

UC Berkeley

UC Berkeley Electronic Theses and Dissertations

Title

The Toric Code at Finite Temperature

Permalink

<https://escholarship.org/uc/item/56k7v7kq>

Author

Freeman, Christian Daniel

Publication Date

2018

Peer reviewed|Thesis/dissertation

The Toric Code at Finite Temperature

by

Christian Daniel Freeman

A dissertation submitted in partial satisfaction of the

requirements for the degree of

Doctor of Philosophy

in

Physics

in the

Graduate Division

of the

University of California, Berkeley

Committee in charge:

Professor Birgitta Whaley, Co-chair

Professor Joel Moore, Co-chair

Professor Irfan Siddiqi

Professor Martin Head-Gordon

Summer 2018

The Toric Code at Finite Temperature

Copyright 2018
by
Christian Daniel Freeman

Abstract

The Toric Code at Finite Temperature

by

Christian Daniel Freeman

Doctor of Philosophy in Physics

University of California, Berkeley

Professor Birgitta Whaley, Co-chair

Professor Joel Moore, Co-chair

Alexei Kitaev’s toric code is a rich model, that has birthed and stimulated the development of topological quantum computing, error correction, and field theory. It was also the first example of a quantum error correcting code that could resiliently store quantum information in certain types of noisy environments without the need for active error correction. Unfortunately, the toric code loses much of its power as a noise-resilient quantum memory at any finite temperature. Many of the problems with the toric code at finite temperature are likewise shared among its cousin stabilizer codes. These problems can be traced to the proliferation of stringlike “defects” in these codes at finite temperature. The aim of this thesis is then twofold.

First, I characterize both numerically and theoretically the failure modes of the toric code at finite temperature with fairly modest bath assumptions. I achieve this by numerically sampling the nonequilibrium dynamics of the toric code using a continuous time monte carlo algorithm. From this analysis, I was able to derive an exact expression for the low and high temperature dynamics of the toric code, as well as a regime over which the toric code is “most” stable to noise.

Secondly, by leveraging the results of this analysis, I construct algorithms that aim to suppress these noise channels. These algorithms are broadly separated into measurement-free and few-measurement protocols. The measurement-free algorithms are reminiscent of dynamical-decoupling pulse sequences, applied in parallel or in serial to a target stabilizer code, and are entirely unitary. I find that measurement-free algorithms can provide a constant factor increase to the lifetime of a large class of stabilizer codes at finite temperature. The few-measurement protocols are more complex, but I provide evidence that they can provide a threshold for the toric code with asymptotically fewer measurements than was previously known to be achievable.

To my parents, my brother Math, Michael, and Laura,
Because at my core, I am a superposition of all of you.

Contents

Contents	ii
List of Figures	v
1 Introduction	1
1.1 Overview	2
1.2 High Level Background	3
2 Errors at zero temperature	5
2.1 From Schrodinger to von Neumann	5
2.2 From Error Maps to Unitaries	8
2.3 Stabilizer Codes	11
2.4 Perfect decoding of the Ising model	13
3 The Toric Code	15
3.1 Preliminaries	15
3.2 Ground and excited states	16
3.3 Construction of Ground State	18
3.4 Quasiparticles	19
3.5 Zero temperature resilience	20
3.6 Decoders	21
3.7 Zero temperature mapping to RBIM	24
3.8 Finite temperature mapping to 1-D Ising Model	25
4 Errors at finite temperature	28
4.1 The Lindblad Equation - A Simple Discrete Derivation	28
4.2 The Lindblad Equation - The Long Continuous Derivation	32
5 An interlude on numerical methods	39
5.1 Markov Processes	39
5.2 Continuous time Monte Carlo	43
6 The toric code at finite temperature	46

6.1	Introduction	46
6.2	The toric code	48
6.3	Dynamics of the Toric Code in contact with an external bath	51
6.4	Topologically non-trivial random walks on the torus	58
6.5	Real time Monte Carlo Simulation of the toric code dynamics	63
6.6	Discussion	67
6.7	Acknowledgments	70
6.8	Resummation method	70
6.9	Finite size scaling in the 1D Ising model	71
7	Autonomous algorithms without measurements	73
7.1	Introduction	73
7.2	Stabilizer Codes	75
7.3	1D Ising Model	78
7.4	The Protocol	81
7.5	Higher Dimensions and Generalization	89
7.6	Discussion	99
7.7	Acknowledgments	100
7.8	Syndrome Decoding for 1D Ising Model	100
7.9	MATCHSEQ and Error Correction	100
7.10	Algorithm for 1-D Ising Model	104
8	Autonomous algorithms with measurements	105
8.1	Introduction	105
8.2	Stabilizer codes at Finite Temperature	107
8.3	Few Measurement Error Correction Algorithm	110
8.4	Finite Temperature Simulations	115
8.5	Generalization to Higher Dimension	118
8.6	Discussion	120
8.7	Acknowledgments	122
8.8	Bayesian Decoding for the Ising Model	123
8.9	Alternative Algorithm for Estimating Defect Lifetimes	125
8.10	Alternative decoding schemes	128
9	Additional topics	129
9.1	Tensor Networks, Transport, and Efficient State Ansatzes	129
9.2	Topology and Geometry in Neural Networks	133
10	Conclusion	136
10.1	Overview and Outlook	136
10.2	Contemporaneous and ongoing works on quantum memories	137
10.3	Epilogue	138

Bibliography**139**

List of Figures

3.1	A piece of square toric code lattice in solid black lines, with its dual lattice indicated in light blue dotted lines. Vertices of the dual lattice rest on plaquettes of the primary lattice, and vice versa. Note that by individually rotating the edges of the black lattice, one retrieves the dual.	17
3.2	Ground state splitting in the toric code as a function of applied field strength. Adapted from Trebst et al. https://arxiv.org/pdf/cond-mat/0609048.pdf . Our η corresponds to their “loop tension”, h	22
3.3	Carpet of σ_x errors after error correction for a system at criticality. The string colored in red is the shortest nontrivial error. All other errors are homologically trivial. Figure generated from data. The locations where the red string seems to break are simply points where the error string wraps around the lattice.	23
3.4	A small sublattice of the toric code with a string of errors (bold red), a corrective operation (dotted red), and auxiliary plaquette labels (plus and minus) indicating possible spin configurations on an equivalent RBIM lattice.	25
3.5	The transformed toric code lattice numbering convention for Wen’s Plaquette model.	26
5.1	A directed graph representation of an example Markov chain with possible state variables a, b , and c . Transition probabilities / transition rates are encoded by the directed quantities γ_{ij}	40
6.1	The vertex (A_v) and plaquette (B_p) operators of the toric code as defined in (8.16). Edges marked in red are operated on by σ^x while those marked in blue are operated on by σ^z	49
6.2	A torus with a self-correcting error string (left) and an uncorrectable error string (right).	52
6.3	A depiction of the operation of the Lindblad operators in the master equation describing the interaction of the toric code with an external bath, as defined in (8.4).	56

6.4	The eight starting geometries (translucent green, blue) for Monte Carlo simulations of random walks. The solid green site denotes the origin at which the “fixed” quasi-particle sits. Blue configurations were sampled twice as often as green configurations, owing to the different likelihoods of different starting geometries. The simulation was terminated when the traveling quasiparticle reached one of the translucent red vertices—i.e. an annihilation geometry.	60
6.5	Probability of annihilation $p^t(n)$ after n steps on the torus for the model described in section 6.4, as computed via Monte Carlo. We see that the value of $p^t(n)$ agrees with the planar value $p^p(n)$ (see (6.34) and [94]) up until a characteristic value of n where it is possible for the walker to make topologically nontrivial walks on a finite size lattice.	61
6.6	The finite-size scaling of the characteristic departure (n_d) and crossing (n_c) times from an analysis of the Monte Carlo data shown in Fig. 6.5. The lines represent the best fit power laws to the three largest system sizes.	62
6.7	The probability of topologically non-trivial annihilations on a torus, $P_{2D}^Q(L)$ as a function of system size, as computed by Monte Carlo simulations. Also shown are the bounds P_c and P_d , which are computed from $p^t(n)$ and $p^p(n)$, as described in the text. The lines represent fits to $(\ln L)^{-1}$ of the three largest system sizes. . .	63
6.8	Time evolution of the expectation value of the Π_{++} operator (defined in section 6.5) computed via Monte Carlo, where $1/4$ has been subtracted to reveal the exponential decay. These simulations were initialized to a pure ground state with $L = 128$ with $T = \{0.02, 0.08, 0.14, 0.2\}$ respectively for subfigures (a)-(d). The black lines represent exponential fits to the Monte Carlo data. Note the stretched exponential behavior for early times in (c). γ_0 has been set to 1.	64
6.9	justification=raggedright	65
6.10	Low temperature regime: Finite-size scaling of Γ_{++} in the low temperature regime where we have collapsed the temperature dependence of the data according to (6.26). The solid lines are the low temperature model (T) predictions as described in (6.28); these lines nearly completely overlap due to the weak residual temperature dependence in (6.26). The dotted line indicates the best fit to purely linear scaling in L which is expected in the high temperature regime (T_H).	67
6.11	High temperature regime: finite-size scaling of Γ_{++} in the high temperature regime. The solid line represents a fit of all high temperature data (T_H) to a linear scaling in L . The dotted line indicates a best fit to the poly-log scaling $L^2/\log L$ which is expected in the low temperature regime (T_C).	68
6.12	The dynamical crossover temperature T_{dyn}^* as a function of system size L . The line represents the fit to $\ln(L)^{-1}$ scaling for the largest system sizes. Also shown is the equilibrium crossover temperature T_{eq}^* as defined in Ref. [23].	69

6.13	Various configurations of domain walls in 1 dimension. a) An “annihilation geometry”, where the domain walls (in red) are separated by a single spin. b) A “free” pair of domain walls. This configuration will annihilate trivially with probability $1/2$. c) Domain walls are further separated. d) Domain walls are separated by half the system size. The rightmost domain wall is the same as the leftmost domain wall due to periodic boundary conditions.	71
7.1	A snapshot of a linear stabilizer code with detection/correction operators O . The optimal error correction operator has nontrivial system size scaling, and in general depends on the particular code being used.	78
7.2	Circuit for performing error suppression for a subregion of the 1D Ising model. Qubits q_1 through q_4 are lattice sites on an Ising chain. s_1 through s_3 are ancillas used to read out the syndrome measurements of ZZ on the nearest-neighbor Ising lattice sites. Based on the results of the syndrome measurements, the conditional unitary operator CU corrects the errors present. A table which defines CU is provided in Appendix 7.8. This entire circuit represents a possible realization of one such operator O from Fig. 7.1.	82
7.3	Sequence of DSWAPs, denoted C , for a $\lambda = 3$ λ -mixing protocol. If a pair of domain walls exist anywhere between sites q_1 through q_7 , then they will necessarily be brought adjacent to each other by this sequence of swaps. Gates are applied sequentially with waiting time $1/\chi$ between each gate.	85
7.4	The same sequence from Fig. 7.3 but shown acting on domain-wall variables. Here, it is clear that the sequence of DSWAPs is designed not to mix domain walls between the two regions of size $\lambda = 3$. Site b_1 sits between q_1 and q_2 , b_2 between q_2 and q_3 , etc.	85
7.5	One possible snapshot of the error correction process. <i>a)</i> depicts a system with two domain walls present, each sitting in adjacent λ -domains. <i>b)</i> depicts the state of the system after the protocol has been applied—domain walls have been shuttled to the shared boundary. In <i>c)</i> , the bath dissipates the domain walls, and the system returns to the ground state.	86
7.6	Lifetime of the Ising model, Γ^{-1} , expressed in terms of inverse units of Γ_0 (i.e., (7.13)) as a function of the correction rate, χ , rescaled by the translation rate, γ_0 , for different values of λ and for fixed system size $L = 96$, temperature, $T = .07$, and translation rate, $\gamma_0 = .0007$. Protocols were implemented in parallel on $48/\lambda$ blocks (see text). In the absence of the protocol, the lifetime of the Ising model for these parameters corresponds to approximately Γ_0^{-1} , i.e. (7.13). This is the value which all three protocols converge towards in the limit of $\chi/\gamma_0 \ll 1$. Note the decrease in lifetime for $\chi/\gamma_0 \approx 1$, as well as the universal scaling in the lifetime up until the λ -dependent cutoff (i.e., (7.17)), which increases as λ decreases. The only remaining mismatch between the curves is due to the residual λ dependence in the $1/(L - 2 - 2\lambda)$ term of (7.15).	90

7.7	Lifetime of the Ising model as a function of the correction rate, χ , rescaled by the translation rate, γ_0 , for different values of λ , and for fixed system size, $L = 96$, temperature, $T = .07$, and translation rate, $\gamma_0 = .0007$. Protocols were implemented serially (see text). The scaling of lifetime with χ is characteristically similar to the parallel case; however, the maximal lifetime is correspondingly smaller for the serial implementation. Note that smaller λ still yields a larger enhanced lifetime.	91
7.8	Lifetime of the Ising model as a function of the correction rate, χ , rescaled by the translation rate, γ_0 , for different values of system size, L , and for fixed $\lambda = 3$, $T = .07$, $\gamma_0 = .0007$. Protocols were implemented in parallel on $L/2\lambda$ blocks (see text). Note the linear scaling in χ for small values, as well as the shift in the maximum of the lifetime as a function of L	92
7.9	This figure contains the same data as Fig. 7.8, but with the χ axis rescaled to χL , and the $1/\Gamma_{\text{Cyc}}$ axis rescaled by (7.14). Hence, the linear scaling in χ , and the slight residual system size dependence have been removed. Note the steep, sudden dropoff in lifetime after $\chi L \sim 55$	93
7.10	The critical cycling rate, χ_c , rescaled by the translation rate, γ_0 , as a function of $\frac{1}{L}$ for $\lambda = 3$, $T = .07$, $\gamma_0 = .0007$. Protocols were implemented in parallel on $L/2\lambda$ blocks. This scaling is consistent with the error model ansatz in (7.17). Fit to $1/L$ in red. Errors are dominated by systematic effects, not sampling error.	94
7.11	A single random application of the bath hopping operator causes a defect to move between adjacent regions of size λ . Once a pair of defects are separated by this distance, the protocol will not be able to correct them with certainty.	96
7.12	A single random application of the bath hopping operator causes a defect to move between nonintersecting regions of size λ , depicted in (a) and (b). (c) depicts a new lattice tiling where no single swap can move defects into two λ regions which do not share a boundary.	97
7.13	An example of the result of applying a winning sequence $M(G)$ to a graph with defect pairs present. No matter where the defects are, the sequence of DSWAPs brings pairs adjacent, whereupon they immediately fuse.	103
8.1	This cartoon illustrates the “centering” procedure for detected defects on a unit cell. Spin variables are in gray, and domain wall variables are in blue (no defect present) and red (defect present). When a defect is detected on a measurement patch (blue box), it is swapped to the center of the measurement patch via the DSWAP operator (black arrows). The defect immediately adjacent to it is also swapped onto the measurement patch so as not to pull apart defect pairs that would have otherwise dissipated. Measurement patch lengthscale λ indicated by arrow on top, and unit cell lengthscale λ indicated by arrow on bottom. The measurement fraction is defined as $m \equiv \lambda_m/\lambda$	110

