possibilities for obtaining confidence intervals through statistical bootstrapping. We also discuss a statistical analysis to possibly reject the inner model(s) of a set of nested models using an empirical likelihood ratio test with statistical bootstrapping.

The log-likelihood function for an arbitrary model is as follows. The probability L_j of observing c_j successes out of w_j trials at the sequence length n_j is

$$L_j = \binom{w_j}{c_j} (P_{\boldsymbol{\theta}}(n_j))^{c_j} (1 - P_{\boldsymbol{\theta}}(n_j))^{w_j - c_j}. \tag{7.1}$$

The total probability is obtained by taking a product over all sequence lengths in the list (n_j). It follows that the full log-probability Θ is

$$\Theta = \sum_{j=1}^{j_{\max}} \left(\log \binom{w_j}{c_j} + c_j \log P_{\boldsymbol{\theta}}(n_j) + (w_j - c_j) \log \left(1 - P_{\boldsymbol{\theta}}(n_j)\right) \right). \tag{7.2}$$

We note that the dependence on the model parameters $\boldsymbol{\theta}$ is entirely through $P_{\boldsymbol{\theta}}(n)$. For the analysis of randomized benchmarking performed in this thesis, in order to numerically maximize Θ we choose several initial values for $\boldsymbol{\theta}$, use standard gradient ascent methods to maximize Θ for each choice of the initial value, and use the point that maximizes Θ over all of those choices.

Parametric or non-parametric bias-corrected bootstrapping [18, 19] can be used to obtain confidence intervals for one or more parameters. In typical uses of bootstrapping in quantum characterization, the bootstrap assumptions are not satisfied, often because the parameters are statistically close to the boundary. As a result, the coverage probabilities do not necessarily closely match the nominal confidence levels used. A coverage probability is the probability that, for data drawn from a statistical model at a particular reference point, a confidence interval obtained by bootstrapping will contain the reference point. If the coverage probability matches the nominal confidence levels used, then the confidence interval is interpreted as 'accurate'. Even if a confidence interval obtained by bootstrapping is not mathematically guaranteed to be 'accurate', at moderate confidence levels, the confidence intervals obtained are useful for interpretation but should be treated as approximate. For more information about potential issues with bootstrap coverage probabilities we refer to [11, 64] and for examples and discussion in the context of quantum information science we refer to [3, 4]. When we optimize the design of an experiment according to the procedure in Section 6.1, the anticipated uncertainty that we minimize is intended to approximate the size of confidence intervals obtained according to the Gaussian assumption, absent any boundary issues. However, for experiments of finite duration the confidence intervals in general do not exactly match the anticipated uncertainty, even if the reference point used for the optimization is equal to the true point. In the limit that the experiment duration and the amount of data become large and a

Gaussian model is a good approximation, the confidence interval sizes should match the anticipated uncertainty. There can still be deviations in the large data limit if the reference point used for the optimization does not match the true point.

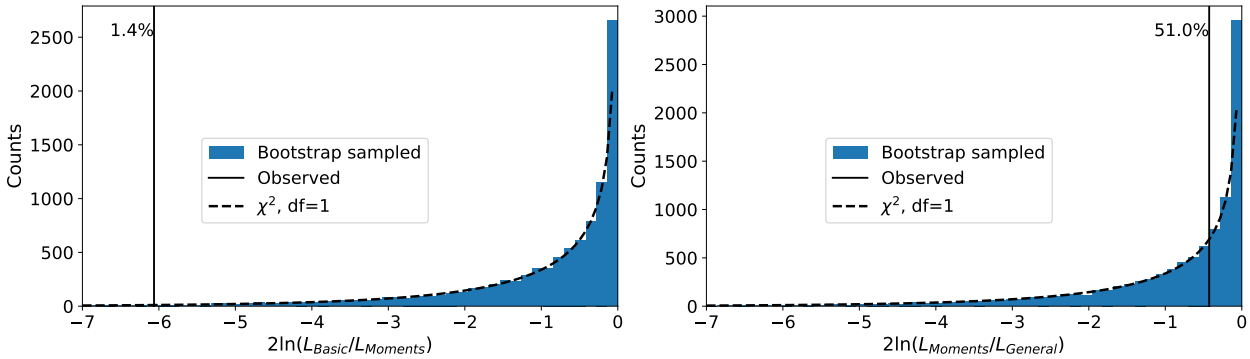
In some experiments there may be two or more relevant statistical models that are nested, meaning that the inner model can be obtained from the outer model by fixing some of its parameters at constant values. In such a situation, it may be useful to perform a statistical analysis to attempt to reject the inner model. One such method is to use an empirical likelihood ratio test with statistical bootstrapping [5, 31, 62]. A standard likelihood ratio test with a chi-squared analysis would be sufficient if a Gaussian model were accurate. However, in many relevant cases the Gaussian model does not hold and this can lead to noticeable statistical issues [65]. For this reason, one may use an empirical likelihood ratio test which we now describe. We denote the outer model by $P_{\boldsymbol{\theta},\boldsymbol{\phi}}(n)$, where now there are two sets of statistical parameters $\boldsymbol{\theta}, \boldsymbol{\phi}$. The inner model is obtained by setting $\boldsymbol{\phi}$ to some particular value. For a given set of data, a maximum likelihood analysis can be run for both models, and the ratio of the maximum likelihood values can be computed. Assuming the inner model is true, the distribution of likelihood ratios can be estimated, empirically, by bootstrap resampling the data according to the inner model and computing the likelihood ratio for each resampled dataset. With this analysis one can reject the inner model at a particular confidence level, which is based on the percentile of the observed likelihood ratio within this empirical distribution of bootstrapped likelihood ratios.

As a concrete example, we conduct a simulated empirical likelihood ratio test to check for deviations from the basic model of fully randomized benchmarking. We consider a model where the SPAM error is fixed at $\theta_0 = 3 \times 10^{-2}$ and for each trial the step error θ_1 is drawn independently from a Gaussian distribution with mean 1×10^{-4} and standard deviation 2.5×10^{-5}. To choose an experiment design for the simulated experiment, we use the four-parameter moments model and perform the optimization described in Section 6.1. For this optimization we choose the reference point to match the moments of the actual Gaussian distribution of the step error, and we minimize the standard deviation of the parameter θ_2 as a proxy for maximizing the statistical power to reject

the basic model. We choose θ_2 as a proxy because θ_3 is zero for the chosen distribution of step errors. In the optimized experiment, the anticipated standard deviation of the step error θ_1 is 1.1×10^{-6}. If the experiment were instead optimized to minimize the standard deviation of θ_1, the optimal experiment in that case would have an anticipated standard deviation of 8.0×10^{-7}. This illustrates the fact that the decision to optimize the experiment to maximize statistical power to reject the basic model has a relatively small effect on the performance of inferring the step error. The optimized experiment is constrained so that the total run time is 3 hours, assuming that each step takes 10^{-5} s and state preparation and measurement takes 10^{-3} s. Once the experiment design has been chosen, we simulate a dataset for this experiment by drawing a step error θ_1 independently for each trial. Once a value of θ_1 has been drawn, we then draw 'success' or 'failure' with the corresponding probability obtained from the basic model for the drawn value of θ_1 and the particular sequence length in question. With the simulated dataset we then perform a bootstrapped empirical likelihood ratio test where the inner model is the basic model and the outer model is the three-parameter moments model. We also perform the same bootstrapped empirical likelihood ratio test where the inner model is the three-parameter moments model and the outer model is the general model. For each choice of inner model and outer model we follow the procedure outlined in the previous paragraph, and the results are shown in Fig. 7.1. According to the distribution of bootstrapped likelihood ratios, we can reject the basic model relative to the three-parameter moments model at a p-value of 1.4%. No significant deviation from the three-parameter moments model relative to the general model was detected (p-value of 51.0%). These results agree with the intuition that the Gaussian fluctuation in the step error is detectable via the second moment, and that the fourth and higher moments can be safely neglected in this scenario.

7.2 Analysis of achievable uncertainty improvements

We illustrate the advantages of full randomization by comparing the uncertainties achieved in several published experiments to the uncertainties that could have been achieved with fully randomized benchmarking with the same experiment design. We also demonstrate that additional



(a) Basic model vs Moments model

(b) Moments model vs General model

Figure 7.1: Results of simulated empirical likelihood ratio tests of three nested statistical models of fully randomized benchmarking. The three statistical models that we consider are the basic model, the three-parameter moments model, and the general model. Further details about these models are in Section 5. The test is conducted using a simulated dataset obtained by drawing the step error θ_1 independently for each trial from a Gaussian distribution with mean 1×10^{-4} and standard deviation 0.25×10^{-4}. To perform the test we use the procedure described in Section 7.1 in two cases, the first where the inner model is the basic model and the outer model is the three-parameter moments model, and the second where the inner model is the three-parameter moments model and the outer model is the general model. In the first case we can reject basic model with a p-value of 1.4%, and in the second case the p-value to reject the three-parameter moments model is 51.0%. The χ-squared fits show good agreement with the χ-squared distribution with one degree of freedom, which indicates that the statistical power of this experiment is sufficiently large that a Gaussian model of the statistical fluctuations is reasonably accurate.

reductions in uncertainty could have been achieved by optimizing the experiment design according to the procedure in Section 6.1. The published randomized-benchmarking experiments that we use for specific comparisons are Refs. [7, 41, 42]. For each past experiment we assume that the basic model in Eq. 3.16 is accurate and we use the procedure in Section 6.2 to construct the optimal linear estimator for step error according to the reported sequence lengths and reported total number of trials at each sequence length, and using the reported step error and SPAM error as the reference point. In all of these experiments the same random sequences were repeated many times, but our construction of the optimal linear estimator assumes that the experiment was fully randomized and that a new random sequence was drawn for each trial. Therefore, we interpret the standard deviation of the optimal linear estimator as the anticipated uncertainty if the experiment had been fully randomized, and we compare it to the uncertainty actually reported by each experiment. Then, we run the optimization described in Section 6.1 to construct the optimal experiment design according to the basic model. For Refs. [41, 42] we assume that the step time is equal to the SPAM time and for Ref [7] we assume that the step time is 100 times smaller than the SPAM time. We choose these ratios to roughly approximate the ratios of step time to SPAM time for the respective computational platforms of these experiments. We interpret the anticipated uncertainty returned by the optimization as the size of the confidence interval for each experiment if it had been fully randomized and the optimal experiment design had been used. Finally, we repeat the optimization for the four-parameter moments model to see how the anticipated uncertainty is affected by a more general model. All of these observations are recorded in Table. 7.1. We generally observe that reductions in uncertainty are possible both from fully randomizing the experiment and from using the optimal experiment design.