8.2	Lifetime <i>enhancement</i> of several system sizes L as a function of temperature T using a measurement fraction of $m = 3/7$. Dotted line indicates example fit for threshold temperature T_{th} extraction for $L = 224$ data. Below $T \approx 0.16$, we find the lifetime to grow exponentially with L , indicative of a finite-temperature threshold. <i>Inset</i> : finite size scaling of T_{th} with inverse length $1/L$. Extrapolation to the infinite system limit yields a threshold temperature of $T_{th} = .155(6)$	116
8.3	justification=raggedright	117
8.4	justification=raggedright	119
8.5	justification=raggedright	120
8.6	Here we sketch one possible $m = 63/144$ (for a 12x12 unit cell) geometry for the measurement rails for a realization of our protocol. More sparse geometries can be realized simply by moving the rails of measurement farther apart. Measured sites are in light blue, and vertex locations for the toric code are circles. Spin variables (not pictured) reside directly between neighboring vertices.	121
8.7	Comparison of three different functional forms used as proxies for $P(d_1 : d_2)$. Protocol 1 is simply the error function expression given by Eq. 8.12. Protocol 2 is the probability density $\frac{1}{2\pi D t_2-t_1 } \exp(-\frac{ x_2-x_1 ^2}{2D t_2-t_1 })$. Protocol 3 is the more complicated expression in the last line of Eq. 8.18. In practice, each approximation for $P(d_1 : d_2)$ is seen to perform approximately equally well.	127
9.1	Graphical shorthand for tensors. Unique tensors are associated with vertices of graphs. Edges connecting vertices correspond to summed indices. Free edges are free indices.	131
9.2	A cartoon of a neural network mapping an input vector x_i to a single output. Grey boxes indicate the combined process of multiplying input data by a matrix followed by application of a nonlinearity. Edges indicate flow of data through the graph.	134

Acknowledgments

My six years at Berkeley can be reconstructed with high fidelity by considering projections along several distinct axes:

First, my Chemistry Friends, Danie Monahan, David Herlihy, Daniel Levine, Tobias Sjlander, Marissa Weichman, Paul Romer, (honorary chemist) Elaina Present, David Litt, Erin O'Brien—you were there, be it at parties, on roofs, in halls, at meals, at great meals, at Dinners, at Au Coquelet, during bad movies, during good movies, during mediocre movies, during puzzle hunts, during mystery hunts, during escape rooms, and everything else I probably forgot that was important or not and in-between. You were my graduate school.

Second, the Whaley group, particularly: Brendan Abolins, for many lunches and walks spanning worlds imagined and real, Chris Herdman, for being a patient and outstanding collaborator, Stephan Hoyer, for early and late words of wisdom, Norm Tubman, for listening to me and joining us in Berkeley (and for many a conversation, lunch, and nugget of advice), Bill Huggins, for patiently entertaining all of my crazy ideas, Song Zhang, for helping me back into our office every single time I locked myself out, and Gabriel Durkin, for more than a few wonderful conversations.

Third, to my advisor, and the leader of our little program, Birgitta Whaley, for guidance, for exacting standards, and for tolerating the frequent diversions of my random-walks in research space.

Fourth, to a variety of random collaborators and mentors throughout the years, particularly Joan Bruna, for jumpstarting my deep dive into machine learning, Brian Swingle, for a fantastic collaboration, Phill Geissler, for his impeccable lectures and advice, long-time Whaley group ancilla Mohan Sarovar, for many a long ride to Sandia, for guidance, and for a delightful collaboration, Bhaskar Dutta, for preparing me for the world of physics, and Simon North, for looking out for me throughout the entirety of my scientific career.

Fifth, to those that were there before, and will always be there after, my brother Math, my (scientific) brother Michael Grubb, my parents Mary and Den, and of course, above all and throughout, my wife, Laura.

Chapter 1

Introduction

This was first demonstrated in a clear way by Alan M. Turing (1912-1954), whose 1936 paper laid the foundations of the earth

King James Programming Tumblr

One of the holy grails of the field of quantum information is the construction fault tolerant qubits. These objects are the most fundamental, atomistic building blocks of a full quantum computing architecture—the quantum analogue of the bits of a classical device. Hundreds of proposals for achieving fault tolerance have been fielded since Peter Shor’s classic 9-qubit error correcting code—an active protocol for detection of correction of arbitrary single-qubit errors. What makes the toric code—the focus of this thesis—special, is its connection to a second, more theoretical (to the extent that any theoretical thing can be “more theoretical” than any other) goal: the finite temperature quantum memory.

The technical term “quantum memory” is a bit opaque—it evokes the idea of a hard drive—a register—something permanent and passively stable yet nonetheless quantum mechanical. “Permanent”, “passive”, and “stable” are hardly words often used to describe quantum mechanical devices, and for much of the early years of quantum information, there was very little expectation they could ever be describable as such. It was only after a suprising collection of facts were proven about early candidate quantum memories like the toric code, that people started seriously considering the possibility of a finite temperature quantum memory.

Probably the most intuitive way to define a quantum memory is by analogy with classical devices. The bits in the hard drives of classical computers are stable in part thanks to classical error correcting algorithms, but they’re also fundamentally stable because of a finite temperature Ising-like phase transition. Below the critical temperature (which is considerably higher than room temperature for, say, a laptop) the probability of a bit flip error scales like:

$$P(0 \rightarrow 1) \sim e^{-\zeta L}, \quad \text{for } T < T_c \quad (1.1)$$

where ζ is an inverse correlation length, T_c is the critical temperature, and L is the physical lengthscale of the bit—i.e., a mesoscopic chunk of iron-rich alloy. Exponentials grow quickly, thus bit flip errors can be extremely well suppressed by working with a large enough chunk of ferromagnetic stuff.

A quantum mechanical finite-temperature stable bit would be defined in much the same way. We would image some array of qubits where bit flip errors and phase flip errors are exponentially suppressed as a function of system size below some critical temperature:

$$P(|0\rangle \rightarrow -|0\rangle) \sim e^{-\zeta_1 L} \quad (1.2)$$

$$P(|0\rangle \rightarrow |1\rangle) \sim e^{-\zeta_2 L}, \quad \text{for } T < T_c \quad (1.3)$$

The “finite temperature” part bears elaboration. Much of the excitement of the toric code arose from its zero-temperature properties, and much of the body of this thesis details how the toric code *fails* at finite temperature. Much of the body of this thesis also concerns how to go about “fixing” the toric code. So, while having a finite temperature phase transition would be great for a realistic quantum device, it unfortunately appears to be out of reach for the near term in two and three dimensions, and all current proposals for quantum memories, including this one, only achieve fault tolerance using active error detection and correction methods—i.e., not relying completely on a finite temperature phase transition. For a review in the state of the art in proposals for theoretical realizations of quantum memories, see [22].

Morally, entirely relying on a finite temperature phase transition probably isn’t practical for an eventual quantum device, in exactly the same way that we don’t rely entirely on finite temperature phase transitions for classical devices. Eventual quantum devices will almost certainly have error detection/correction cycles fundamentally “baked-in” to their architecture. With this in mind, the protocols I develop in the latter half of this thesis are motivated by a desire to *minimize* resource usage—using no measurements at all, or as few as possible.

1.1 Overview

The remainder of this dissertation is organized into two main sections. The first section includes chapters 2, 3, 4, and 5 which develop the theoretical minimum for understanding the later chapters in a self contained way. Chapter 2 introduces quantum error correcting codes via the 1D Ising model—an analogy which will prove useful in later chapters. Chapter 3 introduces the toric code, and summarizes its surprising zero-temperature properties. Chapter 4 makes precise what I mean by “finite temperature” via the Lindblad formalism, and provides a discrete-time and a continuous-time derivation of the Lindblad equation, emphasizing important theoretical quantities used heavily in later chapters. Wrapping up the background chapters, Chapter 5 treats the continuous time monte carlo algorithm employed for all of my numerical results, as well as a discussion of Markov Chains which were used for modeling purposes.

The second section is composed of chapters 6, 7, and 8 which detail my studies of the Toric Code at finite temperature, as well as new algorithms I have developed for suppressing finite temperature noise. Chapter 6 explicitly derives the low and high temperature dynamics for the toric code subject to an Ohmic, Markovian bath, and is primarily the content of Freeman et al. [55]. This crossover is most unambiguously demonstrated in Figure 6.9, wherein the low and high temperature behaviors are shown to follow the proposed theoretical forms of the error rates. Chapter 7 contains a measurement-free algorithm which can increase the lifetime of a stabilizer code with string-like errors by a constant factor, and is primarily the content of Freeman, Herdman, and Whaley [54]. In sum, the performance of this algorithm for different resource requirements is depicted in 7.6, and the error rate of the system in the presence of the algorithm is described by 7.14. Finally, Chapter 8 contains a few-measurement algorithm which provides a threshold for the toric code using asymptotically fewer measurements than any previous code, and is the work of Freeman et al. [56]. The finite temperature crossover to the regime where the algorithm functions is depicted in 8.2. Crucially, the measurement density can be any fixed constant fraction, so long as temperature is sufficiently low. Chapter 9 contains summaries of a variety of side, non-toric-code projects I contributed to over the years, and Chapter 10 contains spillover discussions and musings on the future.

1.2 High Level Background

While the remainder of the thesis spends several chapters building up the necessary tools used to understand the results presented in Chapters 7, 8, and 9, this short introductory section will serve as a high level, “expert” overview of the historical place and motivation of this work.

This thesis was primarily motivated by a series of papers, centered on efforts from the Whaley group, starting in the early 2010s. Owing to the growing popularity of laser trap systems, and the possibility that these experimental setups might’ve been available for experiments in the near term, the group focused on several different research thrusts involving different ways of emulating different quantum systems in varying “near term” architectures. The landscape at the time was far different—superconducting qubits were not quite as dominant, NMR systems were not scaling as quickly as some wanted, and ion traps/neutral atom traps looked to be on the cusp of supporting systems large enough for interesting physics. Topological systems weren’t exactly *new*, but the thought of probing them directly in emulated systems / numerically was attractive enough that many groups were trying to devise schemes for doing so.

Starting with [65], the goal was to implement a scheme for effectively generating a topologically ordered system. As an example, this “stroboscopic” scheme introduced in Ref. [65] was applied to the toric code, and they showed that, owing to the toric code’s resilience to unitary-type noise, that it could be simulated even if the scheme used to generate the emulated toric code was fairly noisy. Shortly thereafter, Ref. [137], using similar methods, introduced a scheme for coupling an emulated quantum system to a simulated bath. The

primary target for this paper was an optical lattice type experiment, where a second lattice could be interleaved with the system lattice, and used as a resource for optically driving the system to some effective temperature.

At the same time, a handful of groups were (and had been) investigating the thermal properties of the toric code. The earliest work of which I'm aware to explicitly point out the thermal breakdown of the toric code is a series of papers by Nussinov and Ortiz in the late 2000s[102, 101], which directly mapped the toric code onto the Ising model, thus demonstrating its thermal fragility. Contemporaneous with this work, Castelnovo and Chamon[23, 24] demonstrated that the topological entanglement entropy of the toric code retained some finite temperature resilience, passing through a finite temperature phase transition. These results contributed to a flurry of schemes to try and restore thermal order in the toric code through a variety of different mechanisms, like long-range interactions, disorder, and coupling to bosonic fields[29, 30, 69, 109, 82].

It was around this time that I picked up on a project within the group started by Dylan Gorman (when he was a senior undergraduate) to try and understand, dynamically, what was going on with the toric code, as the vast majority of studies before then had considered only equilibrium arguments. When I joined the Whaley group, it housed around 6 different Monte Carlo experts, thus it seemed somewhat natural to tackle a numerics-heavy project in my first year. This work eventually became [55]—i.e., Ch. 6 of this thesis—and the rest is history.

Chapter 2

Errors at zero temperature

if you can trust what we know, then we can be even more confident that n is prime.

King James Programming Tumblr

Traditionally, quantum error correcting codes are defined in a “zero temperature” setting. Qubits are imagined to be exposed to some noise source which produces errors of some type independently at random to each of the qubits. The procedure of introducing uncontrolled errors to the unitary dynamics of the Schrodinger equation invariably leads one to the von Neumann equation, and then to the theory of completely positive maps. In this section, I briefly review the theory of error maps, as well as the stabilizer formalism first introduced by Gottesman[59]. As a framing device for future chapters, I also work through the theory of the 1D Ising model reinterpreted as the “simplest” stabilizer code.

The theory of quantum error correction has a rich history with a tremendous number of different classes of error correcting algorithms. While this thesis focuses on stabilizer codes in particular, refer to [87] or [107] for a more complete survey of the breadth of alternative protocols.

2.1 From Schrodinger to von Neumann

Towards an bit-flip error map

To be able to consider something like a random bit-flip error on a quantum mechanical system, we will have to move beyond the Schrodinger equation, because Schrodinger dynamics are purely unitary. That is, the most general sort of time evolution available for a state vector for Schrodinger type dynamics is some evolution like

$$\psi(t) = U(t)\psi(0) \tag{2.1}$$

Where $U(t)$ is some possibly time dependent unitary operator. Usually, this U is the time evolution operator which depends on the Hamiltonian, H , governing that particular quantum system—i.e., $U = \exp(-iHt)$.

The primary difficulty with this sort of dynamics for the purposes of modeling errors, is that most errors are not unitary. While some errors arise from incorrectly applied unitary operators (through poor calibration, experimenter-error, or systematic effect), the vast majority arise from interactions with uncontrolled environments. Bits are flipped—atoms are lost from the trap—photons are absorbed—information is irretrievably scrambled, lost, or leaked. Such processes are not *reversible*, and the Schrodinger equation considers dynamics that are manifestly *reversible*. This suggests something of a puzzle for the fledgling quantum mechanics practitioner—if these sorts of errors are not unitary, but quantum mechanics is fully unitary (and is the underlying theory describing the world)—where do they come from?

While the world is fully unitary (as far as we can tell!), trying to describe it with a fully unitary model is often tremendously inconvenient, or even practically impossible. The full quantum state of all of the gas particles in a room, for example, is a perfectly well-defined object, but everyday experience suggests that one need not have full access to the full wavefunction of every gas particle in a room to be able to make predictions about those gas particles. That is to say, on scales relevant to the everyday living of humans, the reversible, unitary nature of quantum mechanics averages out, and leaves us with a world that does not appear reversible at all. This averaging—this coupling of particles to enormous numbers of extra degrees of freedom is intimately related to how the process of measurement *also* breaks the unitarity of quantum mechanics.

A great deal has been written about this process—i.e., the emergence of the classical world from the quantum mechanical one—and treating it further would quickly leave the scope of this thesis. For our purposes, it will suffice to consider the principled, mathematical way of “averaging” quantum mechanical behavior, which naturally leads to a notion of non-unitary dynamics not encapsulated by the bare Schrodinger equation.