In the case of Ref. [41], we observe more than a factor of four improvement in anticipated uncertainty if the experiment is fully randomized. As we show in Section 7.3, the size of the improvement in uncertainty from fully randomizing depends on the true error model, and is larger if the true errors are closer to unitary errors. For the parameters reported in Ref. [41] we find that an improvement of this size from fully randomizing is possible if the true errors are unitary. Further

Table 7.1: Results of a numerical signal-to-noise comparison between past randomized benchmarking experiments and experiments optimized according to the procedure in Section 6.1. The columns show the referenced benchmarking experiment; the gate error and uncertainty reported by each experiment; the anticipated uncertainty for a fully randomized experiment with the reported experiment design; obtained as in Appendix 6.2; and the expected uncertainty if the experiment design is optimized, as described in Section 6.1, for the basic model and the four-parameter moments model respectively.

Experiment	Reported step error	Reported uncertainty	Fully randomized anticipated uncertainty (basic model)	Optimized anticipated uncertainty (basic model)	Optimized anticipated uncertainty (moments model)
Ref. [7]	2.0×10^{-5}	2×10^{-6}	2.1×10^{-6}	1.0×10^{-6}	1.6×10^{-6}
Ref. [42]	8.3×10^{-3}	2×10^{-4}	1.2×10^{-4}	1.1×10^{-4}	1.7×10^{-4}
Ref. [41]	5.3×10^{-2}	4×10^{-3}	8.8×10^{-4}	4.3×10^{-4}	1.3×10^{-3}

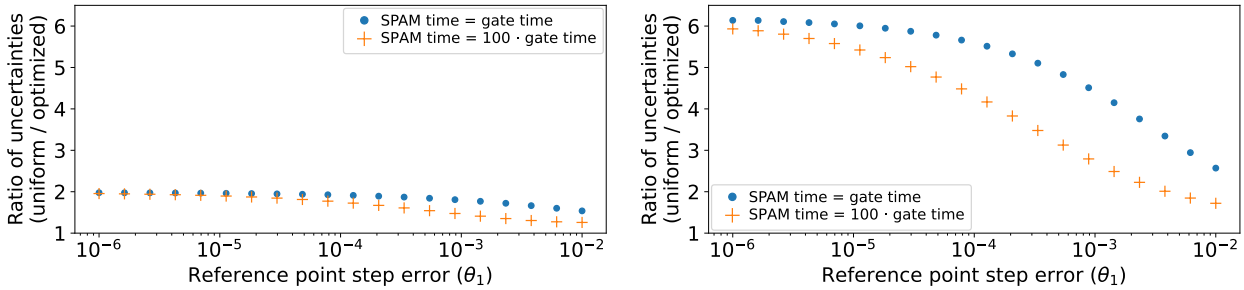
details of this comparison are in Section 7.3.

In addition, we numerically explore the improvement obtained by optimizing the experiment design for hypothetical fully randomized experiments. The comparisons are made between uniform experiment designs where the sequence lengths are chosen uniformly in a fixed range and the same number of trials are performed at each sequence length, and optimized experiment designs constructed according to the method in Section 6.1. The optimized experiment designs are constrained to take the same total time as the corresponding uniform experiments. To compare uniform experiment designs to optimized designs, we compute the standard deviations of the optimal linear estimator in the linearized model, as described in Section 6.1 and Section 6.2, and take the ratio of these standard deviations. Larger ratios indicate a larger benefit from optimizing and the square of this ratio roughly corresponds to the ratio of experiment times required to achieve the same standard deviation because the standard deviation decreases with the square root of the number of measurements. The results of these comparisons are shown in Fig. 7.2. In plot (a) we use the basic model with the spam error parameter θ_0 set to 10^{-2} and step error $\theta_1 \in [10^{-6}, 10^{-2}]$. In plot (b) we use the moments model with four total parameters, where the reference values of the moments parameters are set to zero, θ_0 is set to 10^{-2}, and θ_1 ranges over $[10^{-6}, 10^{-2}]$. In both plots the uniform experiment design

consists of 20 uniformly spaced sequence lengths in the range $[1, 1/\theta_1]$. The ratio of the SPAM time to the step time is either set to 1 or 100 and both options are shown in the plots. At a step error of 10^{-6} and when the ratio of the SPAM time to the step time is 100, we observe a reduction in standard deviation by a factor of 1.96 for the basic model and by a factor of 5.9 for the moments model with four parameters. These improvements correspond to time savings by factors of 3.8 and 35.2 respectively.

7.3 Excess variance from not fully randomizing

For randomized benchmarking with a fixed set of sequence lengths and a fixed number of trials at each sequence length, fully randomized benchmarking generally yields lower uncertainty than randomized benchmarking with multiple trials for each chosen sequence. The uncertainty reduction depends on the error channels and is due to the fact that, for a given sequence length, the true success probability in general depends on the particular random sequence that is used for a trial. The reduction is particularly pronounced for unitary error channels and may be analyzed by fixing the sequence length n and evaluating the variance of the empirical estimate of $P(n)$ for the general scenario where we run $M = kl$ independent trials consisting of k randomly chosen sequences where each sequence is run l times. For related work on the relationship between the number of random sequences and the variance of a randomized benchmarking experiment, we refer to Refs. [34, 70]. The fully randomized scenario has $k = M$ and $l = 1$. We assume a sequence-dependent probability of success s. Since the sequence is chosen randomly, the probability of success can be considered as a random variable with probability measure on $s \in [0, 1]$ given by $\mu(s)$ that depends on n and the two-design used. The goal is to estimate the average probability of success, which is given by $\bar{s} = \langle s \rangle_\mu = \int d\mu(s) s$. For $i = 1, \ldots, k$, let $\hat{c}_i$ be the number of observed successes for the i'th sequence. The minimum variance estimator for $\bar{s}$ is the empirical average $\hat{s} = \frac{1}{k} \sum_i \frac{\hat{c}_i}{l}$. Because the sequences are independent and identically distributed, each $\hat{c}_i$ is identically distributed according to a random variable C which is the sum of l Bernoulli random variables with success probability S. The variance of C given S is $lS(1 - S)$ and the mean of C given S is lS. The variance of C can be



(a) **(b)**

Figure 7.2: Comparison between hypothetical fully randomized experiments that use either an optimized experiment design or a uniform design of evenly-weighted sequence lengths. The plots show the ratio of the anticipated standard deviations of the step error (uniform/optimized), as a function of the reference step error. In all cases, the SPAM parameter θ_0 is fixed at 10^{-2} and the total experiment time set to a constant. For the uniform experiment design, 20 evenly-spaced sequence lengths from $[1, 1/\theta_1]$ are used and the number of trials at each length is the same. In both plots, the dots show the comparison assuming that the SPAM time is equal to the step time and the plus signs show the comparison assuming that the SPAM time is larger than the step time by a factor of 100. Larger ratios indicate a larger benefit from optimizing the experiment design. In plot (a) we use the basic model and in plot (b) we use the moments model with four total parameters.

computed according to the law of total variance [72] as

$$\begin{aligned}\mathrm{var}(C) &= E(\mathrm{var}(C|S)) + \mathrm{var}(E(C|S)) \\ &= l\langle S(1-S)\rangle_\mu + l^2\langle (S-\bar{s})^2\rangle \quad = \quad l\bar{s}(1-\bar{s}) + l(l-1)\mathrm{var}(S). \end{aligned} \tag{7.3}$$

This expression appears, for example, in Appendix A of Ref. [34]. The variance of $\hat{s}$ is $\frac{1}{k}\mathrm{var}(C)/l^2$. Accordingly,

$$\mathrm{var}\hat{s} = \frac{\bar{s}(1-\bar{s})}{M} + \frac{(l-1)}{M}\left(\left(\int d\mu(s)s^2\right) - \bar{s}^2\right). \tag{7.4}$$

The second term in Eq. 7.4 vanishes if $l = 1$, so we can interpret it as the excess variance due to not fully randomizing and we denote it by Z. An important point is that Z depends on the exact error model via $\int d\mu(s)s^2$. For example, if all errors are depolarizing channels then the success probability is independent of the random sequence and $\int d\mu(s)s^2 = \bar{s}^2$, which implies that $Z = 0$. In contrast, if the error channel is a fixed unitary, S depends on the particular sequence and Z may be significant. In this regard the observed variance in success probabilities over random sequences can provide a measure of the amount of coherent error in a randomized benchmarking experiment. This has been observed qualitatively in Ref [44], where the large variance in fidelities at each sequence length is attributed to coherent errors. For related work to distinguish coherent and incoherent errors in a randomized benchmarking experiment by inferring a quantity called the unitarity, we refer to Refs. [15, 68].

To better understand the size of the excess variance Z, we consider a specific error model with unitary error channels. Consider an error model where the final state ψ is assumed to be equal to the target state χ with probability λ and is a random pure state with probability $1-\lambda$. To express λ in terms of $\bar{s}$, we note that

$$\bar{s} = \lambda + (1-\lambda)\int d\psi_H f_\psi = \lambda + \frac{1}{D}(1-\lambda), \tag{7.5}$$

where $d\psi_H$ denotes the Haar measure over pure states and f_ψ denotes the success probability for

each random pure state ψ. This relationship can be inverted to solve for λ as a function of $\bar{s}$

$$\lambda = \frac{\bar{s} - 1/D}{1 - 1/D}. \tag{7.6}$$

In order to determine the variance of $\hat{s}$ by substituting into Eq. 7.4 for this error model we first evaluate

$$\int d\mu(s) s^2 = \lambda + (1 - \lambda)\frac{2}{D(D+1)}, \tag{7.7}$$

where we have used the fact that $\int d\psi_H f_\psi^2 = \frac{2}{D(D+1)}$ (Section 7.4). Substitution for λ according to Eq. 7.6 leads to the final expression

$$\mathrm{var}\hat{s} = \frac{\bar{s}(1-\bar{s})}{M} + \frac{l-1}{M}\left(\frac{\bar{s} - 1/D}{1 - 1/D} + \frac{1-\bar{s}}{1-1/D}\frac{2}{D(D+1)} - \bar{s}^2\right). \tag{7.8}$$

For any given experiment, it is possible to estimate the excess variance due to repetition of sequences by considering the statistics obtained at a particular sequence length. For example, consider a sequence length of 20 in the experiment reported in Ref. [41]. In this experiment, $D = 4$, $k = 51$ and $l = 125$. At a sequence length of 20, the reported success probability is 0.31 and the 95 % confidence interval has a total size of approximately 0.06 as determined from Fig. 4a of Ref. [41]. If the experiment had been fully randomized, we would expect a total size of this confidence interval of 0.023 when analyzed according to the basic model. For comparison, with the unitary error model of the previous paragraph and the parameters reported in Ref. [41], the 95 % confidence interval would have had a total size of 0.16. It is therefore possible that the increased size of the reported confidence interval relative to the anticipated confidence interval from fully randomized benchmarking can be explained by coherent errors in the actual experiment.