First, we consider density matrices, ρ rather than state vectors:

$$\rho := |\psi\rangle\langle\psi| \quad (2.2)$$

The dynamics of density matrices follow immediately from the Schrodinger equation and its adjoint, and is called the von Neumann equation:

$$i\hbar\frac{\partial\rho}{\partial t} = [H, \rho] \quad (2.3)$$

where H is the Hamiltonian for the system, and $[\cdot, \cdot]$ is the commutator. This equation is formally solved by,

$$\rho(t) = e^{-iHt}\rho_0e^{iHt} \quad (2.4)$$

For a generic quantum state, this formalism at first appears only to wildly complicate the process of performing calculations in quantum mechanics. Instead of one ψ , now we have two! Worse still, these dynamics are *still* unitary.

The beauty of the density matrix formalism is in its ability to describe a more rich set of quantum states—states for which it is difficult to write down the corresponding state vector, $|\psi\rangle$.

Consider the density matrix,

$$\rho_m = \begin{pmatrix} 1/2 & 0 \\ 0 & 1/2 \end{pmatrix} \quad (2.5)$$

It's a short exercise to convince oneself that ρ_m cannot be formed by taking any $\psi = \alpha|0\rangle + \beta|1\rangle$ and forming the outer product $|\psi\rangle\langle\psi|$, for any α, β .

Even though it cannot be written as arising from a state vector, it doesn't mean such states don't exist, it simply means some interpretation is in order. States like ρ_m are the natural consequence of having uncertainty about a quantum state. Such a state is the natural description of a situation where we have an equal, incoherent admixture of $|0\rangle$ and $|1\rangle$ states in some well isolated box. It is likewise the natural description of a single quantum state that has undergone some uncontrolled interaction with certain types of environments. This density matrix is also special in that it is, in a fairly precise mathematical sense, a state of maximal uncertainty about the quantum system of interest.

We can easily arrive at these sorts of states *from* everyday normal state vectors through the operation of *partial trace*. This operation will also take us beyond purely unitary dynamics, and into the realm of irreversibility.

First, we subdivide a quantum system into tensor products, $|\psi\rangle_A |\phi\rangle_B$. Then, the trace over system B is defined as:

$$Tr_B(|\psi\rangle_A |\phi\rangle_B \langle\psi|_A \langle\phi|_B) := \sum_i \langle\chi|_i |\psi\rangle_A |\phi\rangle_B \langle\psi|_A \langle\phi|_B |\chi\rangle_i \quad (2.6)$$

$$= \sum_k \alpha_k |\lambda\rangle_k \langle\lambda|_k \quad (2.7)$$

where $\{|\chi\rangle_i\}$ span subsystem B and $\{|\lambda\rangle_i\}$ span subsystem A . The resulting state is a statistical admixture of states in system A . Systems A and B can be imagined as individual qubits in a Bell pair, a system qubit and a large environment, or even the left and right halves of spins on a spin chain. All that is necessary to perform the subdivision is for the underlying quantum state to have the tensor product structure between the two subsystems.

As a brief aside, it is straightforward to verify that $Tr_B((|0_A 0_B\rangle + |1_A 1_B\rangle)(\langle 0_A 0_B| + \langle 1_A 1_B|))$ results in ρ_m . It is no coincidence that performing this averaging operation on an entangled state results in such an extremely “mixed” state. A reoccurring theme of this thesis will be the idea that interactions give rise to entanglements, and tracing out such entanglements causes concomitant losses of quantum information.

2.2 From Error Maps to Unitaries

The cleanest way to treat errors in a quantum mechanical system is by way of this trace—we start with a fully quantum mechanical system comprised of a system and a bath, and then we formally trace out the bath degrees of freedom. This is the natural procedure when one has an experimental apparatus, and a good sense of the sorts of quantum mechanical interactions that are occurring within the system of interest.

However, it's sometimes more operationally useful to start with a specific error model in mind, and then derive the specific system-bath Hamiltonian that gives rise to that particular error model. So, for concreteness and simplicity, we will consider a bit-flip type error dynamics on the density matrix for our system, and then work backwards to derive the system-bath interactions which gives rise to this system dynamics.

For a simple error model, it's convenient to use the Kraus operator representation of the error map,

$$\begin{aligned}\rho_{t+1} &= \sum_i E_i \rho_t E_i^\dagger \\ &= \mathcal{E}(\rho_t)\end{aligned}\tag{2.8}$$

for a system density matrix, ρ , and for error operators—also called Kraus operators—satisfying $\sum_i E_i^\dagger E_i = I$. 2.8 can be taken as a *definition* for the sorts of transformations that can occur to a system density matrix after having traced out another subsystem. Such transformations are called *quantum channels*, or, in the mathematical literature, trace preserving completely positive maps (TPCP maps). Of salience for this thesis, these maps are linear, preserve hermiticity ($\rho = \rho^\dagger$), preserve trace ($\text{Tr}(\rho) = \text{Tr}(\mathcal{E}(\rho))$), and preserve positivity (eigenvalues remain non-negative). For a bit flip channel, the Kraus operators are.

$$E_0 = \sqrt{1-p} I, E_1 = \sqrt{p} \sigma_x\tag{2.9}$$

Intuitively, E_0 encodes an identity operation from the environment with probability $1-p$, and E_1 encodes a bit flip error with probability p . These probabilities appear under radicals because the error operators appear in pairs in 2.8. Again, these E_i could arise from many different system-bath interactions, but written this way, only their effect on the system is retained. We can arrive at one possible system-bath interaction providing these dynamics by using the Stinespring dilation theorem. For a set of Kraus operators, E_i , and the system states on which they act, $|\psi\rangle$, we can promote the map in 2.8 to an isometry by the identification:

$$U_{S \rightarrow S'E} : U |\psi\rangle |0\rangle_E \equiv \sum_i E_i |\psi\rangle \otimes |i\rangle\tag{2.10}$$

where the environmental states $|i\rangle$ can be thought of as recording the action of the Kraus operators. This map is an isometry because it really only describes how to boost from the system Hilbert space S to the system + environment Hilbert space SE by way of adjoining a $|0\rangle_E$ state to $|\psi\rangle$, i.e.: $|\psi\rangle |0\rangle_E$. The evolution from an initial state like $|\psi\rangle |1\rangle_E$, for example,

is left unspecified, but is necessary for defining a properly unitary operator. Promoting this operator to a full unitary can be done by simply choosing the rest of the matrix elements so that U is unitary[98]. These environmental states $|i\rangle$ might seem mysterious, but they represent exactly the quantum states that are necessarily averaged over to produce the behavior in 2.8. Alternatively, this can be thought of as a *definition* of the Kraus operators:

$$E_i = \langle i|_E U_{S \rightarrow S'E} |0\rangle_E \quad (2.11)$$

where $\langle i|_E$ indexes some orthogonal set of environment states, and U is the underlying isometry acting between the system and the environment, ultimately giving rise to the error dynamics in the system of interest.

Returning to the bit-flip dynamics, using 2.10, we have:

$$U_{S \rightarrow S'E} : U |\psi_S\rangle |0\rangle_E \equiv \sqrt{1-p} I_S |\psi_S\rangle \otimes |0\rangle_E + \sqrt{p} \sigma_x |\psi_S\rangle \otimes |1\rangle_E \quad (2.12)$$

where the environmental state $|0\rangle_E$ records an “identity” operation by the bath, and where the bath state $|1\rangle_E$ records a bit-flip error. Again, this map is *almost* a unitary. One possible unitary matrix representation of U is:

$$U_{bit \ flip} \equiv \begin{pmatrix} \sqrt{1-p} & 0 & \sqrt{p} & 0 \\ 0 & \sqrt{1-p} & 0 & \sqrt{p} \\ 0 & \sqrt{p} & 0 & -\sqrt{1-p} \\ \sqrt{p} & 0 & -\sqrt{1-p} & 0 \end{pmatrix} \quad (2.13)$$

This particular form comes from a more general procedure where one “stacks” the error operators E_i as the first block-column of the unitary, and then fills in the rest of the entries with the only constraint that it be unitary[98]. Note that the top left entries are exactly E_0 and the bottom left entries are E_1 . The other entries were chosen to ensure $U^\dagger U = I$.

After a great deal of matrix algebra, this unitary can be rewritten as arising from the interaction between a system qubit and an environment qubit under an effective Hamiltonian. That is,

$$U \equiv \exp(-iH_{eff}t) \quad (2.14)$$

Where H_{eff} takes the form:

$$\begin{aligned} H_{eff} \equiv & -\frac{\pi}{4} I \otimes I - \frac{\pi}{4} \sigma_x \otimes I \\ & + \frac{\pi\sqrt{p}}{4} I \otimes \sigma_x + \frac{\pi\sqrt{p}}{4} \sigma_x \otimes \sigma_x \\ & - \frac{\sqrt{p}}{2} I \otimes \sigma_y + \frac{\sqrt{p}}{2} \sigma_x \otimes \sigma_y \\ & + \frac{\pi\sqrt{1-p}}{4} I \otimes \sigma_z + \frac{\pi\sqrt{1-p}}{4} \sigma_x \otimes \sigma_z \end{aligned} \quad (2.15)$$

and where our only approximation made is that p is small. It is well known how to engineer such Hamiltonians in a variety of systems. More transparently, this interaction factorizes:

$$H_{eff} = \frac{\pi}{4} I \otimes (-I + \sqrt{p}\sigma_x - (2\sqrt{p}/\pi)\sigma_y + \sqrt{1-p}\sigma_z) + \frac{\pi}{4} \sigma_x \otimes (-I + \sqrt{p}\sigma_x + (2\sqrt{p}/\pi)\sigma_y + \sqrt{1-p}\sigma_z) \quad (2.16)$$

which is the sum of a local field on the environmental qubit, and an entangling interaction with the system qubit.

While it can be illuminating to have the interaction in this more physically motivated form, it's ultimately more convenient to work with the quantum channel representation of the bit flip error (i.e., 2.8).

Before proceeding, it's important to clarify an apparent paradox. We began by saying that bit-flip type error processes were not easily captured by unitary dynamics, and wrote down an error map in the more general formalism of density matrices that we claimed *could* capture these processes. Then, we immediately used Stinespring's theorem to write down a unitary that we claimed gave rise to the bit-flip type error process.

So, can bit-flip errors be captured with unitary dynamics or not?

The apparent paradox resolves when considering the ramifications of having access to complete information of the environmental qubit. If one knows the complete state $|\psi\rangle|\phi_E\rangle$, then there is no error process—the system happily evolves under a unitary evolution. The full quantum state can be perfectly reconstructed for all times. Only when the evolution of the environment is averaged over—when it is traced out—does the error process manifest. As we will see, the density matrix is known for all times, but a perfect quantum state cannot necessarily be reconstructed from a density matrix.

Thus, if a single qubit is interacting with an additional, uncontrolled qubit, the reduced dynamics of that first, single qubit will “look like” a qubit undergoing an evolution with uncontrolled bit flip errors acting on it.

Error Dynamics in the System

Having reviewed the procedure for zooming out of a particular quantum channel to see the unitary dynamics which give rise to it, for the remainder of this thesis, we will zoom back in and focus on the subsystem dynamics generated by that quantum channel—after all, the environment is not generally accessible to a quantum engineer.

Combining 2.8 and 2.9, and specializing to the initial density matrix,

$$\rho_{init} = \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix} \quad (2.17)$$

after n steps, the error map $\mathcal{E}^n(\rho_{init})$ yields,

$$\rho(n) = \begin{pmatrix} \frac{1}{2}(1 + (1 - 2p)^n) & 0 \\ 0 & \frac{1}{2}(1 - (1 - 2p)^n) \end{pmatrix} \quad (2.18)$$

This discrete picture, where the error process is imagined to occur with each individual application of the error map, \mathcal{E} , can be extended to a continuous picture by making the identification $p \rightarrow \Gamma t/n$. In the limit $n \rightarrow \infty$, then the discrete error process becomes a continuous error process, where the bit flip errors occur with rate Γ . Equivalently,

$$\rho(t) = \begin{pmatrix} \frac{1}{2}(1 + e^{-2t\Gamma}) & 0 \\ 0 & \frac{1}{2}(1 - e^{-2t\Gamma}) \end{pmatrix} \quad (2.19)$$

A useful tool for diagnosing whether a state vector can be extracted from a given density matrix is *purity*, \mathcal{P} , and is defined,

$$\mathcal{P} \equiv \text{Tr}(\rho^2) \quad (2.20)$$

\mathcal{P} satisfies $1/d \leq \mathcal{P} \leq 1$, for system Hilbert space dimension d , and only equals 1 for density matrices that can be written as $|\psi\rangle\langle\psi|$ for some ψ . Conversely, the closer \mathcal{P} is to $1/d$, the closer the density matrix is to the totally mixed state—a state of maximal uncertainty about the quantum system of interest.

Recapping, the initial goal was to describe the effect of a bit-flip type error process on a qubit system. The Schrödinger equation by itself was not flexible enough, so we instead wrote down a quantum channel—an operation on *density matrices*—that *is* flexible enough to describe this error process.

Here, we see the system dynamics due to the presence of bit-flip errors are particularly disastrous—the system, if prepared initially in a 0 state, will exponentially decay into a state of maximal uncertainty—equally likely to be measured in either the 0 or 1 state, and saturating the lower bound for \mathcal{P} .

This dynamics arises exactly from tracing out—i.e., averaging over—the interaction of a system qubit with an environmental qubit, where the interaction Hamiltonian is of the form 2.16.

This exponential decay will be another recurring theme of this thesis. In a sentence, this thesis is an exercise in trying to correct bit-flip errors—in trying to avoid exactly this exponential decay in the information stored in the qubit. We will find that codes that are otherwise quite resilient to certain classes of errors will fall prey to these uncontrolled, continuous bit-flips in later chapters.

2.3 Stabilizer Codes

Definitions

With some understanding of how errors can be modeled in quantum mechanics, we're equipped to discuss the tools quantum information theorists have developed to correct them.

In this section, we briefly review the theory of stabilizer error correcting codes[59, 87]. These codes are but one in a veritable zoo of different schemes for storing and protecting information. We focus on them in this section primarily because stabilizer codes are theoretically tractable, and are the only sorts of codes studied in this thesis.

Given n qubits, a collection of operators S_i , and 2^k states $|\psi\rangle_i, i = 1, \dots, 2^k$ which span a subspace in which k encoded qubits are defined, let,

$$S_i |\psi\rangle_i = +1 |\psi\rangle_i \quad (2.21)$$

$$[S_i, S_j] = 0 \quad (2.22)$$

for all i, j . Furthermore, suppose there are m error operators $E_j, j = 1, \dots, m$, and that for each of them, there exists some not necessarily unique operator S_j such that

$$\{E_i, S_j\} = 0 \quad (2.23)$$

Stabilizer codes are those collections of states $|\psi\rangle_i$ and operators $\{S_j\}$ which satisfy the above conditions for a collection of error operators $\{E_k\}$, usually belonging to some subset of the Pauli group—tensor products of Pauli operators with the identity.

Condition 7.1 demands that the encoded states, $|\psi\rangle$ are all +1 eigenstates of the *stabilizers*, S_i . Intuitively, this means that measuring stabilizers does not disturb encoded states.

Condition 7.2 ensures that all stabilizers mutually commute. This means that it does not matter what *order* we measure stabilizers in—they can be measured simultaneously, swept, randomly sampled etc.

Finally, 7.3, while the most opaque, ensures that *error states*, or states that are perturbed from encoded states by the application of some number of error operators—i.e., $E_i E_j \dots E_k |\psi\rangle_l$ —are related to -1 eigenstates of some stabilizer operators. There may exist some sequence of errors $E_i E_j \dots E_k$ which can cause a transition between encoded states. That is to say, $E_i E_j \dots E_k |\psi_0\rangle = |\psi_1\rangle$ can occur, but there will be some minimum number of errors necessary to cause such a transition.

All together, these three conditions provide a set of operators, S_i , that can be measured in any order, which do not disturb the encoded states, and which provide some information about the presence of errors. Depending on what information they provide, errors can sometimes be corrected perfectly.

For example, given three qubits, let $|\psi\rangle_1 = |\uparrow\uparrow\uparrow\rangle$ and $|\psi\rangle_2 = |\downarrow\downarrow\downarrow\rangle$. Then the set of operators S_i satisfying (7.1) and (7.2) is $\{\sigma_z \sigma_z I, I \sigma_z \sigma_z\}$. One can easily determine that the set of error operators E_i corresponding to these two *stabilizer* operators is:

$$\{\sigma_x II, I \sigma_x I, II \sigma_x, \sigma_x \sigma_x I, \sigma_x I \sigma_x, I \sigma_x \sigma_x\}. \quad (2.24)$$

More transparently, this 3-qubit stabilizer code encodes two protected states. If some noise source were to apply any single qubit σ_x operator, or any two-qubit $\sigma_x^i \sigma_x^j$ operator,

measurement of the set of stabilizer operators would indicate the presence of an error. Furthermore, the code can actually detect *and* correct single σ_x errors. For example, a measurement result of $-1, +1$ of the stabilizers $\sigma_z \sigma_z I$ and $I \sigma_z \sigma_z$, respectively, indicates either an error on the first qubit or two errors on the latter two qubits. In a sufficiently noisy environment, these two errors would be indistinguishable—i.e., degenerate—, but for many noise models, the single error situation is much more likely, thus a single σ_x operator applied to the first qubit will more often than not return the qubit back into the protected subspace.

This entire procedure easily generalizes to an arbitrarily long chain of spins. In fact, by construction, the protected states of such a chain of L spins exactly coincide with the ground states of a 1D quantum Ising model.

2.4 Perfect decoding of the Ising model

The Ising model is the prototypical spin system considered by both classical and quantum statistical physics. We will encounter several flavors of Ising systems in this thesis, but the simplest is the 1-dimensional model, defined by the Hamiltonian on a 1-dimensional array of spins with periodic boundary:

$$H = \sum_i -J \sigma_i^z \sigma_{i+1}^z \quad (2.25)$$

For $J > 0$, the ground state is two-fold degenerate, and thus can encode a single qubit by the identification $|00000\dots\rangle \rightarrow |\psi\rangle_0$ and $|11111\dots\rangle \rightarrow |\psi\rangle_1$.

Now, consider the spin chain initialized to the $|\psi\rangle_0$ state, and further suppose some pattern of bit flip errors befalls the system—i.e., σ^x are applied with some probability to each site, independently at random.

In the classical domain, “decoding” for the 1D Ising model is straightforward: one simply measures the chain of spins, and flips those spins which are not pointing the majority direction. If on every error detection cycle, a bit flip error occurs with probability p independently at random on each site, then the probability that an error has occurred is equivalent to the sum of the probabilities of all the configurations with more than half of the spins flipped. This grows like:

$$P_{bit\,flip\,error} = \sum_{i>L/2} \binom{L}{i} p^i (1-p)^{L-i} \quad (2.26)$$

$$\leq e^{-\frac{1}{2p} L(p-1/2)^2} \quad (2.27)$$

where the second inequality follows from Chernoff’s bound[28]. This bound can be made tighter, but the relevant dependence is the exponential suppression in the chance of a bit flip error in L .

The quantum analogue is similar. One could measure the σ_z projection of the individual spins (via coupling to an ancilla, for example), as in the classical case, but this would, generically, collapse superposition states, resulting in a loss of quantum information.

Instead of accessing the projections of individual spins, one can preserve the quantum information by measuring the stabilizers, S_i , which indicate the presence of domain walls—virtual objects indicating a disagreement between neighboring spins. A “majority rule” update heuristic is equivalent to merging pairs of domains walls via a perfect matching procedure. This is achieved by adding fictitious labels to spins $+$ and $-$, where the label alternates after the presence of any domain wall. Domain walls are then joined so that the dominant label becomes the only label. I’ve provided a cartoon of this in Fig. 2.4.

While this procedure protects both classical and quantum Ising chains from bit-flip errors, it does not preserve *all* types of quantum mechanical errors. For this, we must turn to a more complicated code.

Chapter 3

The Toric Code

there is a season, and a time to avoid exhausting memory.

King James Programming Tumblr

In the previous chapter, we saw how uncontrolled unitary interactions brought about bit-flip errors in quantum mechanical systems. In generality, quantum mechanical systems are subject to many more types of errors. Ignoring errors due to the literal loss of a qubit—e.g., an atom escaping a laser trap—the full set of errors that any full quantum error correcting code must handle are Pauli-errors: σ_x , σ_y , and σ_z .

Peter Shor’s 9-qubit code was the first example of an error correcting code that could successfully detect and correct the full set of Pauli errors. Shortly thereafter, an explosion of code families erupted throughout the late 90s and early 2000s, all with different tradeoffs in implementability, code distance (i.e., the number of errors necessary to cause an uncorrectable error), and noise threshold.

Essentially all of these proposals were for systems that needed to be actively measured, and then corrected with some form of feedback to remain in a protected state. Kitaev’s Toric Code, first constructed in 1997, was the first example of a candidate *self-correcting* code—an error correcting code that seemed to be intrinsically resilient to errors, in much the same way that classical hard drives are resilient to noise because of the stability of the ordered phase of ferromagnetic materials.

In this chapter, we review some of the primary features of the toric code.

3.1 Preliminaries

Because the toric code can be analyzed as both a stabilizer error correcting code *and* as a Hamiltonian, to avoid confusion, we will specialize to the Hamiltonian case for the majority of this chapter. That is, we will imagine a system of qubits whose evolution is governed by the toric code Hamiltonian. We will comment on the stabilizer error correcting code version of the toric code near the end of this chapter in Sec. 3.6.

The toric code is defined on a square lattice of qubits, with qubit degrees of freedom residing on the edges of the lattice. The Hamiltonian is a sum over “stars” and “plaquettes”:

$$H_{\text{TC}} = -J_e \sum_v A_v - J_m \sum_p B_p, \quad (3.1)$$

$$A_v \equiv \prod_{j \in v} \sigma_j^z, \quad B_p \equiv \prod_{j \in p} \sigma_j^x, \quad (3.2)$$

where the A_v operator is defined on every cluster of four qubits sharing a vertex, and where the B_p operator is defined on every cluster of four qubits around a square, or plaquette. These are the stabilizers of the toric code. The coupling constants J_e and J_m are taken to be positive. Conveniently, the A_v and B_p commute with each other, and can often be analyzed separately.

The finite, planar version of the toric code (as a stabilizer code) called the *surface code* has become popular in the superconducting qubit community[50], but for this thesis, we specialize to 2D periodic boundary conditions.

3.2 Ground and excited states

With periodic boundaries, there are four degenerate ground states in the toric code, thus, independent of lattice dimension, the ground state subspace of the toric code encodes 2 qubits. The four-fold degeneracy of the ground state can be understood from the following argument:

Without loss of generality, suppose that the toric code is in some ground state, call it $|\psi\rangle_{00}$. Compare this state to a copy of the ground state modified by the additional application of a single horizontal line of flipped bits (i.e., a loop of σ_x operators winding horizontally around the lattice). Call this second state, with a line of flipped bits $|\psi\rangle_{10}$.

It is straightforward to verify that $|\psi\rangle_{10}$ is also +1 eigenstate of every A_v and B_p , so long as $|\psi\rangle_{00}$ is, thus $|\psi\rangle_{10}$ is also a ground state. It is slightly more involved to show that $|\psi\rangle_{10}$ and $|\psi\rangle_{00}$ are orthogonal.

We proceed by introducing the *winding operators*:

$$W_{1,2}^x \equiv \prod_{j \in \Gamma_{1,2}} \sigma_j^x, \quad W_{1,2}^z \equiv \prod_{j \in \tilde{\Gamma}_{1,2}} \sigma_j^z, \quad (3.3)$$

where $\Gamma_{1,2}$ represents some closed path of qubits wrapping entirely around the lattice, and where $\tilde{\Gamma}_{1,2}$ is a closed path along the dual lattice. For a pictorial representation of a piece of the toric code graph and its dual, see Fig 3.1. Paths on the lattice are defined by a sequence of edges between vertices, and paths on the dual lattice are defined by a sequence of edges between centers of plaquettes. Simultaneously rotating all edges on a pictorial graph of the

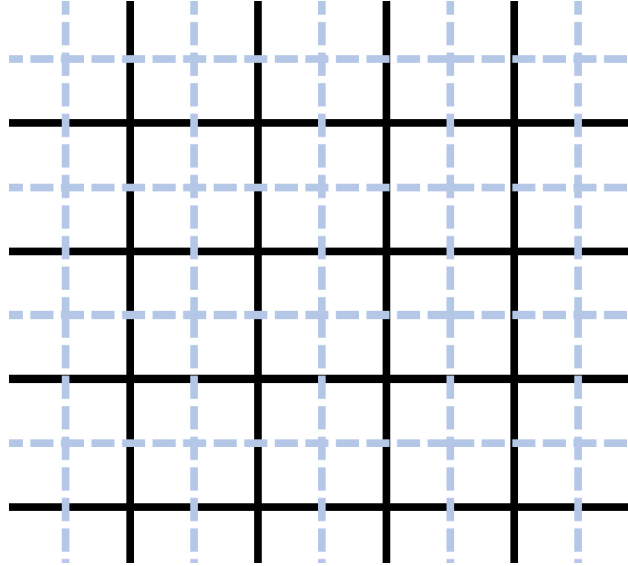


Figure 3.1: A piece of square toric code lattice in solid black lines, with its dual lattice indicated in light blue dotted lines. Vertices of the dual lattice rest on plaquettes of the primary lattice, and vice versa. Note that by individually rotating the edges of the black lattice, one retrieves the dual.

toric code exchanges the lattice with its dual. Note that the winding operators mutually commute with H_{TC} , thus they are simultaneously diagonalizable.

Fix $\tilde{\Gamma}_2$ to be a closed loop of σ_z operators running along some fixed vertical column of plaquettes—i.e., a closed loop around the dual lattice. Again, W_2^z commutes with H_{TC} , and, because it is composed entirely of paulis, has eigenvalues ± 1 .

Now, by choice, we can associate $|\psi\rangle_{00}$ with the $+1$ eigenvalue of W_2^z , and $|\psi\rangle_{10}$ with the -1 eigenvalue. Because eigenstates of symmetric operators with distinct eigenvalues are orthogonal, these two states thus must be orthogonal.

To see that there must be four total such states and no more, we resort to a counting argument. For a square lattice with L vertices on an edge, there are L^2 unique A_s operators, and L^2 unique B_p operators. The ground state is the simultaneous $+1$ eigenstate of these operators, and excited states violate some number of terms in the hamiltonian, yielding -1 measurements on some subset of the stabilizers. We can then count the total number of states on the toric code by indexing the states based on which ground state we start in, and by which stabilizers are being violated.

While there are L^2 unique A_s and B_p operators, respectively, only $L^2 - 1$ of them, each, are independent. Thus, there are $2^{L^2-1} \cdot 2^{L^2-1} = 2^{2L^2-2}$ independent patterns of stabilizer outcomes for A_s and B_p , total. However, there are 2^{2L^2} total states in the Hilbert space defined on the lattice. Thus, the full Hilbert space is covered by considering the $4 \cdot 2^{2L^2-2} = 2^{2L^2}$ states arising from all possible combinations of stabilizer outcomes associated

with the four different ground states.

One might worry that some of these combinations of stabilizers might be double counted—i.e., that some combination of B_p violations in one ground state corresponds to the same state as some combination of B_p violations in another ground state. This cannot happen for the same reason that ground states are orthogonal—the winding operator construction guarantees that these states are always orthogonal.

Intuitively, the four ground states can be best understood by how they differ from each other. All four ground states are fairly complicated superpositions of “loops” of spin states. If spin-down is zero, and spin-up is one, then a plaquette of ones is the simplest of such loops. Transitioning from one ground state to another involves applying a closed loop of σ_x operators winding entirely around the system (or of σ_z operators winding around the dual lattice). Because the system has the topology of a torus, there are 2 topologically distinct ways in which one can draw such a loop.

The winding operators defined above can only discern the *parity* of the number of such loops in a given toric code ground state. Thus, the ground states can be organized into the four combinations of even, odd \times even, odd parities of loops winding around the system horizontally and vertically.

In fact, to a *local* observer, the four ground states of the toric code are completely indistinguishable. This is a manifestation of the so-called *topological order* of the toric code. One must perform a measurement that interrogates the toric code entirely along the length of the system to determine which state the system is in at any given time.

3.3 Construction of Ground State

In the previous section, we derived some properties of the ground state manifold that must be true, but did not actually provide the state itself. There are several ways to construct the ground state of the toric code, with varying levels of completeness. Here, we provide an intuitive algorithmic procedure by which one can exactly reconstruct the ground state, because the details of the exact wavefunction in the basis of spins are fairly involved. We also provide a simple PEPS procedure for generating the ground state, which is sometimes numerically convenient. The most convenient representation for the ground states for this thesis does not use either of these representations, and is treated in the following section.

Starting with a lattice of spins all initialized to the 0, or down state, the following algorithm generates the state:

In words, this algorithm starts with a state that is initialized to be all 0s. Then, every unique state that differs from this original state by some number of applications of B_p on different plaquettes are added to a collection C . One ground state is the equal superposition of all such states in C .

The other three ground states can be generated as follows: instead of starting with $|\psi\rangle$ initialized to all zero, instead initialize to all zeros with a column of flipped bits, a row of flipped bits, or both a column and row of flipped bits, respectively.

Algorithm 1 Ground State Construction

```

1:  $|\psi\rangle \leftarrow$  All spins initialized to zero (or down)
2:  $C \leftarrow \{|\psi\rangle\}$ 
3: while  $\|C\| \leq 2^{2L^2-1}$  do
4:   for  $b$  in  $\{B_p\}$  do
5:     for  $|\phi\rangle$  in  $C$  do
6:       if  $b|\phi\rangle \notin C$  then Append  $b|\phi\rangle$  to  $C$ 
7: Return  $\frac{1}{\sqrt{\|C\|}} \sum_{|\phi\rangle \in C} |\phi\rangle$ 

```

These four choices (all zeros, 1 horizontal row of flipped bits, 1 vertical row of flipped bits, both 1 horizontal and 1 vertical row of flipped bits) are not unique. Again, by the properties of the winding operators, any choice consistent with the *parity* of the number of either rows or columns of flipped bits will result in the same generated ground state. For example, by starting with 2 rows of flipped bits and 2 columns of flipped bits, the same ground state is generated as by starting with all bits as zero.