7.4 Details of variance analysis of fully randomized benchmarking

Here we verify the fact that

$$\int d\psi_H f_\psi^2 = \frac{2}{D(D+1)}, \tag{7.9}$$

where $d\psi_H$ denotes the Haar measure over pure states, and f_ψ is the fidelity of the random pure state $|\psi\rangle$ with the target state $|\chi\rangle$. The Haar-random pure state $|\psi\rangle$ can be expressed as $U|\chi\rangle$ for a Haar-random unitary U, so this integral can be written as

$$\int d\psi_H f_\psi^2 = \int dU_H \mathrm{tr}\left[(U\otimes U)|\chi\rangle\langle\chi|^{\otimes 2}(U^\dagger\otimes U^\dagger)|\chi\rangle\langle\chi|^{\otimes 2}\right]. \tag{7.10}$$

In the notation of Lemma 3.5 of Ref. [17], we can express this as

$$\int d\psi_H f_\psi^2 = \mathrm{tr}\left[E(M)M\right], \tag{7.11}$$

where $M = |\chi\rangle\langle\chi|^{\otimes 2}$ and $E(M)$ is defined to be

$$E(M) = \int dU_H (U\otimes U)M(U^\dagger\otimes U^\dagger). \tag{7.12}$$

According to Prop. 2.2 in Ref. [12] and Lemma 3.5 in Ref. [17], it follows from Schur-Weyl duality that

$$E(M) = \alpha\mathbb{1} + \beta F, \tag{7.13}$$

where F is the swap operator and the coefficients α, β satisfy $\alpha D^2 + \beta D = \mathrm{tr}\left[M\right]$ and $\alpha D + \beta D^2 = \mathrm{tr}\left[MF\right]$. Here we have $M = |\chi\rangle\langle\chi|^{\otimes 2}$ so $\mathrm{tr}\left[M\right] = \mathrm{tr}\left[MF\right] = 1$ and it follows that $\alpha = \beta = \frac{1}{D(D+1)}$. Consequently, $\mathrm{tr}\left[E(M)M\right] = \frac{2}{D(D+1)}$.

Chapter 8

Fully randomized benchmarking experiment at NIST

This chapter is substantially based on work published in Ref. [45]. To provide a concrete comparison between non-fully-randomized benchmarking and optimized fully randomized benchmarking, we designed and implemented three randomized benchmarking experiments. To realize these experiments we perform single qubit rotations on a $^{25}Mg^{+}$ ion in a microfabricated surface-electrode ion trap, in the apparatus described in Refs. [8, 67]. We use the states $|F = 3, m_F = 1\rangle$ (logical $|1\rangle$) and $|F = 2, m_F = 1\rangle$ (logical $|0\rangle$) in the $^2S_{1/2}$ ground-state hyperfine manifold to realize a qubit. The qubit transition frequency of $\omega = 2\pi \times 1686$ MHz is first-order insensitive to the magnetic field at $B \approx 213$ G, mitigating against errors caused by fluctuations in the total magnetic field. Qubit rotations around X and Y are implemented with microwave magnetic fields applied at the transition frequency with differing phase, while Z rotations are implemented by adding a phase offset to the microwave control signal for subsequent rotations. The qubit is prepared with optical pumping followed by microwave pulses to transfer population to the $|1\rangle$ state. Qubit readout is accomplished by applying a laser resonant with the $^2S_{1/2}$ to $^2P_{3/2}$ cycling transition and detecting state-dependent ion fluorescence as described in the supplementary material of Ref. [67]. Full randomization is achieved by choosing gates in real time with a pseudorandom number generator (PRNG) [56] running on the same FPGA (field programmable gate array) that is used to generate the gate pulses applied to the ion. The ideal stabilizer (assuming no errors) is stored and concurrently updated on the FPGA as new gates are chosen, such that when the required number of random gates have been applied, the stabilizer can be used to return the qubit to the measurement basis and indicate

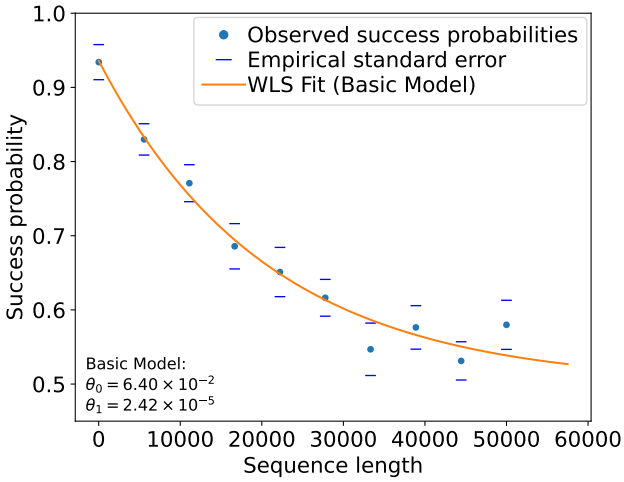
the expected measurement outcome. The on-the-fly calculation process for sequences can also be configured to enable intentional repetition of random gate sequences.

The three experiments in the comparison are as follows. First, we constructed an experiment where 10 sequence lengths were set uniformly in the range from 5 to $1/x_0$ where $x_0 = 2 \times 10^{-5}$ is the best guess for the step error prior to the experiment. At each sequence length we drew 24 random sequences and repeated each of them 24 times. This experiment took roughly 53.5 minutes of total time. Then, we repeated the same experiment but fully randomized the sequences, so at each sequence length a total of $24 \times 24 = 576$ random sequences were drawn and run once. The time to run the experiment is unaffected by fully randomizing, so this experiment also took 53.5 minutes of total time. Finally, we designed an optimized, fully randomized experiment using the methods in Section 6.1. The reference point for the optimization has a SPAM parameter of 3×10^{-2} and a step error parameter of 2×10^{-5} the value zero for all higher moments parameters, and the optimization minimizes the standard deviation of the step error according to the four-parameter moments model. This reference point was chosen based on our best guess for the parameters before we ran the experiment and was informed by prior calibration data. The total time of the optimized experiment was constrained to match the total time of the non-optimized experiments. For experimental simplicity we rounded the number of trials at each sequence length to a multiple of four, so that each experiment could be divided into four equal blocks. Within the first block, the order of experimental trials is randomly chosen and then the same order of trials is repeated for the remaining three blocks. Rounding the number of trials at each sequence length to a multiple of four had a negligible effect on the wall-clock time and anticipated standard deviations.

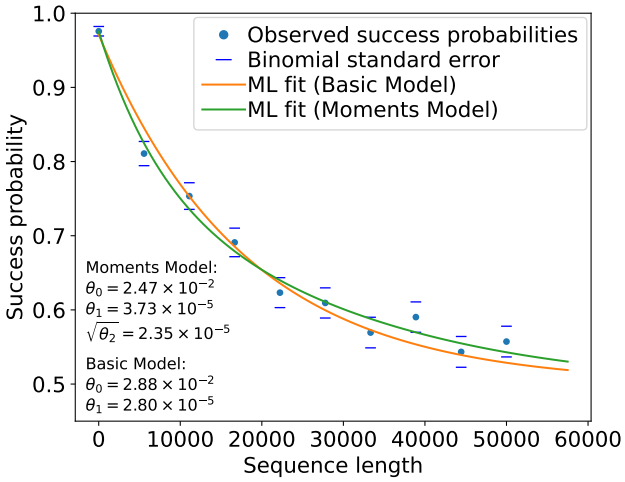
To analyze the randomized benchmarking experiment with repeated sequences, we ran a weighted least squares fit to the basic model. The weights in the fit are the squared inverses of the empirical standard errors of the success probabilities at each sequence length. The empirical standard errors are obtained by computing the empirical standard deviation of the estimated success probabilities of the random sequences and dividing by the square root of the number of sequence repetitions. For further information about weighted least square fits in randomized benchmarking we

refer to Ref. [51]. To analyze the fully randomized experiments we perform the maximum likelihood inference that we outline in Section 7.1. We perform this maximum likelihood analysis for both the basic model and the moments model with three total parameters. The results for all three experiments are shown in Fig. 8.1. To obtain confidence intervals on the step error for the various experiments, we perform bias-corrected parametric bootstrapping with $10,000$ bootstrap samples, as described in Section 7.1 and in Ref. [19]. For the first experiment, which has intentionally repeated sequences, the bootstrap samples are obtained by following the procedure in Ref. [51]. To summarize, first we resample the list of sequences with replacement, and then for each sequence we binomially resample the success and failure counts. Then, for each bootstrapped dataset the step error is estimated with a weighted least squares fit to the model. For the second and third experiments, which are fully randomized, the bootstrap samples are obtained by parametrically resampling according to the parameters obtained from the maximum likelihood analysis on the original data. The bootstrap histograms, point estimates, and 68% bootstrapped confidence intervals are shown in Fig. 8.2, where we run the analysis according to both the basic model and the three-parameter moments model. For the uniform design with repeated sequences, we report a step error of $2.42^{+0.30}_{-0.22} \times 10^{-5}$ when analyzing according to the basic model. For the optimized fully randomized experiment we report a corresponding step error of $2.57^{+0.07}_{-0.06} \times 10^{-5}$, which has a confidence interval that is roughly four times smaller. To test the basic model of the optimized, fully randomized experiment, we performed the empirical likelihood ratio test described in Section 7. The results are shown in Fig. 8.3. We found a p-value of 6.0% to reject the basic model. This shows weak evidence of deviation from an exponential decay, which we interpret as evidence of non-Markovian or time-dependent behavior.

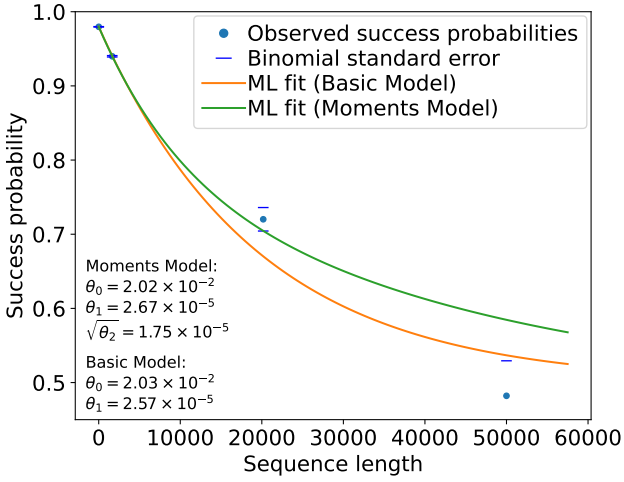
After completing these three experiments, we intentionally introduced a unitary error by miscalibrating the gates in the 2-design and repeated the same comparison between non-fully-randomized benchmarking and optimized fully randomized benchmarking. The size of the miscalibration was chosen to give a step error of approximately 5×10^{-4}. We followed the same procedure that we used previously to construct three randomized benchmarking experiments. For the first experiment we chose 10 sequence lengths uniformly in the range $[5, 2000]$, where the maximum sequence length



(a) Uniform design, repeated sequences



(b) Uniform design, fully randomized



(c) Optimized design, fully randomized

Figure 8.1: The observed decays in success probability for each of the three randomized benchmarking experiments run at NIST comparing non-fully randomized benchmarking to optimized fully randomized benchmarking. The experiment in plot (a) has sequence lengths chosen uniformly in the range $[5, 5 \times 10^4]$ and for each sequence length 24 random sequences are drawn and run 24 times each. The orange trace is the best fit to the basic model, obtained by a weighted least squares fit to the observed success probabilities, and the best fit parameters are shown inset in the lower left. The weights in the fit are the squared inverses of empirical standard errors of the observed success probabilities at each sequence length. These empirical standard errors are shown with the blue tickmarks. The experiment in plot (b) has the same sequence lengths as the first experiment, but is fully randomized so at each sequence length 24×24 random sequences are drawn and run once each. The experiment in plot (c) is designed according to the optimization routine in Section 6.1, where the total experiment time is constrained to match the total experiment time of each of the previous two experiments. In plots (b,c) the orange and green traces are the maximum likelihood fits to the basic model and the three-parameter moments model respectively, and the maximum likelihood parameters are shown inset in the lower left. The blue ticks show the binomial standard errors of the observed success probabilities.

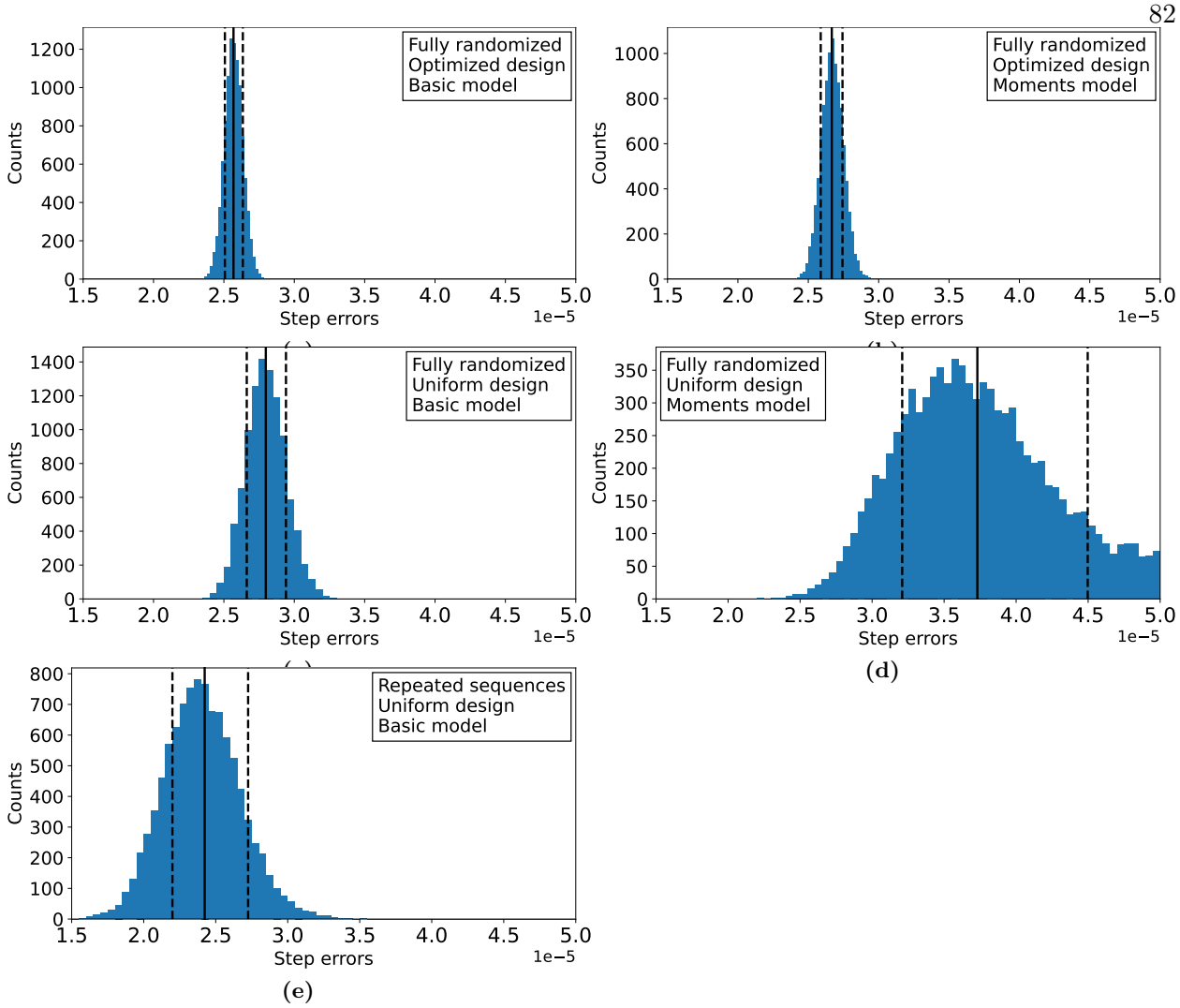


Figure 8.2: The bootstrap distributions obtained during the analysis of the three experiments run at NIST during our comparison between non-fully-randomized benchmarking and optimized fully randomized benchmarking. The plots in the left column correspond to the analysis according to the basic model and the plots in the right column correspond to the analysis according to the three-parameter moments model. The plots in the first row are for the optimized, fully randomized experiment, the plots in the second row are for the uniform, fully randomized experiment, and the plot in the third row is for the uniform experiment with repeated sequences. We do not include the plot for the uniform experiment with repeated sequences analyzed according to the moments model because performing a weighted least squares fit to the moments model is not a standard technique in randomized benchmarking. For all the plots, the solid black line indicates the step error parameter of the best fit to the original data and the dashed black lines denote the 68% confidence interval obtained with bias-corrected bootstrapping. The best fit step errors for the plots are (a) $2.57^{+0.07}_{-0.06} \times 10^{-5}$, (b) $2.67^{+0.08}_{-0.08} \times 10^{-5}$, (c) $2.80^{+0.14}_{-0.14} \times 10^{-5}$, (d) $3.73^{+0.77}_{-0.52} \times 10^{-5}$, (e) $2.42^{+0.30}_{-0.22} \times 10^{-5}$.

again corresponds to $1/x_0$. At each sequence length we drew 100 random sequences and repeated

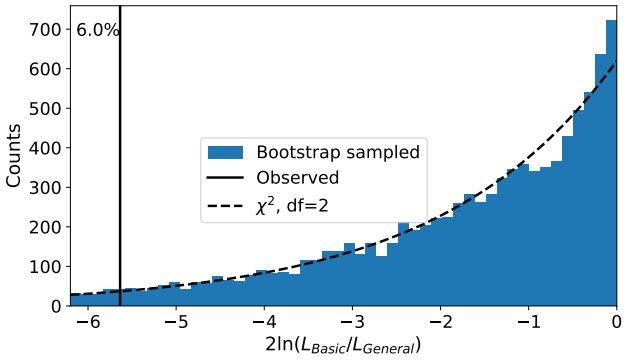
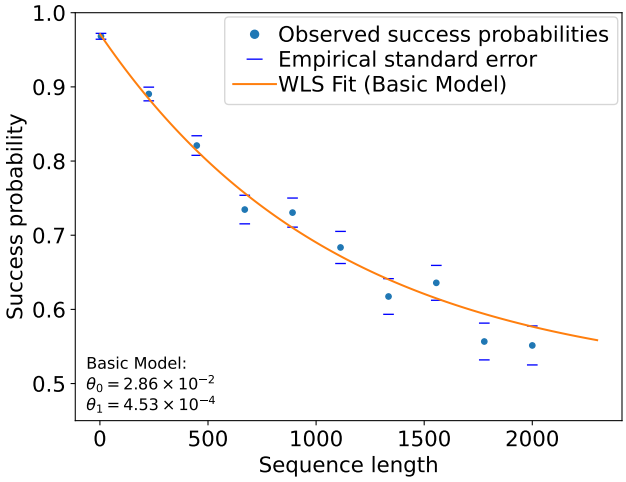
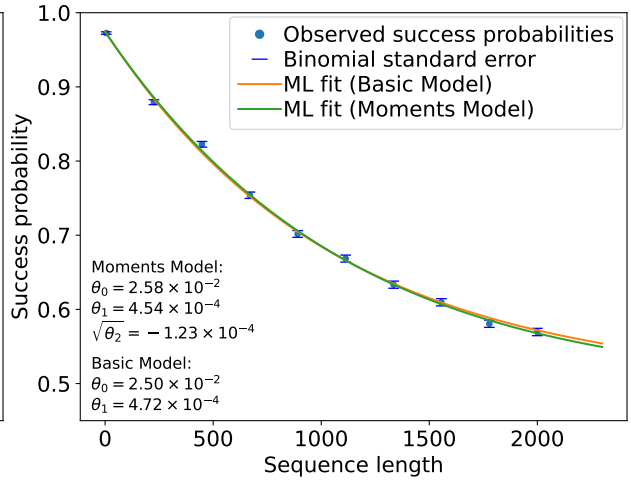


Figure 8.3: Results of an empirical likelihood ratio test for the optimized, fully randomized experiment. We perform the procedure described in Section 7.1, with the basic model as the inner model and the general model as the outer model. Further details about these models are in Ch. 7. We find a p-value of 6.0% to reject the basic model. This shows weak evidence of deviation from an exponential decay, which we interpret as evidence of non-Markovian or time-dependent behavior.