This procedure is exactly equivalent to forming the projector

$$\prod_s (1 + A_s) |\psi\rangle \quad (3.4)$$

for the different initial $|\psi\rangle$ as described.

3.4 Quasiparticles

The previous picture of the ground state is instructive for understanding the “loop gas” picture of the toric code. Note that applications of B_p operators essentially form closed loops of spins on the lattice. Thus, the ground states can be understood as equal superpositions of every possible combination of closed loops of spins on the lattice.

But this picture is extremely inconvenient for thinking about dynamics, because one must keep track of the details of exponentially many states. It is instead much more intuitively clear to think of the different ground states of the toric code as orthogonal vacua—empty states.

Excited states, then, can be easily tracked by monitoring how the stabilizers respond to different types of excitations. It is straightforward to verify that applying any continuous, open string of σ_x operators to a ground state results in a state that violates at most $2 A_s$ stabilizers, with total energy $4J_e$ above the ground state. Further, any continuous, open string of σ_z operators applied to a ground state violates at most $2 B_p$ stabilizers, with total energy $4J_m$ above the ground state energy. These two types of excitations are sometimes called *electric* and *magnetic* quasiparticles—language that derives from interpreting the toric code as a Z_2 lattice gauge theory. They are considered “particles” because the only data relevant for the dynamics of the toric code is the location of the *endpoints* of the string of errors.

Finally, σ_y type errors can always be rewritten as combinations of σ_z and σ_x errors by the identity:

$$\sigma_a \sigma_b = i \epsilon_{abc} \sigma_c + \delta_{ab} I \quad (3.5)$$

for $a = x$, $b = z$, and $c = y$, which follows immediately from the commutation and anti-commutation relations governing the pauli group.

Thus, excited states of the toric code are characterized by the number of open ended strings of excitations. These excitations always come in pairs—note that even a single σ_x or σ_z will always violate 2 neighboring stabilizers. It is sometimes convenient, then, to think of the quasiparticles as “residing” on either the stars or plaquettes of the lattice, as a shorthand for indicating which stabilizer are being violated by a string of errors that starts on one stabilizer and ends on the other. With a record of which ground state a system was initialized to, and with a record of where quasiparticles reside, one *almost* has a complete specification of the state of the toric code. If one has access to the actual errors applied, then this fully specifies the state. In reality, the strings of applied errors are not actual observables in the toric code—only the endpoints are detectable. Additionally, the *parity* of the error string—whether it winds left to right/top to bottom, or wraps around the lattice—can be detected by a reference winding operator. Thus, measurements of the two winding operators and measurements of stabilizers fully specify the state of the toric code.

It’s important to reemphasize that the actual pattern of errors in the code is not observable (or even detectable!)—only the endpoints, and the total parity of all error strings with respect to the winding operators. Note that by applying B_p operators to some fixed pattern of σ_x errors, the endpoints of strings of errors remain fixed, and the parity of the errors with respect to the winding operators remains fixed, but the pattern of errors changes. That the pattern appears to change is something of an artifact of the record-keeping scheme. This will be made explicit near the end of this chapter in Sec. 3.6 when we discuss error *correction* in the toric code.

3.5 Zero temperature resilience

That one must interrogate an entire edge’s worth of qubits to determine which ground state the system happens to be in is intimately relate do the resilience of the toric code to noise. It almost immediately implies that, to cause an error to a state encoded in the ground state subspace of the toric code, the environment must apply a sequence of errors that runs entirely along the axis of the toric code. In a unitary noise model, as discussed in Sec. 2.1, such errors are extremely unlikely. Consider a perturbation to the toric code Hamiltonian of the following form:

$$H_{\text{Noisy TC}} = -J_e \sum_v A_v - J_m \sum_p B_p + \eta \sum_i \sigma_i^z \quad (3.6)$$

This Hamiltonian is meant to model a toric code in the presence of some noisy field of strength η . Note that, if η is perturbatively small—i.e., that $\eta \ll J_{m,p}$ —then this perturbation is exactly zero up to L^{th} order in perturbation theory, for lattice edge length L . This is because the only nonzero matrix elements of the perturbation require an L -fold interaction with the perturbing field—i.e., terms like,

$$\text{Nonzero contributions} \sim \langle \psi |_{00} \sigma_1^z \sigma_2^z \dots \sigma_L^z | \psi \rangle_{01} \quad (3.7)$$

In other words, unitary noise is exponentially suppressed in L , so long as it is sufficiently small compared to the toric code Hamiltonian energy scale.

Surprisingly, η must grow to be a significant fraction of the Hamiltonian energy scale before any appreciable splitting appears in the ground state manifold. This splitting as a function of noise strength is depicted in Fig. 3.2. There, the on-site field must be more than half the strength of the Hamiltonian before there is any discernible effect, after which there is a sharp phase transition into a symmetry-broken phase.

This resilience to noise was the primary reason that people expected the toric code to be a powerful quantum memory.

3.6 Decoders

So far, we have seen that the ground state subspace of the toric code possesses strong robustness to *unitary* noise, and that this robustness is related to the extensively long strings of errors necessary for transitions to occur. But the toric code is also a *stabilizer error correcting code*, with stabilizers $S = \{A_s, B_p\}$, code states $|\psi\rangle_{ij}$, $i, j \in \{00, 01, 10, 11\}$, and errors in the pauli group. Thus, there exist efficient algorithms whereby, conditioned on the outcomes of stabilizer measurements, one can return the toric code to an encoded state with very high probability.

The details of the decoding procedure depend *strongly* on the type of error model used. For σ_x and σ_z errors applied independently at random with probability p on each edge, decoders will succeed with high probability so long as $p \lesssim .109$. If one additionally introduces noisy *measurements*, but keeps the same error model per site, this *threshold* drops down to $p \lesssim .029$ [96]. Further decoding algorithm details or geometric constraints on allowed operations (e.g., nearest-neighbor only) further constrain the threshold [113].

If the error probability is higher than the threshold, then the system crosses a percolation/correctability phase transition, where it becomes likely that an extensive error string will exist somewhere in the system. Such a string is *not* correctable, because it isn't detectable by stabilizers. Fig. 3.3 depicts such an uncorrectable error for a numerically simulated system.

The vast majority of these decoders work within this “infinite temperature” limit, wherein errors occur independently at random on each edge. We can quantify the probability of some pattern of errors based on the following formula:

$$P(\gamma) = p^{\|\gamma\|} (1 - p)^{N - \|\gamma\|} \quad (3.8)$$

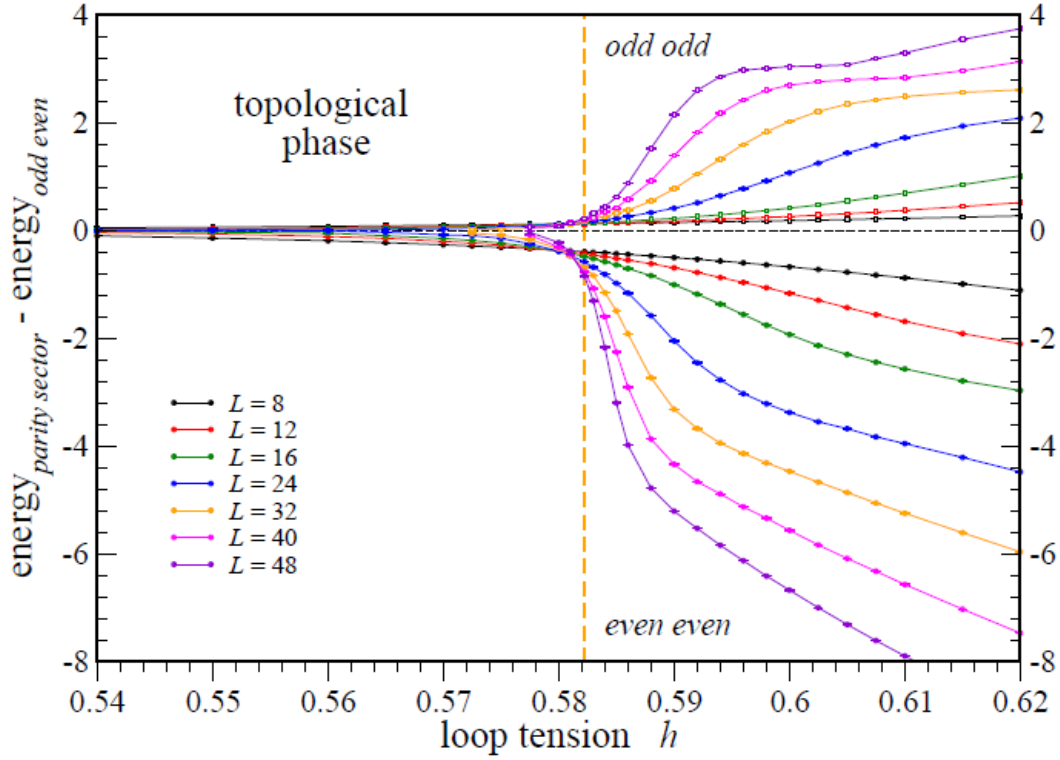
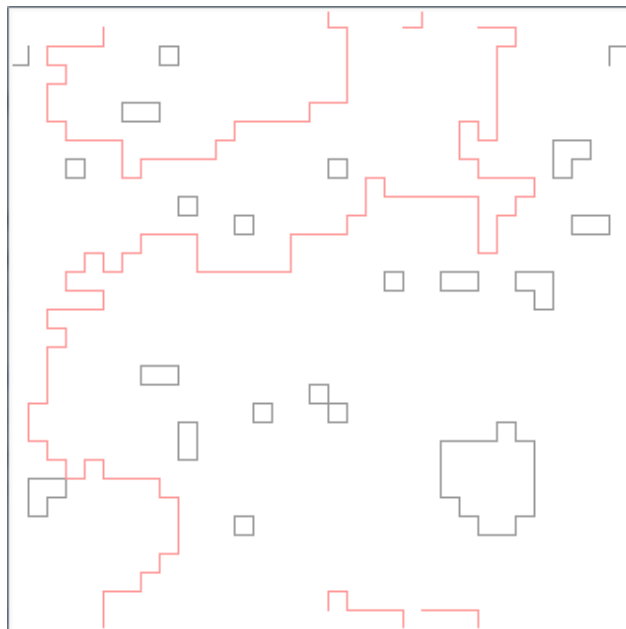


Figure 3.2: Ground state splitting in the toric code as a function of applied field strength. Adapted from Trebst et al. <https://arxiv.org/pdf/cond-mat/0609048.pdf>. Our η corresponds to their “loop tension”, h .

where γ is some error pattern, $\|\cdot\|$ is the number of errors in that error pattern, and N is the total number of qubits on the lattice.

In general, there are many possible error patterns consistent with any given record of stabilizers. A powerful error correcting heuristic is to consider the *minimum weight* error pattern—i.e., the error pattern with the smallest $\|\gamma\|$ consistent with some measurement of stabilizers, as this is a proxy for the most likely *actual* pattern of errors present in the system.

This pattern of errors does not necessarily coincide with the real pattern of errors in the system, nor does it necessarily guarantee that correcting this error will always actually perform the right corrective operation. The ultimate goal is to perform the most likely operation to achieve error correction—not simply reverse the most likely pattern of errors (these are subtly different quantities). Finding the most likely operation to correct the code involves summing over all possible patterns of errors consistent with a given record of



stabilizers. Fortunately, using the most likely pattern of errors (rather than the most likely corrective operation) as a proxy only reduces the threshold by a small amount, and can be calculated efficiently (whereas the full algorithm for determining the most likely corrective action has poor scaling).

The most likely pattern of errors, then, is easily found as follows: construct a completely connected graph, one independent graph for each type of stabilizer, with each node given by the location of a quasiparticle (i.e., a -1 stabilizer measurement). Edges are defined as the minimum edge-wise distances between pairs of quasiparticles of the same type. Edmond’s algorithm provides an efficient (i.e., polynomial time) algorithm for finding the most likely error, because the most likely error corresponds to the *minimum weight perfect matching* of the so-defined graph—a widely studied graph theoretic object[38].

Thus, in the presence of this type of noise, measurements, decoding, and corrective operations can protect information in the toric code.

3.7 Zero temperature mapping to RBIM

Establishing that the toric code can be corrected with high probability (i.e., 1 in the thermodynamic limit) below the error threshold relies on a statistical physics mapping to the Random Bond Ising Model (RBIM)[86]. The probability of uncorrectable errors appearing at $p \approx .109$ can be related to the fluctuation of domains of spins in a corresponding RBIM at certain critical temperature. In this section, we will sketch this mapping.

The RBIM is defined on a square lattice, with (classical) spin degrees of freedom living on the faces of the lattice. One could alternatively choose to place the spins on vertices, but identifying the spins with faces will be convenient for the toric code correspondence.

The form of the classical Hamiltonian is:

$$H_\tau(s) = - \sum_{\langle i,j \rangle} \tau_{ij} s_i s_j \quad (3.9)$$

where $\langle i, j \rangle$ is a sum over nearest neighbor spins, and $\tau_{ij} = \pm 1$, and equals -1 with probability p . This actually defines a *family* of statistical models, because the τ_{ij} do not fluctuate thermodynamically—they are fixed, and then the properties of the system are derived from fluctuations in the spin degrees of freedom. Sometimes, these such τ are referred to as quenched random variables[86] The partition function for this system is then,

$$Z(\beta, \tau) = \sum_{\{s\} \in S} e^{-\beta H_\tau(s)} \quad (3.10)$$

for all configurations of spins, S . This partition function sensitively depends on the values of p and β . The statistical mapping between the RBIM and the toric code is only valid when,

$$e^{-2\beta} = \frac{p}{1-p} \quad (3.11)$$

so, we specialize to this slice for the remainder of the section. This particular slice of the phase diagram is called the Nishimori line[114].

The precise details of the mapping are subtle, and rely on a fair amount of algebraic manipulation, so we refer the reader to Lidar and Brun [86] for a complete treatment. Intuitively, the presence of an error string can be related to the existence of an extensively large, fluctuating domain of spin variables in the RBIM.

To be precise, for a given pattern of errors on the toric code, one can also consider the corrective operation—itself a string of operators—that one would apply to return the toric code to its ground state. These are depicted in solid red and dotted red, respectively, in Fig. 3.4. For successful corrective operations, all error strings will be closed into topologically trivial (i.e., not wrapping around an entire axis) loops. The analogue of these loops in the

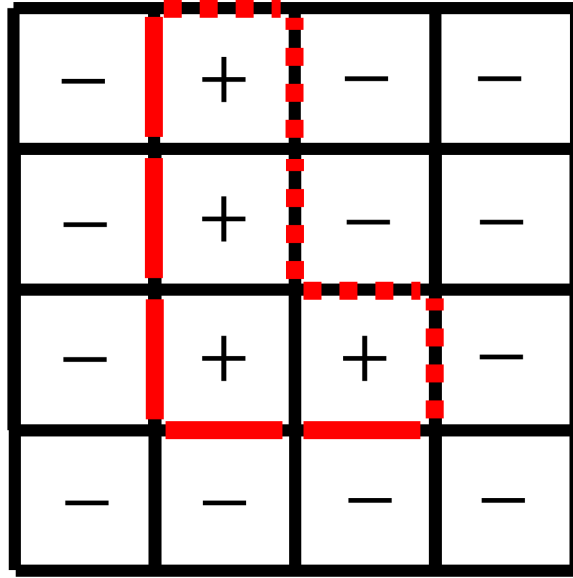


Figure 3.4: A small sublattice of the toric code with a string of errors (bold red), a corrective operation (dotted red), and auxiliary plaquette labels (plus and minus) indicating possible spin configurations on an equivalent RBIM lattice.

corresponding RBIM are domains of spin variables with differing sign. All adjacent spin variables with the same sign have a ferromagnetic coupling ($\tau_{ij} = 1$), and spin variables with disagreeing sign have antiferromagnetic coupling ($\tau_{i'j'} = -1$).