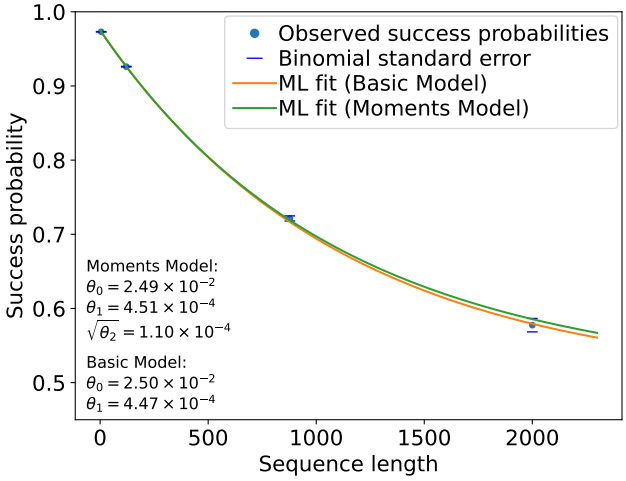
each of them 100 times. This experiment took roughly 40 minutes of total time. Second, we repeated the same experiment but fully randomized the sequences so at each sequence length 100×100 random sequences were drawn and run once. Third, we performed an optimized fully randomized experiment that took the same wall-clock time, where the optimization was again done to maximize statistical power to infer the step error using the four-parameter moments model. The results of this analysis are reported in Fig. 8.4, and the bootstrap distributions are reported in Fig. 8.5. We again observe a confidence interval for the optimized fully randomized experiment that is roughly 4 times smaller than that of the uniform experiment with repeated sequences. We also run the same empirical likelihood ratio test between the basic model and the general model. The results are shown in Fig. 8.6 and we observe no significant deviation from the basic model.



(a) Uniform design, repeated sequences



(b) Uniform design, fully randomized



(c) Optimized design, fully randomized

Figure 8.4: The observed decays in success probability for each of the three experiments where we intentionally introduced coherent errors. The experiment in plot (a) has sequence lengths chosen uniformly in the range [5,2000] and for each sequence length 100 random sequences ar drawn and run 100 times each. The experiment in plot (b) has the same sequence lengths as the first experiment, but is fully randomized so at each sequence length 100×100 random sequences are drawn and run once each. Each of the three experiments takes the same total time of approximately 40 minutes. All other aspects of the plots are the same as in Fig. 8.1. As we describe in Fig. 8.6 we do not find significant evidence to reject the basic model. As a result we conclude that the values of the second moments estimated here are not significantly different from zero.

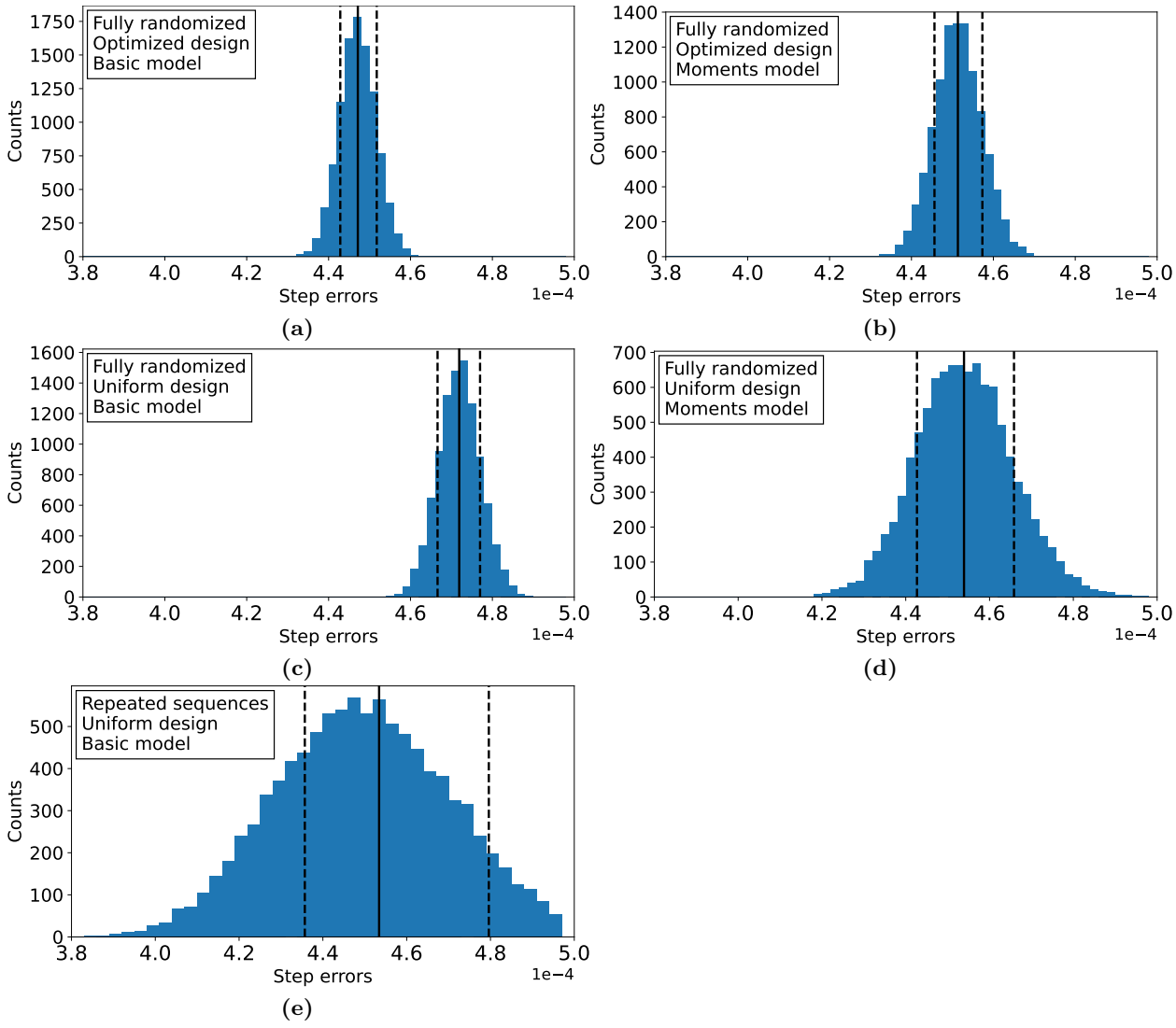


Figure 8.5: The bootstrap distributions obtained during the analysis of the three experiments we ran during the comparison where we intentionally introduced coherent errors. All aspects of the plots are the same as in Fig. 8.2. The best fit step errors for the plots are (a) $4.47^{+0.05}_{-0.04} \times 10^{-4}$, (b) $4.51^{+0.06}_{-0.06} \times 10^{-4}$, (c) $4.72^{+0.05}_{-0.05} \times 10^{-4}$, (d) $4.54^{+0.12}_{-0.11} \times 10^{-4}$, (e) $4.53^{+0.26}_{-0.18} \times 10^{-4}$.

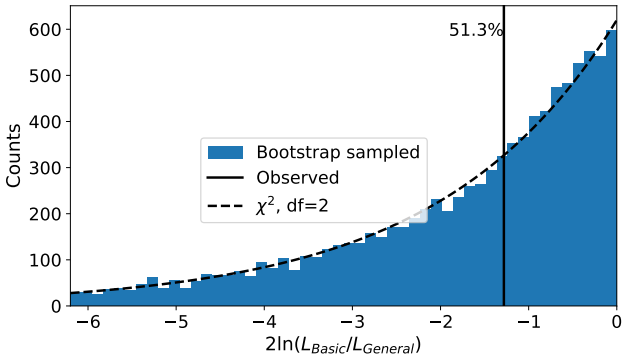


Figure 8.6: Results of an empirical likelihood ratio test for the optimized, fully randomized experiment. We perform the procedure described in Section 7.1, with the basic model as the inner model and the general model as the outer model. We find a p-value to reject the basic model of 51.3%, which indicates little evidence for rejection. All other aspects of the plots are the same as in Fig. 8.3.

Chapter 9

Non-markovianity in FRB

In this chapter we consider error processes for randomized benchmarking that are modeled as a quantum channel acting jointly on the system S of interest and an environmental system E that cannot be directly accessed. In contrast, most work on the error analysis of randomized benchmarking assumes that the error processes can be described by a quantum channel acting only on the system S of interest. Error channels that act only on system S are called Markovian error processes, and error processes that must be modeled as a quantum channel acting jointly on S and E are typically referred to as non-Markovian error processes. For prior work on non-Markovian errors in randomized benchmarking, we refer to Refs. [26, 27, 28, 38]. Here we specifically study the situation where error channels acting on the joint SE system are twirled with perfect Pauli or Clifford gates acting only on system S. The motivation for considering Pauli and Clifford randomization is to understand the space of possible error processes that describe Pauli-randomized and Clifford-randomized benchmarking experiments respectively, in the presence of gate-independent non-Markovian errors. We will consider the concatenation of such twirled channels, without assuming that the channels in the concatenation are the same. This allows us to consider error processes that are possibly time-dependent.

9.1 Perfect Pauli Twirling

We start by analyzing Pauli-randomized error processes. Let Λ_{SE} denote the error channel on the joint system SE and let K_{SE} be a Kraus operator of the channel. Without loss of generality we

can assume that the error channel Λ_{SE} has been purified so that K_{SE} is the only Kraus operator. We denote Pauli operators on the joint system by $P_i^{(S)} \otimes P_j^{(E)}$, and the full set of Pauli operators can be obtained by allowing the index i to run over all Pauli operators on system S and the index j to run over all Pauli operators on system E. The Kraus operator K_{SE} can be decomposed into

$$K_{SE} = \sum_{ij} C_{ij} P_i^{(S)} \otimes P_j^{(E)}. \tag{9.1}$$

The corresponding action on a state ρ_{SE} can be written as

$$K_{SE} \rho K_{SE}^\dagger = \sum_{ii'jj'} C_{ij} C_{i'j'}^* P_i^{(S)} \otimes P_j^{(E)} \rho_{SE} P_{i'}^{(S)} \otimes P_{j'}^{(E)}. \tag{9.2}$$

Pauli randomization on system S with perfect Pauli gates will eliminate all coefficients with $i \neq i'$. The action on ρ_{SE} then simplifies to

$$\sum_{ijj'} C_{ij} C_{ij'}^* P_i^{(S)} \otimes P_j^{(E)} \rho_{SE} P_i^{(S)} \otimes P_{j'}^{(E)}. \tag{9.3}$$

This action on ρ_{SE} can be reinterpreted as a completely-positive trace-preserving map with Kraus operators indexed by i, where the ith Kraus operator $\widetilde{K_i}$ is

$$\widetilde{K_i} = P_i^{(S)} \otimes \left(\sum_j C_{ij} P_j^{(E)} \right). \tag{9.4}$$

If we denote the system E operator $\sum_j C_{ij} P_j^{(E)}$ by $\widetilde{K}_i^{(E)}$, we write this as

$$\widetilde{K_i} = P_i^{(S)} \otimes \widetilde{K}_i^{(E)}. \tag{9.5}$$

The state after the twirled joint channel is

$$\rho_{\text{out}} = \sum_i P_i^{(S)} \otimes \widetilde{K}_i^{(E)} \rho_{SE} P_i^{(S)} \otimes \widetilde{K}_i^{\dagger(E)}. \tag{9.6}$$

This can be interpreted as a mixture of completely-positive maps, indexed by i. The ith map has one Kraus operator $P_i^{(S)} \otimes \widetilde{K}_i^{(E)}$, and the mixture probabilities p_i are determined by the trace according to

$$p_i = \mathrm{tr}\left[P_i^{(S)} \otimes \widetilde{K}_i^{(E)} \rho_{SE} P_i^{(S)} \otimes \widetilde{K}_i^{\dagger(E)}\right]. \tag{9.7}$$

As a concrete example, we consider a one qubit system S and a one qubit environmental system E and the following error process, which corresponds to an erroneous coherent rotation around $Z_S \otimes Z_E$,

$$\Lambda_{SE} = \cos\theta I_S \otimes I_E + \sin\theta Z_S \otimes Z_E. \tag{9.8}$$

After perfect Pauli twirling on system S, the resulting channel has two Kraus operators. The first Kraus operator K_0 is

$$K_0 = \cos\theta I_S \otimes I_E, \tag{9.9}$$

and the second Kraus operator K_1 is

$$K_1 = \sin\theta Z_S \otimes Z_E. \tag{9.10}$$

In this particular case the twirled channel is a joint Pauli channel on the full system SE.

9.2 Perfect Clifford twirling

Clifford twirling on system S will have the effect of converting the Pauli gates in Eq. 9.6 into depolarizing channels. To see this directly, consider a completely-positive map Φ with a single Kraus operator $K = P^{(S)} \otimes K^{(E)}$, as in Eq. 9.5. For now we will assume that $P^{(S)}$ is a non-identity Pauli on system S. In this case, applying Clifford twirling on system S to Φ will lead to a map $\widetilde{\Phi}$ with Kraus operators $\{\widetilde{K}_i\}_i$, where

$$\widetilde{K}_i = \frac{1}{|\mathcal{P}| - 1} P_i^{(S)} \otimes K^{(E)}, \tag{9.11}$$

and where $|\mathcal{P}|$ is the number of generalized Pauli operators on system S. The second tensor factor in Eq. 9.11 does not depend on the index i, which means that $\widetilde{\Phi}$ can be expressed as

$$\widetilde{\Phi} = \Phi^{(S)} \otimes \Phi^{(E)}, \tag{9.12}$$

that is, as the tensor product of a completely-positive map $\Phi^{(S)}$ on system S with a completely-positve map $\Phi^{(E)}$ on system E. In this tensor product, $\Phi^{(S)}$ is the maximal-parameter depolarizing channel D_e, as described in Ch. 2, and $\Phi^{(E)}$ has a single Kraus operator $K^{(E)}$. It remains to consider mixtures of Kraus operators of the form in Eq. 9.5, where the mixture is over the different non-identity Paulis on system S. This will now lead to a mixture over tensor product channels each of the form in Eq. 9.12. In this mixture, the system S tensor factor is always the maximal-parameter depolarizing channel D_e, and the system E tensor factor can depend on the mixture component. As a result, the full channel is still of the form in Eq. 9.12, but the completely-positive map $\Phi^{(E)}$ can in general have multiple Kraus operators. Finally, we consider the case where the operator $P^{(S)}$ is the identity operator. In this case, the Clifford twirling on system S has no effect, and the resulting map is of the form in Eq. 9.12 where $\Phi^{(S)}$ is the identity channel.

Altogether, it follows that applying perfect Clifford twirling on system S to an arbitrary error channel Λ_{SE} will transform it into an error channel $\widetilde{\Phi}$ that is a mixture of at most two completely positive maps that are tensor product across SE. The first completely positive map in the mixture is

$$\widetilde{\Phi}_0 = I^{(S)} \otimes \Phi_0^{(E)} \tag{9.13}$$

and the second is

$$\widetilde{\Phi}_e = D_e^{(S)} \otimes \Phi_e^{(E)}, \tag{9.14}$$

where $\Phi_0^{(E)}$ and $\Phi_e^{(E)}$ are completely-positive maps on system E that are determined by the original channel Λ_{SE}.

9.3 Concatenated Clifford-twirled error channels

To study randomized benchmarking with perfect Clifford twirling and non-Markovian time-dependent errors, it is useful to study the concatenation of Clifford-twirled non-Markovian error processes. First we consider the composition of two such error processes, and then generalize to an arbitrary composition. For clarity we drop the S, E, SE labels in what follows. Let $\Lambda^{(1)}$ and $\Lambda^{(2)}$ be two channels on the joint system SE. According to the results of the previous section, after perfect Clifford twirling on system S, the resulting channels can each be described as a mixture of two tensor product completely-positive maps. To simplify the summation notation, we will introduce the notation D_0 for the identity channel and write the two completely-positive maps in Eq. 9.13 and Eq. 9.14 as $D_0 \otimes \Phi_0$ and $D_e \otimes \Phi_e$ respectively. We will use superscripts in parenthesis to distinguish between the maps associated to $\Lambda^{(1)}$ and $\Lambda^{(2)}$. Altogether, the composition of the twirled channels can be written as follows

$$\Phi^{(1)} \circ \Phi^{(2)} = \sum_{a_1 \in \{0,e\}} \sum_{a_2 \in \{0,e\}} D_{a_1} \circ D_{a_2} \otimes \Phi^{(1)}_{a_1} \circ \Phi^{(2)}_{a_2}. \tag{9.15}$$

This can be interpreted as a mixture of different compositions of depolarizing channels on system S, given by $D_{a_1} \circ D_{a_2}$, where each composition in the mixture is correlated with some completely-positive map on system E, given by $\Phi^{(1)}_{a_1} \circ \Phi^{(2)}_{a_2}$. The probabilities of the mixture components are determined by the initial state on the SE system and the maps $\Phi^{(1)}_{a_1} \circ \Phi^{(2)}_{a_2}$ on system E.

In the case of a sequence of error channels $\{\Lambda^{(i)}\}_i$, the expression for the composition of the twirled channels is similar. Using product notation to indicate a composition, we have

$$\prod_i \Phi^{(i)} = \sum_{a_i \in \{0,e\}} \prod_i D_{a_i} \otimes \prod_i \Phi^{(i)}_{a_i}. \tag{9.16}$$

In the previous equation the notation $\sum_{a_i \in \{0,e\}}$ indicates a sum over all assignments of the symbols $\{0, e\}$ independently to each of the indices a_i. The probabilities of the mixture components in Eq. 9.16 are determined by the initial state on the SE system and the maps $\prod_i \Phi^{(i)}_{a_i}$ on system E.

An important point that will be used in the next section is that the composition of depolarizing channels $\prod_i D_{a_i}$ for some assignment $a_i \in \{0, e\}$ will be equal to the maximal-parameter depolarizing channel composed with itself k times, for some k. This follows because D_0 is the identity channel and can simply be dropped whenever the index a_i is assigned to 0.

9.4 Revivals in RB with non-Markovian errors

A model of a randomized benchmarking trial with perfect Clifford twirling on the system S can be obtained by using the results of the previous section. We consider a list of system SE error channels $\{\Lambda^{(i)}\}_i$ and the list of corresponding error channels $\widetilde{\Phi}_i^{(i)}$ obtained by applying perfect Clifford twirling on system S to each element of $\{\Lambda^{(i)}\}_i$. For any given set of original error processes in the sequence, there will be a set of success probabilities at each sequence length that we denote $P(n)$. The function $P(n)$ is determined by the composition of the twirled error channels $\prod_i^n \widetilde{\Phi}^{(i)}$.