Thus, when the RBIM has large domains of disagreeing spins, this is analogous to a toric code system with long error strings. The correctable and uncorrectable phases of the toric code then correspond exactly to the ordered and disordered phases of the RBIM.

Note that the invariance of the error strings of the toric code to application of the B_p operator are recapitulated by the RBIM Hamiltonian as well. Applying a B_p to an error string is equivalent to flipping both a RBIM spin, s_k , and the sign of all of the coupling constants connecting it to its neighbors, τ_{kj} . This is a gauge symmetry of the RBIM Hamiltonian—i.e., this is a transformation that leaves the Hamiltonian fixed.

3.8 Finite temperature mapping to 1-D Ising Model

Thus far, we have seen that the toric code is resilient to unitary noise, and if stabilizers are measured and decoded with sufficient frequency for certain noise models, it can be efficiently error corrected.

In this section, we'll see that, despite this resilience, the toric code is *not* resilient to thermal noise.

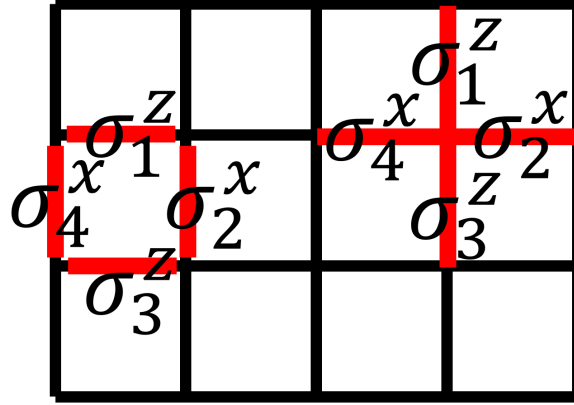


Figure 3.5: The transformed toric code lattice numbering convention for Wen's Plaquette model.

From the Toric Code to Wen's Plaquette Model

First, we apply the following unitary to the rows (sometimes called the even sublattice) of the toric code lattice (i.e., half of the total spins):

$$U_w = \prod_{k \in \text{even}} e^{i\pi \frac{\sqrt{2}}{4} (\sigma_k^z + \sigma_k^x)} \quad (3.12)$$

Rotating H_{TC} by this operator, we retrieve,

$$U_w^\dagger H_{\text{TC}} U_w = \sum_{s,p} -K \sigma_1^x \sigma_2^z \sigma_3^x \sigma_4^z \equiv H_W \quad (3.13)$$

where the sum runs over every star and plaquette, and where 1, 2, 3 and 4 follow the convention indicated in Fig. 3.5. This result follows readily from the identity,

$$e^{-i\vec{\sigma} \cdot \hat{n} \theta / 2} \vec{a} \cdot \vec{\sigma} e^{i\vec{\sigma} \cdot \hat{n} \theta / 2} = (\cos(\theta) \vec{a} + \sin(\theta) \hat{n} \times \vec{a}) \cdot \vec{\sigma} \quad (3.14)$$

That is, U_w maps H_{TC} into a model called Wen's Plaquette Model (WPM)[130]. As we have only performed a change of basis, WPM has the same spectrum as the toric code.

From Wen's Plaquette Model to the 1-D Ising Model

From here, it's only a matter of comparing partition functions derived from the WPM Hamiltonian and the Ising model Hamiltonian. Following Zhang, Kou, and Deng [140], first, let

$$w_i \equiv \sigma_{1i}^x \sigma_{2i}^z \sigma_{3i}^x \sigma_{4i}^z \quad (3.15)$$

The operator w_i clearly has eigenvalues ± 1 , and commutes with every other $w_{j \neq i}$. Thus, the partition function,

$$Z = \text{tr}(e^{-\beta H_W}) \quad (3.16)$$

factorizes as:

$$Z = \prod_k (4e^{-\beta K} + 4e^{+\beta K}) \propto \cosh(\beta K)^{2L^2-2} \quad (3.17)$$

where k runs from 1 to $2L^2 - 2$ —i.e., the number of independent terms in H_W . But this is exactly the partition function for the 1-D Ising model with no applied field (both quantum and classical) on $2L^2 - 2$ sites. While the 1-D Ising model has a *zero* temperature ordered phase, it does *not* have any finite temperature order-disorder transition. Thus, it is thermodynamically unstable at all finite temperatures.

Then, working backwards through the mapping, Wen's plaquette model does not have a finite temperature ordered phase, thus the toric code does not have a finite temperature ordered phase.

Chapter 4

Errors at finite temperature

ye may be able to extract the bare, untagged datum and send this to the environment.

King James Programming Tumblr

In the previous chapter, we saw a collection of properties of the toric code—its resilience to unitary noise, its error-correctability—but we also saw a very simple argument for why the toric code cannot be stable to noise at any finite temperature.

Up until this point, I’ve been loose with this notion of “temperature”, essentially only using it as a free variable in partition functions. The purpose of this chapter is to make “temperature” overwhelmingly precise, because it is not immediately obvious what “temperature” even means in a quantum mechanical setting. While it is straightforward to, for example, define and construct thermal density matrices like:

$$\rho_{\text{th}} \equiv e^{-\beta H} \tag{4.1}$$

and interpret them as “classical” quantum states, it does not provide any intuition for *why* one uses such states, or what at all they have to do with temperature.

Clarifying this intuition entails an overview of the Lindblad formalism, as well as a discussion of different types of environments, and how they give rise to something that can be identified as a temperature for quantum mechanical systems.

4.1 The Lindblad Equation - A Simple Discrete Derivation

We have already seen how tracing out the part of the system dynamics give rise to error dynamics in our system in 2.1. The Lindblad equation is the result of performing this tracing-out procedure with some additional assumptions about the sort of interaction occurring between the system and the environment, or bath.

First, we will write our total system + bath wavefunction as,

$$|\Psi\rangle = |\psi_s\rangle \otimes |\phi_b\rangle \quad (4.2)$$

This explicitly imposes a delineation of the total system into two components—the system qubits and the bath qubits. Next, we assume these two systems are interacting via some operator U that can be decomposed as

$$U \equiv \sum_j S_j \otimes B_j \quad (4.3)$$

We won't worry about the precise form of the S_j and B_j operators for now. After the interacting via this unitary, the system dynamics are,

$$\rho_s^1 = \text{Tr}_B(U |\Psi\rangle \langle\Psi| U) \quad (4.4)$$

$$= \sum_{j,j'} \text{Tr}_B((S_j |\psi_s\rangle \langle\psi_s| S_{j'}^\dagger) \otimes (B_j |\phi_b\rangle \langle\phi_b| B_{j'}^\dagger)) \quad (4.5)$$

$$= \sum_{j,j'} \langle\phi_b| B_{j'}^\dagger B_j |\phi_b\rangle S_j |\psi_s\rangle \langle\psi_s| S_j^\dagger \quad (4.6)$$

where between the second and third lines we have made use of the identity

$$\sum_k \langle k| O |\phi_b\rangle \langle\phi_b| O' |k\rangle = \sum_k \langle\phi_b| O' |k\rangle \langle k| O |\phi_b\rangle \quad (4.7)$$

$$= \langle\phi_b| O' O |\phi_b\rangle \quad (4.8)$$

because the trace runs over a complete set of states in the bath—i.e., $\sum_{k \in \text{bath}} |k\rangle \langle k| = I_{\text{Bath}}$. It will be convenient to work with the eigenvectors of the bath matrix elements, $\langle\phi_b| B_{j'}^\dagger B_j |\phi_b\rangle \equiv M_{jj'}$, so let μ_k and λ_k satisfy:

$$M\mu_k = \lambda_k\mu_k \quad (4.9)$$

Then, define the sums of *system* operators (weighted by the appropriate matrix elements from the bath):

$$A_K \equiv \sqrt{\lambda_k} \sum_j \mu_{kj} S_j \quad (4.10)$$

This set of operators $\{A_k\}$ are exactly the Kraus operators already encountered in Sec. 2.1. Thus, we can rewrite,

$$\begin{aligned}
 \rho_s^1 &= \sum_{j,j'} M_{jj'} S_j |\psi_s\rangle \langle \psi_s| S_{j'}^\dagger \\
 &= \sum_{j,j'} \left(\sum_k \lambda_k \mu_{kj} \mu_{kj'}^* \right) S_j |\psi_s\rangle \langle \psi_s| S_{j'}^\dagger \\
 &= \sum_{j,j'} \left(\sum_k \sqrt{\lambda_k} \mu_{kj} S_j |\psi_s\rangle \langle \psi_s| \sqrt{\lambda_k} \mu_{kj'}^* S_{j'}^\dagger \right) \\
 &= \sum_k (A_k |\psi_s\rangle \langle \psi_s| A_k^\dagger)
 \end{aligned} \tag{4.11}$$

where we have made use of the singular value decomposition of M into a product of the matrix formed by its eigenvectors, μ_k , a diagonal matrix of eigenvalues, λ_k , and finally the conjugate transpose matrix of eigenvectors, μ_k^* .

Eq. 4.11 is a fully general discrete master equation for the reduced dynamics of the system in the presence of some interaction with a bath via U . There are several lengthy derivations of the full Lindblad equation, which is a specialization to a particular limit approximation to this equation, in the literature [1, 104, 12]. Here, following Lidar and Brun [86] we provide a simple heuristic derivation for a perturbatively small interaction with a bath. This is the *weak coupling* limit. Further, we assume that, with each interaction with the bath, the bath is reset into its initial starting state. This is a simple way of imposing *Markovianity*—simply that the bath retains no memory of interactions with the system. This property emerges naturally in real bath models when there's a separation of timescales between interactions in the bath and interactions within the system. Let,

$$U = \sum_j S_j \otimes B_j \equiv e^{-i\epsilon \sum_k s_k \otimes b_k} \tag{4.12}$$

$$\approx I - i\epsilon \sum_k s_k \otimes b_k - \frac{(\epsilon)^2}{2} \left(\sum_k s_k \otimes b_k \right) \left(\sum_k s_k \otimes b_k \right) + O(\delta t^3) \tag{4.13}$$

Where terms of $O(\epsilon^2)$ are retained on the second line. Collecting these terms and replacing the S_k in Eq. 4.11 with the appropriate terms from Eq. 4.13, one retrieves

$$\begin{aligned}
 \rho'_s &= \sum_k A_k \rho_s A_k^\dagger \\
 &= \rho_s - i\epsilon \sum_k \langle \phi_b | b_k | \phi_b \rangle s_k \rho_s + i\epsilon \rho_s \sum_k \langle \phi_b | b_k | \phi_b \rangle s_k \\
 &\quad + \epsilon^2 \sum_{kk'} m_{kk'} s_k \rho_s s_{k'} \\
 &\quad - \frac{\epsilon}{2} \sum_{kk'} m_{kk'} s_k s_{k'} \rho_s - \frac{\epsilon}{2} \sum_{kk'} m_{kk'} \rho_s s_k s_{k'}
 \end{aligned} \tag{4.14}$$

where $m_{kk'} \equiv \langle \phi_b | b_{k'}^\dagger b_k | \phi_b \rangle$. After a small amount of rearranging, this becomes:

$$\rho'_s = \rho_s - i\epsilon [H_{\text{Lamb}}, \rho_s] + \sum_k \epsilon^2 (a_k \rho_s a_k^\dagger - \frac{1}{2} \{a_k^\dagger a_k, \rho_s\}) \tag{4.15}$$

where the a_k are defined in the same way as 4.10, and where $H_{\text{Lamb}} \equiv \sum_k \langle \phi_b | b_k | \phi_b \rangle s_k$, often called the Lamb Shift Hamiltonian. This nomenclature arises because the Lamb shift is similar in spirit to the Lamb shift in atomic spectra—it's a contribution to the dynamics that is purely unitary, and causes a shift to the energy levels of the system. Because we're often only interested in the non-unitary part of the dynamics being captured by the Lindblad equation, it is often useful to absorb this term into a change of basis, or to simply absorb it into the definition of the Hamiltonian. We will do the latter.

Remembering that this equation specifies the evolution for a single interaction with the bath over a timescale we choose, then we can define that time interval to be δt , and rearrange once more, dividing both sides through by that time interval and re-including the von Neumann equation contribution:

$$\frac{\rho'_s - \rho_s}{\delta t} \equiv \frac{\partial \rho_s}{\partial t} = -i[H_s, \rho_s] + \sum_k \frac{\epsilon^2}{\delta t} (a_k \rho_s a_k^\dagger - \frac{1}{2} \{a_k^\dagger a_k, \rho_s\}) \tag{4.16}$$

This is *almost* the Lindblad equation. Finally, making the identification $(\epsilon/\sqrt{\delta t})a_k \equiv L_k$, we retrieve:

$$\frac{\partial \rho_s}{\partial t} = -i[H_s, \rho_s] + \sum_k (L_k \rho_s L_k^\dagger - \frac{1}{2} \{L_k^\dagger L_k, \rho_s\}) \tag{4.17}$$

This equation describes the evolution of a system interacting weakly with a Markovian—i.e., memoryless—bath. We will make extensive use of this equation in later chapters to describe the evolution of the toric code in thermal contact with a reservoir. In particular, the rates $\frac{\epsilon}{\sqrt{\delta t}}$ will entirely determine the interesting thermal properties for stabilizer codes at finite temperature.

4.2 The Lindblad Equation - The Long Continuous Derivation

We still have yet to see a notion of “temperature”. In this section, we will finally *define* temperature in the quantum mechanical setting. To do so, we will have to be slightly more careful in deriving an equation governing the dynamics of a system interacting continuously with some bath. A notion of temperature will naturally emerge as a condition on the sorts of bath states that give rise to “well-behaved” system evolutions. This derivation follows the rough trajectory of Breuer and Petruccione [21] with frequent deviations for clarity.

A Series of Approximations

We will again assume some sort of interaction Hamiltonian between a bipartition of a quantum system into system and bath. To simplify the derivation, we work in the interaction picture with respect to the system Hamiltonian, simply so that there are not too many extraneous Hamiltonian terms wandering around the derivation.