Revivals in the success probability occur if for any n we have $P(n+1) > P(n)$. Intuitively, revivals can occur because information about the computational state that has leaked to the environment can persist for some time and later return to the computational system. This behavior is not possible for Markovian error channels. Here we wish to bound the 'amount' of revivals possible in RB with non-Markovian, gate independent, error channels. To accomplish this, we must first choose an appropriate quantification of revivals, and then prove an upper bound for this quantity. To make an appropriate quantification of revivals, we first introduce the following quantity

$$r(P_1, P_2) = \max(0, P_2 - P_1). \tag{9.17}$$

Note that the function r is convex in P_1, P_2, see Chapter 3 of Ref. [6]. We then quantify the amount of revivals in a set of success probabilities $P(n)$ by defining the function R as follows

$$R = \sum_{n=1}^{N} r(P(n), P(n+1)). \tag{9.18}$$

The function R is a positive sum of convex functions and is therefore convex over the vector of inputs $P(n)$. As shown in Eq. 9.16, the composition of twirled channels that determine $P(n)$ can be interpreted as a mixture over tensor product completely-positive maps that are concatenated depolarizing channels on system S and completely-positive maps on system E. Because R is convex, it is bounded by its maximum over the success probabilities corresponding to the different components in the mixture in Eq. 9.16. A key observation is that, conditioned on a particular mixture component, the success probabilities $P(n)$ are determined only by the particular system S depolarizing channels in the concatenation up to the nth depolarizing channel. As shown in Eq. 9.13 and Eq. 9.14, the available depolarizing channels in the concatenation are only the identity and the maximal-parameter depolarizing channel. The identity channel does not change the success probability, and the function R has no additional dependence on n other than through $P(n)$. As a result, R is upper bounded by its value on the set of success probabilities obtained by considering an infinite sequence of maximal-parameter depolarizing channels acting only on system S. In such a case, R can be computed in closed form by summing a geometric sequence, as shown in the following section. This puts a concrete bound on the amount of revivals in a randomized benchmarking sequence, under the assumption that the Clifford twirling is performed perfectly.

9.5 Bounding revivals

As described in Chapter 1, if a depolarizing channel with parameter λ is concatenated with itself n times, the depolarizing parameter $\widetilde{\lambda}$ of the concatenated channel satisfies

$$\widetilde{\lambda} = 1 - (1 - \lambda)^n. \tag{9.19}$$

As stated in Section 2.3, the maximal-parameter depolarizing channel has a depolarizing parameter of $\lambda = 1 + 1/(D^2 - 1)$, where D is the Hilbert space dimension. Therefore, if the maximal-parameter depolarizing channel is concatenated with itself n times, the resulting depolarizing channel will have

a depolarizing parameter of

$$\widetilde{\lambda}_n = 1 - \left(\frac{-1}{D^2-1}\right)^n. \tag{9.20}$$

The fidelity of this channel satisfies $f_n = 1 - \widetilde{\lambda}_n(D-1)/D$, and substitution of the previous equation yields

$$f_n = 1 - \left(1 - \left(\frac{-1}{D^2-1}\right)^n\right)\left(\frac{D-1}{D}\right) = \frac{1}{D} + \frac{D-1}{D}\left(\frac{-1}{D^2-1}\right)^n. \tag{9.21}$$

Because the success probability of a depolarizing channel is equal to the fidelity, the success probabilities $P(n)$ of a sequence of maximal-parameter depolarizing channels are equal to the f_n in Eq. 9.21. It therefore remains to evaluate Eq. 9.18 for those values of $P(n)$. To do this, observe that the individual contributions in the sum in Eq. 9.18 are only nonzero when n is even. When n is even, the individual contribution is

$$r(P(n), P(n+1)) = \frac{D-1}{D}\left(\frac{1}{D^2-1}\right)^n + \frac{D-1}{D}\left(\frac{1}{D^2-1}\right)^{n-1}. \tag{9.22}$$

Setting an index $k = n-2$ that runs from 0 to infinity, this can be conveniently re-expressed as

$$r(P(n), P(n+1)) = \frac{D-1}{D}\frac{D^2}{D^2-1}\frac{1}{D^2-1}\left(\frac{1}{D^2-1}\right)^k. \tag{9.23}$$

Then, substitution into Eq. 9.18 gives

$$R = \frac{D-1}{D}\frac{D^2}{D^2-1}\frac{1}{D^2-1}\sum_{k=0}^{\infty}\left(\frac{1}{D^2-1}\right)^k. \tag{9.24}$$

This can be evaluated by summing the geometric sequence to get

$$R = \frac{(D-1)D}{(D^2-1)^2}\frac{1}{1-\frac{1}{(D^2-1)^2}} = \frac{D(D-1)}{(D^2-1)^2-1}. \tag{9.25}$$

The maximum possible value of R in any randomized benchmarking sequence with gate-independent errors on Clifford gates is therefore

$$R_{\max} = \frac{D(D-1)}{(D^2-1)^2-1}. \tag{9.26}$$

We summarize this result with the following proposition.

Proposition 7. *Consider a system S of dimension D_S and an environment E of dimension D_E. Consider a sequence of error channels $\{\Lambda_{SE}^n\}_{n=1}^N$, each acting jointly on SE and perfectly Clifford twirled on system S. Let the initial state be a product state $|\psi_S\rangle \otimes |\psi_E\rangle$. Let $P(n)$ be the fidelity of the state after the first n error channels have been applied, with respect to the initial state. To quantify the revivals in the fidelities $P(n)$, let R be defined as in Eq. 9.25, Then, R satisfies $R \leq \frac{D_S(D_S-1)}{(D_S^2-1)^2-1}$.*

References

[1] Andris Ambainis and Joseph Emerson. *Quantum t-designs: t-wise independence in the quantum world*. Number: arXiv:quant-ph/0701126 arXiv:quant-ph/0701126. 2007 (Cited on p. 25).

[2] Stefanie J. Beale and Joel J. Wallman. *Randomized compiling in fault-tolerant quantum computation*. 2023 (Cited on p. 10).

[3] Italo Pereira Bezerra, Hilma M. Vasconcelos, and Scott Glancy. "Quadrature squeezing and temperature estimation from the Fock distribution". In: *Quantum Information Processing* 21.11 (2022), p. 365. ISSN: 1573-1332 (Cited on p. 67).

[4] Robin Blume-Kohout. *Robust error bars for quantum tomography*. 2012 (Cited on p. 67).

[5] Dennis D. Boos. "Introduction to the Bootstrap World". In: *Statistical Science* 18.2 (2003). ISSN: 0883-4237 (Cited on pp. 66, 68).

[6] Stephen Boyd and Lieven Vandenberghe. *Convex Optimization*. Cambridge University Press, 2004. ISBN: 0521833787 (Cited on pp. 53, 93).

[7] K. R. Brown et al. "Single-qubit-gate error below 10^-4 in a trapped ion". In: *Physical Review A* 84.3 (2011). arXiv:1104.2552 [quant-ph], p. 030303. ISSN: 1050-2947, 1094-1622 (Cited on pp. 71, 72).

[8] S. C. Burd et al. "Quantum amplification of mechanical oscillator motion". In: *Science* 364.6446 (2019), pp. 1163–1165. ISSN: 0036-8075, 1095-9203 (Cited on p. 78).

[9] Arnaud Carignan-Dugas, Joel J. Wallman, and Joseph Emerson. "Characterizing universal gate sets via dihedral benchmarking". In: *Physical Review A* 92.6 (2015). ISSN: 1094-1622 (Cited on p. 2).

[10] Athena Ceasura et al. *Non-Exponential Behaviour in Logical Randomized Benchmarking*. 2022 (Cited on p. 5).

[11] Michael R. Chernick and Robert A. Labudde. "Revisiting Qualms about Bootstrap Confidence Intervals". In: *American Journal of Mathematical and Management Sciences* 29.3-4 (2009), pp. 437–456 (Cited on p. 67).

[12] Benoît Collins and Piotr Śniady. "Integration with Respect to the Haar Measure on Unitary, Orthogonal and Symplectic Group". In: *Communications in Mathematical Physics* 264.3 (2006), pp. 773–795. ISSN: 0010-3616, 1432-0916 (Cited on p. 77).

[13] Joshua Combes et al. *Logical Randomized Benchmarking*. 2017 (Cited on p. 5).

[14] Christoph Dankert et al. "Exact and approximate unitary 2-designs and their application to fidelity estimation". In: *Phys. Rev. A* 80 (1 2009), p. 012304 (Cited on pp. 1, 23, 25).

[15] Bas Dirkse, Jonas Helsen, and Stephanie Wehner. "Efficient unitarity randomized benchmarking of few-qubit Clifford gates". In: *Phys. Rev. A* 99 (1 2019), p. 012315 (Cited on p. 75).

[16] Zdenek Dostál. *Optimal quadratic programming algorithms: with applications to variational inequalities*. Vol. 23. Springer Science and Business Media, 2009 (Cited on p. 55).

[17] Frédéric Dupuis et al. "One-Shot Decoupling". In: *Communications in Mathematical Physics* 328.1 (2014), pp. 251–284. ISSN: 0010-3616, 1432-0916 (Cited on pp. 21, 22, 77).

[18] Bradley Efron. *The Jackknife, the Bootstrap and Other Resampling Plans*. Society for Industrial and Applied Mathematics, 1982 (Cited on p. 67).

[19] Bradley Efron and Robert J. Tibshirani. *An Introduction to the Bootstrap*. Monographs on Statistics and Applied Probability 57. Boca Raton, Florida, USA: Chapman & Hall/CRC, 1993 (Cited on pp. 66, 67, 80).

[20] Jens Eisert et al. "Quantum certification and benchmarking". In: *Nature Reviews Physics* 2.7 (2020), pp. 382–390. ISSN: 2522-5820 (Cited on p. 1).

[21] G. Elfving. "Optimum Allocation in Linear Regression Theory". In: *The Annals of Mathematical Statistics* 23.2 (1952), pp. 255 –262 (Cited on p. 50).

[22] Jeffrey M. Epstein et al. "Investigating the limits of randomized benchmarking protocols". In: *Phys. Rev. A* 89 (6 2014), p. 062321 (Cited on p. 26).

[23] Alexander Erhard et al. "Characterizing large-scale quantum computers via cycle benchmarking". In: *Nature Communications* 10.1 (2019). ISSN: 2041-1723 (Cited on pp. 1, 18).

[24] Simon J. Evered et al. "High-fidelity parallel entangling gates on a neutral-atom quantum computer". en. In: *Nature* 622.7982 (2023), pp. 268–272. ISSN: 0028-0836, 1476-4687 (Cited on p. 1).