The time evolution of the full system + bath density matrix in the interaction picture is given by formally integrating the von Neumann equation:

$$\rho_I(t) = \rho_I(t_0) - i \int_{t_0}^t du [H_I(u), \rho_I(u)] \quad (4.18)$$

Then, we’re free to insert this solution back into the von Neumann equation:

$$\begin{aligned} \frac{\partial \rho_I(t)}{\partial t} &= -i[H_I(t), \rho_I(t)] \\ &= -i[H_I(t), \rho_I(0) - i \int_0^t du [H_I(u), \rho_I(u)]] \end{aligned} \quad (4.19)$$

The first approximation will be the assumption that interactions with the bath are suitably weak so that the joint state can always be approximated as $\rho_I(t) = \rho_s(t) \otimes \rho_b$ for a time independent bath state, ρ_b . This is called the Born approximation. Then, the trace over the bath degrees of freedom yields,

$$\frac{\partial \rho_{IS}(t)}{\partial t} = - \int_0^t du \text{Tr}_b \{ [H_I(t), [H_I(u), \rho_I(u)]] \} \quad (4.20)$$

Because of the dependence of the nested commutator on s , this equation is generically quite difficult to solve without further assumptions. The *time-local* form of this equation replaces $\rho_{IS}(s)$ with $\rho_{IS}(t)$, simplifying the integral. The resulting equation after these two replacements is called the *Redfield equation*:

$$\frac{\partial \rho_{IS}(t)}{\partial t} = - \int_0^t du \text{Tr}_b \{ [H_I(t), [H_I(u), \rho_{IS}(t) \otimes \rho_b]] \} \quad (4.21)$$

This integral is still difficult to solve because of the possibly arbitrary time dependence of the Hamiltonian. The *Markov* approximation, used somewhat haphazardly in the previous section, mollifies this by first making the exact integral change of variables from $u \rightarrow (t - u)$, and then the inexact extension of the bounds of the integral to ∞ . This approximation is valid so long as the integral decays sufficiently quickly as u moves away from t . This is equivalent to the statement that the bath has a very short memory—if the integral decays exponentially quickly in $t - u$, then extending the bounds of the integral to ∞ does not appreciably change the integral. These replacements all together are often referred to as the *Born-Markov* approximation, and yield,

$$\frac{\partial \rho_{IS}(t)}{\partial t} = - \int_0^\infty ds \text{Tr}_b \{ [H_I(t), [H_I(t - u), \rho_{IS}(t) \otimes \rho_b]] \} \quad (4.22)$$

The final approximation necessary to (re)arrive at the Lindblad equation is the *Secular* approximation. For this approximation it will be convenient to reintroduce a form for the interaction Hamiltonian,

$$H_I = \sum_k s_k \otimes b_k \quad (4.23)$$

We can also build the spectrally resolved operators,

$$s_k(\omega) \equiv \sum_{\epsilon' - \epsilon = \omega} \Pi_\epsilon s_k \Pi_{\epsilon'} \quad (4.24)$$

where ϵ and ϵ' are projectors onto system eigenstates, and the double sum runs over all pairs of system eigenstates satisfying the equality $\epsilon - \epsilon' = \omega$. The a_k can be reconstructed from these frequency resolved operators by taking the sum $\sum_\omega a_k(\omega) = a_k$. These ω -dependent operators are convenient for reasoning about system transitions, and will provide the appropriate granularity with which to take the Secular approximation shortly.

Now, consider the interaction Hamiltonian in the interaction picture (where we co-rotate with the system Hamiltonian, H_s):

$$H_I = \sum_{k,\omega} e^{iH_s t} s_k(\omega) \otimes b_k e^{-iH_s t} \quad (4.25)$$

This simplifies if we make use of the commutation relations for the $s_k(\omega)$, namely:

$$\begin{aligned}
 \langle \epsilon | [H_s, s_k(\omega)] | \epsilon' \rangle &= \langle \epsilon | (H_s s_k(\omega) - s_k(\omega) H_s) | \epsilon' \rangle \\
 &= \langle \epsilon | \epsilon s_k(\omega) - s_k \epsilon' | \epsilon' \rangle \delta(\epsilon' - \epsilon - \omega) \\
 &= \langle \epsilon | (-\omega) s_k | \epsilon' \rangle \\
 &\rightarrow [H_s, s_k(\omega)] = (-\omega) s_k(\omega) \tag{4.26} \\
 &\rightarrow [H_s, s_k^\dagger(\omega)] = (\omega) s_k^\dagger(\omega) \tag{4.27}
 \end{aligned}$$

Thus, inserting the commutator in 4.25, we retrieve:

$$\begin{aligned}
 H_I &= \sum_{k,\omega} e^{iH_s t} s_k(\omega) e^{-iH_s t} \otimes b_k \\
 &= \sum_{k,\omega} s_k(\omega) e^{i(H_s - \omega)t} e^{-iH_s t} \otimes b_k \\
 &= \sum_{k,\omega} e^{-i\omega t} s_k(\omega) \otimes b_k \tag{4.28}
 \end{aligned}$$

Now, we can insert this form of the interaction Hamiltonian into the Born-Markov master equation (Eq. 4.22).

$$\begin{aligned}
 \frac{\partial \rho_{IS}(t)}{\partial t} &= - \int_0^\infty ds \text{Tr}_b \{ [(\sum_{k,\omega} e^{-i\omega t} s_k(\omega) \otimes b_k), [(\sum_{k,\omega} e^{-i\omega(t-u)} s_k(\omega) \otimes b_k), \rho_{IS}(t) \otimes \rho_b]] \} \\
 &= \sum_{kk'} \sum_{\omega\omega'} e^{i(\omega' - \omega)t} \Gamma_{kk'}(\omega) \left(s_{k'}(\omega) \rho_s(t) s_k^\dagger(\omega') - s_k^\dagger(\omega') s_{k'}(\omega) \rho_s(t) + h.c. \right) \tag{4.29}
 \end{aligned}$$

where the constants $\Gamma_{kk'}$ are the Fourier transforms,

$$\Gamma_{kk'}(\omega) = \int_0^\infty du e^{i\omega u} \text{Tr}_b \{ b_k^\dagger(t) b_{k'}(t-u) \rho_b \} \tag{4.30}$$

Finally, we are prepared to make the Secular approximation. Eq. 4.29 is the sum of many terms, rotating with many different frequencies. The Secular approximation drops all terms from this sum for which $\omega' \neq \omega$ (this is sometimes called the rotating wave approximation (RWA)). This approximation is often made independent of whether or not it is suitable, simply because it turns a complicated double sum into a tractable, simple sum. If there happens to be an intrinsic separation of timescales, where the timescale set by the system-bath coupling is much slower than these transition timescales, then these terms can be safely neglected (i.e., this can be interpreted as a sort of weak-coupling limit). Intuitively, we are not concerned with involving oscillatory terms like $\exp i\omega't$ for ω' incommensurate with ω , because these terms average out. Care must be taken, however, for frequencies close to ω .

Finally, to re-arrive at the Lindblad equation, we need only decompose the bath correlation functions as:

$$\Gamma_{kk'}(\omega) = \frac{1}{2}\gamma_{kk'}(\omega) + i\Lambda_{kk'}(\omega) \quad (4.31)$$

that is, its real and imaginary parts. We will primarily be concerned with the real parts of the bath correlation functions,

$$\gamma_{kk'}(\omega) = \int_{-\infty}^{\infty} du e^{i\omega u} \langle s_k(u)^\dagger s_{k'}(0) \rangle \quad (4.32)$$

It can be shown [21] that the imaginary parts of Γ contribute to the Lamb-Shift contribution that we encountered previously in the discrete case 4.15. In terms of these bath correlation functions, the Lamb shift terms appear as,

$$H_{\text{Lamb}} = \sum_{\omega} \sum_{kk'} \Lambda_{kk'}(\omega) s_k^\dagger(\omega) s_{k'}(\omega) \quad (4.33)$$

Collecting all terms and refactoring, we once again rearrive at the Lindblad equation, but now with a formula relating the bath correlation functions to the rates with which different Lindblad operators act:

$$\frac{\partial \rho_{IS}(t)}{\partial t} = -i[H_{\text{Lamb}}, \rho_{IS}(t)] + \sum_{\omega} \sum_{kk'} \gamma_{kk'}(\omega) (s_{k'}(\omega) \rho_{IS}(t) s_k^\dagger(\omega) - \frac{1}{2} \{s_k^\dagger(\omega) s_{k'}(\omega), \rho_{IS}(t)\}) \quad (4.34)$$

Temperature and the KMS Condition

It is a result from classical thermodynamics that one expects the equilibrium distribution of ergodic states to follow,

$$\rho_{\text{thermal}} \propto e^{-\beta H} \quad (4.35)$$

for inverse temperature β , and for a Hamiltonian, H [75]. For our quantum bath, we will demand the same—that its density matrix is a Gibbs state, but we will additionally demand (as a consequence of the Born-Markov assumptions) that the presence of the thermal bath necessarily drives the *system* to a thermal Gibbs state. To prove that this familiar, classical thermal state is indeed *the* stationary state of the system in the presence of these dynamics, we make use of the Kubo-Martin-Schwinger boundary conditions (the KMS Condition) [79]:

$$\langle b_k^\dagger(t) b_{k'}(0) \rangle = \langle b_{k'} b_k^\dagger(t + i\beta) \rangle \quad (4.36)$$

which is satisfied by thermal states like Eq. 4.35. Verifying:

Proof.

$$\begin{aligned}
& \langle b_k^\dagger(t) b_{k'}(0) \rangle_{\text{thermal}} \\
&= \text{Tr}_b \left(\frac{e^{-\beta H_b} b_k^\dagger(t) b_{k'}(0)}{\text{Tr}_b(e^{-\beta H_b})} \right) \\
&= \text{Tr}_b \left(\frac{e^{-\beta H_b} e^{iH_b t} b_k^\dagger e^{-iH_b t} b_{k'}(0)}{\text{Tr}_b(e^{-\beta H_b})} \right) \\
&= \text{Tr}_b \left(b_{k'}(0) \frac{e^{iH_b(t+i\beta)} b_k^\dagger e^{-i(H_b+i\beta H_b-i\beta H_b)t}}{\text{Tr}_b(e^{-\beta H_b})} \right) \\
&= \text{Tr}_b \left(b_{k'}(0) \frac{e^{iH_b(t+i\beta)} b_k^\dagger e^{-iH_b(t+i\beta)} e^{-\beta H_b}}{\text{Tr}_b(e^{-\beta H_b})} \right) \\
&= \text{Tr}_b \left(b_{k'}(0) \frac{b_k^\dagger(t + i\beta) e^{-\beta H_b}}{\text{Tr}_b(e^{-\beta H_b})} \right) \\
&= \langle b_{k'}(0) b_k^\dagger(t + i\beta) \rangle_{\text{thermal}} \quad (4.37)
\end{aligned}$$

□

The KMS Condition is also extremely useful in establishing a relationship between the forward and reverse transition rates, $\gamma(\omega)$ and $\gamma(-\omega)$, which we use extensively in later chapters. This relationship is often called *detailed balance*:

$$\gamma_{kk'} = e^{-\beta\omega} \gamma_{k'k}(\omega) \quad (4.38)$$

This follows immediately from the KMS Condition, and the self-adjointness of the rate matrix, $\gamma_{kk'}$:

Proof.

$$\begin{aligned}
 \gamma_{kk'}(-\omega) &= \int_{-\infty}^{\infty} du \, e^{-i\omega u} \langle b_k^\dagger(u) b_{k'}(0) \rangle \\
 &= \int_{-\infty}^{\infty} du \, e^{i\omega u} \langle b_k(0) b_{k'}^\dagger(u) \rangle \\
 &= \int_{-\infty}^{\infty} du \, e^{-\beta\omega + i\omega(u-i\omega)} \langle b_k(0) b_{k'}^\dagger(u) \rangle \\
 &= e^{-\beta\omega} \int_{-\infty}^{\infty} du \, e^{i\omega(u)} \langle b_k(0) b_{k'}^\dagger(u + i\beta) \rangle \\
 &= e^{-\beta\omega} \int_{-\infty}^{\infty} du \, e^{i\omega(u)} \langle b_{k'}^\dagger(u) b_k(0) \rangle \\
 &= e^{-\beta\omega} \gamma_{k'k}(\omega)
 \end{aligned} \tag{4.39}$$

□

Finally, we need one more commutator, and we'll be equipped to prove the stationarity of the thermal system state,

$$[\rho_{\text{Thermal}}, s_k(\omega)] = (e^{\beta\omega} - 1) s_k(\omega) \rho_{\text{Thermal}} \tag{4.40}$$

But this follows almost immediately from Eqs. 4.26 and 4.27.

Proof.

$$\begin{aligned}
 &\rho_{\text{Thermal}} s_k(\omega) - s_k(\omega) \rho_{\text{Thermal}} \\
 &= e^{-\beta H_s} s_k(\omega) - s_k(\omega) e^{-\beta H_s} \\
 &= s_k(\omega) e^{-\beta(H_s - \omega)} - s_k(\omega) e^{-\beta H_s} \\
 &= s_k(\omega) e^{-\beta H_s} (e^{\beta\omega} - 1)
 \end{aligned} \tag{4.41}$$

□

or, equivalently:

$$\rho_{\text{Thermal}} s_k(\omega) = e^{\beta\omega} s_k(\omega) \rho_{\text{Thermal}} \tag{4.42}$$

$$\rho_{\text{Thermal}} s_k^\dagger(\omega) = e^{-\beta\omega} s_k^\dagger(\omega) \rho_{\text{Thermal}} \tag{4.43}$$

Finally, inserting ρ_{Thermal} into Eq. 4.34, (and noting that ρ_{Thermal} commutes with the Lamb Shift term,

$$\begin{aligned}
\frac{\partial \rho_{\text{Thermal}}}{\partial t} &= 0 = -i[H_{\text{Lamb}}, \rho_{IS}(t)] + \sum_{\omega} \sum_{kk'} \gamma_{kk'}(\omega) (s_{k'}(\omega) \rho_{\text{Thermal}} s_k^\dagger(\omega) - \frac{1}{2} \{s_k^\dagger(\omega) s_{k'}(\omega), \rho_{\text{Thermal}}\}) \\
&= 0 + \sum_{\omega} \sum_{kk'} \gamma_{kk'}(\omega) (s_{k'}(\omega) \rho_{\text{Thermal}} s_k^\dagger(\omega) - \frac{1}{2} \{s_k^\dagger(\omega) s_{k'}(\omega), \rho_{\text{Thermal}}\}) \\
&= \sum_{\omega} \sum_{kk'} \gamma_{kk'}(\omega) (s_{k'}(\omega) \rho_{\text{Thermal}} s_k^\dagger(\omega) - \frac{1}{2} s_k^\dagger(\omega) s_{k'}(\omega) \rho_{\text{Thermal}} - \frac{1}{2} \rho_{\text{Thermal}} s_k^\dagger(\omega) s_{k'}(\omega)) \\
&= \sum_{\omega} \sum_{kk'} \gamma_{kk'}(\omega) (s_{k'}(\omega) \rho_{\text{Thermal}} s_k^\dagger(\omega) - s_k^\dagger(\omega) s_{k'}(\omega) \rho_{\text{Thermal}}) \\
&= 0
\end{aligned} \tag{4.44}$$

where we commuted ρ_{Thermal} through to the right between the third and fourth lines, and where the last line follows in expectation because of the cyclic property of the trace—i.e., $\text{Tr}(s_{k'} \rho_{\text{Thermal}} s_k^\dagger) = \text{Tr}(s_k^\dagger s_{k'} \rho_{\text{Thermal}})$. Thus, a thermal system state is a fixed point of the evolution of the Lindblad equation.