[25] Valerii Fedorov. "Optimal experimental design". In: *WIREs Computational Statistics* 2.5 (2010), pp. 581–589 (Cited on pp. 51, 56).

[26] P Figueroa-Romero et al. "Operational Markovianization in randomized benchmarking". In: *Quantum Science and Technology* 9.3 (2024), p. 035020 (Cited on p. 88).

[27] Pedro Figueroa-Romero, Kavan Modi, and Min-Hsiu Hsieh. "Towards a general framework of Randomized Benchmarking incorporating non-Markovian Noise". In: *Quantum* 6 (2022), p. 868. ISSN: 2521-327X (Cited on p. 88).

[28] Pedro Figueroa-Romero et al. "Randomized Benchmarking for Non-Markovian Noise". In: *PRX Quantum* 2.4 (2021) (Cited on pp. 5, 26, 88).

[29] Steven T. Flammia. *Averaged circuit eigenvalue sampling*. 2021 (Cited on p. 18).

[30] William Fulton and Joe Harris. *Representation Theory*. Vol. 129. Graduate Texts in Mathematics. New York, NY: Springer New York, 2004. ISBN: 978-3-540-00539-1 978-1-4612-0979-9 (Cited on p. 11).

[31] J. P. Gaebler et al. "High-Fidelity Universal Gate Set for $^{9}Be^{+}$ Ion Qubits". In: *Phys. Rev. Lett.* 117 (6 2016), p. 060505 (Cited on pp. 1, 68).

[32] Jay M. Gambetta et al. "Characterization of Addressability by Simultaneous Randomized Benchmarking". In: *Phys. Rev. Lett.* 109 (24 2012), p. 240504 (Cited on p. 2).

[33] Daniel Gottesman. "Fault-Tolerant Quantum Computation with Higher-Dimensional Systems". In: *Chaos, Solitons & Fractals* 10.10 (1999). arXiv:quant-ph/9802007, pp. 1749–1758. ISSN: 09600779 (Cited on pp. 8, 9, 19).

[34] C. Granade, Christopher Ferrie, and D G Cory. "Accelerated randomized benchmarking". In: *New Journal of Physics* 17.1 (2015), p. 013042. ISSN: 1367-2630 (Cited on pp. 3, 73, 75).

[35] Radoslav Harman and Tomás Jurík. “Computing c-optimal experimental designs using the simplex method of linear programming”. In: *Computational Statistics & Data Analysis* 53.2 (2008), pp. 247–254 (Cited on pp. 50, 51).

[36] Robin Harper et al. “Statistical analysis of randomized benchmarking”. In: *Phys. Rev. A* 99 (5 2019), p. 052350 (Cited on p. 3).

[37] A. K. Hashagen et al. “Real Randomized Benchmarking”. In: *Quantum* 2 (2018), p. 85. ISSN: 2521-327X (Cited on p. 2).

[38] Markus Heinrich, Martin Kliesch, and Ingo Roth. *General guarantees for randomized benchmarking with random quantum circuits*. Number: arXiv:2212.06181 arXiv:2212.06181 [quant-ph]. 2023 (Cited on pp. 1, 88).

[39] J. Helsen et al. “General Framework for Randomized Benchmarking”. In: *PRX Quantum* 3.2 (2022) (Cited on pp. 1, 24).

[40] Jonas Helsen et al. “A new class of efficient randomized benchmarking protocols”. In: *npj Quantum Information* 5 (2019), p. 71. ISSN: 2056-6387 (Cited on p. 2).

[41] W. Huang et al. “Fidelity benchmarks for two-qubit gates in silicon”. In: *Nature* 569.7757 (2019), pp. 532–536. ISSN: 0028-0836, 1476-4687 (Cited on pp. 1, 71, 72, 76).

[42] A. C. Hughes et al. “Benchmarking a High-Fidelity Mixed-Species Entangling Gate”. In: *Phys. Rev. Lett.* 125 (8 2020), p. 080504 (Cited on pp. 1, 71, 72).

[43] Toshinari Itoko and Rudy Raymond. “Sampling Strategy Optimization for Randomized Benchmarking”. In: (2021), pp. 188–198 (Cited on p. 3).

[44] E. Knill et al. “Randomized benchmarking of quantum gates”. In: *Physical Review A* 77.1 (2008) (Cited on pp. 1, 24, 75).

[45] Alex Kwiatkowski et al. *Optimized experiment design and analysis for fully randomized benchmarking*. 2023 (Cited on pp. 24, 46, 50, 66, 78).

[46] Tzon-Tzer Lu and Sheng-Hua Shiou. “Inverses of 2 × 2 block matrices”. In: *Computers and Mathematics with Applications* 43.1 (2002), pp. 119–129. ISSN: 0898-1221 (Cited on p. 55).

[47] Yao Lu et al. “High-fidelity parametric beamsplitting with a parity-protected converter”. In: *Nature Communications* 14.1 (2023), p. 5767. ISSN: 2041-1723 (Cited on p. 1).

[48] Easwar Magesan, J. M. Gambetta, and Joseph Emerson. “Scalable and Robust Randomized Benchmarking of Quantum Processes”. In: *Phys. Rev. Lett.* 106 (18 2011), p. 180504 (Cited on p. 1).

[49] Easwar Magesan, Jay M. Gambetta, and Joseph Emerson. “Characterizing quantum gates via randomized benchmarking”. In: *Physical Review A* 85.4 (2012) (Cited on pp. 1, 24).

[50] Easwar Magesan et al. “Efficient Measurement of Quantum Gate Error by Interleaved Randomized Benchmarking”. In: *Phys. Rev. Lett.* 109 (8 2012), p. 080505 (Cited on p. 61).

[51] Adam M. Meier. *Randomized Benchmarking of Clifford Operators*. 2018 (Cited on pp. 1, 21, 23–25, 41, 80).

[52] S. A. Moses et al. “A Race-Track Trapped-Ion Quantum Processor”. In: *Phys. Rev. X* 13 (4 2023), p. 041052 (Cited on p. 1).

[53] Frank Nielsen. *Cramer-Rao Lower Bound and Information Geometry*. 2013 (Cited on pp. 50, 51, 56).

[54] Frank Nielsen. "An Elementary Introduction to Information Geometry". In: *Entropy* 22.10 (2020), p. 1100 (Cited on p. 56).

[55] Michael A. Nielsen and Isaac L. Chuang. *Quantum Computation and Quantum Information*. Cambridge University Press, 2000 (Cited on pp. 7, 15).

[56] Melissa E. O'Neill. *PCG: A Family of Simple Fast Space-Efficient Statistically Good Algorithms for Random Number Generation*. Tech. rep. HMC-CS-2014-0905. Claremont, CA: Harvey Mudd College, 2014 (Cited on p. 78).

[57] E. Onorati, A. H. Werner, and J. Eisert. "Randomized Benchmarking for Individual Quantum Gates". In: *Phys. Rev. Lett.* 123 (6 2019), p. 060501 (Cited on p. 2).

[58] Felix A. Pollock et al. "Operational Markov Condition for Quantum Processes". In: *Phys. Rev. Lett.* 120 (4 2018), p. 040405 (Cited on p. 25).

[59] Timothy Proctor et al. "What Randomized Benchmarking Actually Measures". In: *Phys. Rev. Lett.* 119 (13 2017), p. 130502 (Cited on pp. 1, 26).

[60] Timothy J. Proctor. *Quantum information with general quantum variables: a formalism encompassing qubits, qudits, and quantum continuous variables*. Number: arXiv:1903.08545 arXiv:1903.08545 [quant-ph]. 2019 (Cited on pp. 8, 9).

[61] Friedrich Pukelsheim. *Optimal Design of Experiments*. Society for Industrial and Applied Mathematics, 2006 (Cited on pp. 50, 51, 56).

[62] Kenneth Rudinger et al. "Probing context-dependent errors in quantum processors". In: *Physical Review X* 9.2 (2019). arXiv:1810.05651 [quant-ph], p. 021045. ISSN: 2160-3308 (Cited on p. 68).

[63] Guillaume Sagnol. "Computing optimal designs of multiresponse experiments reduces to second-order cone programming". In: *Journal of Statistical Planning and Inference* 141.5 (2011), pp. 1684–1708. ISSN: 0378-3758 (Cited on p. 51).

[64] Nathaniel Schenker. "Qualms About Bootstrap Confidence Intervals". In: *Journal of the American Statistical Association* 80.390 (1985), pp. 360–361. ISSN: 01621459 (Cited on p. 67).

[65] Travis L. Scholten and Robin Blume-Kohout. "Behavior of the maximum likelihood in quantum state tomography". In: *New Journal of Physics* 20.2 (2018), p. 023050. ISSN: 1367-2630 (Cited on p. 68).

[66] Anders Sørensen and Klaus Mølmer. "Entanglement and quantum computation with ions in thermal motion". In: *Phys. Rev. A* 62 (2 2000), p. 022311 (Cited on p. 25).

[67] R. Srinivas et al. "High-fidelity laser-free universal control of trapped ion qubits". In: *Nature* 597.7875 (2021), pp. 209–213. ISSN: 0028-0836, 1476-4687 (Cited on p. 78).

[68] Joel Wallman et al. "Estimating the coherence of noise". In: *New Journal of Physics* 17.11 (2015), p. 113020. ISSN: 1367-2630 (Cited on pp. 2, 75).

[69] Joel J. Wallman. "Randomized benchmarking with gate-dependent noise". In: *Quantum* 2 (2018). arXiv:1703.09835 [quant-ph], p. 47. ISSN: 2521-327X (Cited on pp. 1, 25, 26).

[70] Joel J Wallman and Steven T Flammia. "Randomized benchmarking with confidence". In: *New Journal of Physics* 16.10 (2014), p. 103032 (Cited on pp. 2, 73).

[71] Zak Webb. *The Clifford group forms a unitary 3-design*. 2016 (Cited on pp. 19, 20, 23).

[72] N.A. Weiss, P.T. Holmes, and M. Hardy. *A Course in Probability*. Pearson Addison Wesley, 2005. ISBN: 9780321189547 (Cited on p. 75).

[73] Adam Winick et al. *Concepts and conditions for error suppression through randomized compiling*. 2022 (Cited on p. 10).