To recap, in the previous section, we saw how imposing that our bath be in a thermal state necessarily caused a fixed point of the system to also be a thermal state. As a consequence of holding the bath at a fixed temperature, this temperature necessarily sets the temperature scale of the system thermal state's long-time fixed point of the Lindblad evolution. It's quite a bit more difficult to show that this fixed point might be unique, because there are conditions in which it is not. Keeping in mind that this thesis is nominally about *error correction*, that this fairly minimal coupling scheme to an environment necessarily caused a system to evolve into a thermal state after some amount of time should be worrisome—the thermal state is the worst possible state to use to try to encode quantum information!

Nonetheless, this result should not be *surprising*. We already saw back in Sec. 2.2 that even a very simple unitary interaction between two qubits gave rise to system dynamics that ultimately drove the system into a totally mixed state. Understanding these interactions, and, in particular, the timescales of these interactions, controlled by $\gamma_{kk'}$ will be a central component of future chapters.

Chapter 5

An interlude on numerical methods

34:7 What man is there that knoweth not how to go about doing arithmetic on polynomials.

King James Programming Tumblr

Before we can dive in to how exactly thermal environments destroy order in stabilizer error correcting codes, we will take a short excursion into the theory of Markov processes. For the remaining chapters, I regularly use these tools to probe stabilizer codes at finite temperature. This chapter serves as a self-contained treatment of the theory behind those tools. Most introductory texts on statistical methods treat Markov processes in detail, but I found [110, 133] to be quite useful.

5.1 Markov Processes

We have, in fact, already encountered Markov processes in both their discrete and continuous variants. The steady state solution to the bit-flip error dynamics of Sec. 2.2 was a discrete Markov chain in disguise, and the Lindblad equation derived in the previous chapter describes a continuous time Markov process. In this section, we briefly review the theory of Markov processes, and how they can be solved.

Definitions

Most generally, Markov chains are collections of random variables, $X(t)$, where the “time” index may be either discrete or continuous, satisfying the *Markov* property:

$$\begin{aligned} P(X(n+1) = x_{n+1} \mid X(n) = x_n, X(n-1) = x_{n-1}, X(n-2) = x_{n-2}, \dots, X(0) = x_0) \\ = P(X(n+1) = x_{n+1} \mid X(n) = x_n) \end{aligned} \quad (5.1)$$

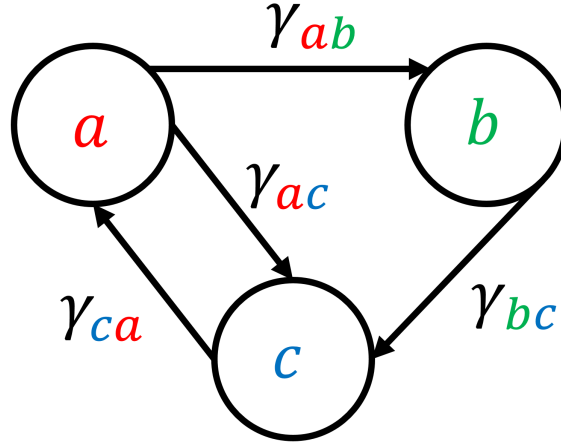


Figure 5.1: A directed graph representation of an example Markov chain with possible state variables a, b , and c . Transition probabilities / transition rates are encoded by the directed quantities γ_{ij} .

where we are using $X(t)$ to refer to the random variable at a particular time, and lowercase x_t to refer to a particular realization of that random at that time. In words, random variables have the Markov property when the probability distribution for the next instantiation of the variable depends only on the current value of the random variable. Often, it is convenient to represent these sorts of systems via a directed graph (see Fig. 5.1, where vertices of the graph encode states (i.e., x_i), and the edges encode transition probabilities.

For continuous time processes, the above Markov property holds, but it must be generalized slightly to account for arbitrary time intervals, and takes the form,

$$P(X(t + \delta t) = i \mid P(t) = j) = \delta_{ij} + \gamma_{ij}\delta t \quad (5.2)$$

where γ_{ij} encodes the transition rates between system states. Note that in either situation, the equation for the probabilities of populations of different states as a function of time can be formed by constructing the appropriate matrix equation. Defining the population vector $P(t)$, for discrete processes:

$$P(k) = (\Gamma_{ij})^k P(0) \quad (5.3)$$

for an initial population vector $P(0)$, and for transition probabilities Γ_{ij} . For continuous processes, is the solution to the matrix differential equation:

$$\begin{aligned}\frac{dP(t)}{dt} &= \gamma_{ij}P(t) \\ \rightarrow P(t) &= e^{\gamma_{ij}t}P(0)\end{aligned}\tag{5.4}$$

In either case, taking the matrix power (discrete) or matrix exponential (continuous) of γ is considerably easier after diagonalizing γ . In general, then, the stationary / long-time behaviors of these equations are entirely controlled by the eigenvalues and eigenvectors of the transition matrix γ . As a word of caution, these transition matrices take slightly different forms for discrete versus continuous processes, which is already implicit in the above definitions, but bears elaboration for the sake of clarity. For discrete processes, the rows of Γ_{ij} are strictly positive and sum to 1 to conserve probability. That is,

$$\sum_j \Gamma_{ij}^{\text{discrete}} = 1 \tag{5.5}$$

These matrices are often called *stochastic* matrices, and form a Lie group. For continuous processes, the γ_{ij} are not probabilities, but are instead transition rates, thus these γ_{ij} are somewhat unimaginatively called *transition rate* matrices, and satisfy a different set of condition:

$$\sum_j \gamma_{ij}^{\text{cont.}} = 0 \tag{5.6}$$

$$\sum_j \gamma_{i \neq j}^{\text{cont.}} = -\gamma_{ii}^{\text{cont.}} \tag{5.7}$$

These transition rate matrices form a Lie algebra—in particular, they are the infinitesimal generators of the group of stochastic matrices[85].

Error maps as Markov chains

Recasting the error process considered in Sec. 2.2 as a Markov chain is nearly immediate: the single-qubit density matrix, ρ , can always be reshuffled into a column vector with the same four entries, and the stochastic matrix for bit flip errors is simply:

$$\Gamma^{\text{bit flip}} = \begin{pmatrix} 1-p & 0 & 0 & p \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ p & 0 & 0 & 1-p \end{pmatrix} \tag{5.8}$$

which has eigenvalues and eigenvectors,

$$\lambda_1 = 1, \vec{v}_1 = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 1 \end{pmatrix} \quad (5.9)$$

$$\lambda_2 = 1 - 2p, \vec{v}_2 = \begin{pmatrix} 1 \\ 0 \\ 0 \\ -1 \end{pmatrix} \quad (5.10)$$

$$(5.11)$$

Inspecting the form of these eigenvalues and vectors, we see immediately that any initial population vector's projection onto the second eigenvector will decay exponentially as the Markov process proceeds in time. Additionally, the stationary distribution, with eigenvalue 1 falls out immediately: the equal population vector. That is, after infinite time, we expect the presence of bit-flip error dynamics to drive us into a state of maximal uncertainty, with equally likelihood of being in the 0 or 1 state.

The Lindblad equation as a Markov chain

If the system's Hamiltonian spectrum is degenerate (which will not be the case for the toric code), or if the Lindblad operators only ever connect single eigenstates to other, single eigenstates rather than superpositions of eigenstates (which *will* be the case for our particular bath model), then the equations of motion for the diagonal entries of the density matrix decouple from the off-diagonal elements under evolution by the Lindblad equation. In this case, the equations of motion for the diagonal entries—i.e., the populations—will take the form of a continuous time Markov process. Sandwiching Eq. 4.34 with bras and kets for a single component of the diagonal of the density matrix, we retrieve[21]:

$$\begin{aligned} \frac{d \langle \epsilon_i | \rho | \epsilon_i \rangle}{dt} \equiv \frac{dP_i(t)}{dt} = \sum_j \left(\sum_{k,k'} \gamma_{kk'}(\epsilon_j - \epsilon_i) \underbrace{\langle \epsilon_j | s_k | \epsilon_i \rangle \langle \epsilon_i | s_{k'} | \epsilon_j \rangle}_{\overline{\gamma_{kk'}}} P_j(t) \right. \\ \left. - \underbrace{\gamma_{kk'}(\epsilon_i - \epsilon_j) \langle \epsilon_i | s_k | \epsilon_j \rangle \langle \epsilon_j | s_{k'} | \epsilon_i \rangle}_{\overline{\gamma_{kk'}}} P_i(t) \right) \end{aligned} \quad (5.12)$$

The underlined quantities, involving the bath correlation functions and system operator matrix elements are the components of the transition rate matrix for the corresponding Markov process, which we'll denote $\overline{\gamma}$ to disambiguate it from the bath correlation functions, $\gamma_{kk'}$. This equation is still in the form of Eq. 5.4 if we identify

$$\overline{\gamma_{kk'}} \equiv \gamma_{kk'}(\epsilon_j - \epsilon_i) \langle \epsilon_j | s_k | \epsilon_i \rangle \langle \epsilon_i | s_{k'} | \epsilon_j \rangle \quad (5.13)$$

Note that we're free to rewrite Eq. 5.4 in the following way:

$$\begin{aligned}
\frac{dP(t)}{dt} &= \sum_j \overline{\gamma_{ij}} P_j(t) \\
&= \sum_j \overline{\gamma_{ij}} P_j(t) \\
&= \left(\sum_{j \neq i} \overline{\gamma_{ij}} P_j(t) \right) + \overline{\gamma_{ii}} P_i(t) \\
&= \left(\sum_{j \neq i} \overline{\gamma_{ij}} P_j(t) \right) - \left(\sum_{j \neq i} \overline{\gamma_{ji}} P_i(t) \right) \\
&= \left(\sum_j \overline{\gamma_{ij}} P_j(t) - \overline{\gamma_{ji}} P_i(t) \right)
\end{aligned} \tag{5.14}$$

where we have used Eq. 5.7 between lines 3 and 4, and the last step extends the sum to include $j = i$ because this is equivalent to adding 0. Thus, the diagonal populations of the density matrix under Lindblad evolution are a continuous Markov process.

5.2 Continuous time Monte Carlo

At first glance, the results of the previous chapter are encouraging—we are but a simple eigenvalue problem away from determining the thermal behavior of our stabilizer code of choice. Unfortunately, the dimensionality of this problem ends up being prohibitively large. Even restricting to the lowest order processes involving only say, 2 quasiparticles in a system with side-dimension L , there are $\binom{L^2}{2} \propto L^4$ such states in the system. Worse, in practice, the computational cost of eigendecomposition scales as $O(n^3)$ (or $O(n^2)$ with sparsity), or $O(L^{12})$. Even making use of all available sparsity cannot reduce this below $O(L^8)$. Because practical implementations of surface codes will likely involve thousands to millions of qubits, eigendecomposition simply cannot scale to systems of practical relevance.

Instead, we will solve the Lindblad equation via continuous time Monte Carlo—a computationally efficient method for sampling random trajectories of the evolution of the system. This method was originally called kinetic Monte Carlo [138], but is also called the N-fold Way by a group that contemporaneously and independently discovered the algorithm [11].

The algorithm

The continuous time monte carlo algorithm is detailed in Alg. 2. In words, the system is initialized, and a *total decay rate* λ is calculated, which represents the sum of all the transition rates out of the current state. λ measures approximately how fast the system

will transition out of any given state. Then, using a number r chosen uniformly at random from the interval $(0, 1)$, we generate a *waiting time* $\delta\tau$ via the expression $\frac{-1}{\lambda}\log(r)$. This quantity represents one random sampling of the amount of time one would expect to wait given an exponentially distributed random variable with parameter λ (see below). Then, we choose a new state to transition into randomly so that the probability of transitioning into any given state is proportional to the fraction of that state's transition rate relative to the total decay rate. As we will see, this guarantees detailed balance, and will ensure that we faithfully sample the real dynamics of the markov process. Equivalently, this ensures that the probability of transitioning into any given state is proportional to the transition rate into that state.

Algorithm 2 Continous time monte carlo of markov processes

- 1: $|i\rangle$, $t = 0 \leftarrow$ Initial conditions
 - 2: $\lambda \leftarrow \|\sum_{j \neq i} \overline{\gamma_{ij}}\|$
 - 3: $r \leftarrow \text{Uniform}(0, 1)$
 - 4: $\delta\tau \leftarrow \frac{-1}{\lambda}\log(r)$
 - 5: Choose new state k so that $\sum_{j \neq i}^{k-1} \frac{\overline{\gamma_{ij}}}{\lambda} < r < \sum_{j \neq i}^k \frac{\overline{\gamma_{ij}}}{\lambda}$
 - 6: Set $|i\rangle \rightarrow |k\rangle$, $t \rightarrow t + \delta\tau$
 - 7: Go to 2. and repeat until $t > T_{max}$
-

This algorithm arises from the following sequence of observations. For an infinitesimal interval δt , the probability of a system remaining stationary—i.e., not transitioning into a new state—is,

$$P_{i \rightarrow i}(\delta t) = 1 - \lambda \delta t, \quad \lambda \equiv \sum_{i \neq j} \overline{\gamma_{ij}} \quad (5.15)$$

For a finite interval, we need only take the limit,

$$P_{i \rightarrow i}(\Delta t) = \lim_{y \rightarrow \infty} \left(1 - \lambda \frac{\Delta t}{y}\right)^y \quad (5.16)$$

$$= e^{-\lambda \Delta t} \quad (5.17)$$

Thus, on average, the system remains stationary for a time λ^{-1} , and then transitions somewhere. Thus, the probability of making a transition *out* of a given state within the time interval δt *after* a waiting time t is the product,

$$w_{i \rightarrow j}(t) \delta t = P_{i \rightarrow i}(t) \cdot \overline{\gamma_{ij}} \delta t \quad (5.18)$$

where $w_{i \rightarrow j}(t)$ is the transition probability density at time t between states i and j . Finally, comparing the transition probability densities between transitions to two different states, we see:

$$\frac{w_{i \rightarrow j}}{w_{i \rightarrow k}} = \frac{\overline{\gamma_{ij}}}{\overline{\gamma_{ik}}} \quad (5.19)$$

That is: the ratio of transition probability densities between two outgoing states is proportional to the ratio of outgoing rates, verifying our state selection choice in the algorithm in step 5.

Thus, a single “run” of the algorithm produces one independent trajectory of the dynamics of the system. In practice, observables can be collected by running many independent copies of the algorithm from the same initial conditions, and then averaging observables at discrete time intervals. This was the workhorse algorithm for collecting data used in the remaining chapters.

Discrete vs. continuous time monte carlo

Intuitively, the algorithm proceeds by initializing the system in some state, randomly selecting a waiting time in a way consistent with the transition rates of the system, and then transitioning forwards, again respecting the relative likelihoods of different outgoing states. This procedure is not overly complicated, but one might wonder why we used this procedure rather than a *discrete* time algorithm, so as not to have to worry about sampling waiting times correctly.

For the systems we studied, there was often a large separation in timescales between different processes. In other words, it was often the case that $\gamma_{ij} \gg \gamma_{kl}$ for many ij, kl pairs, often by a factor of one hundred or more. Thus, any discrete time picture would be doomed to evolve through long stretches of time where nothing of consequence was occurring, because the simulation necessarily would have to be performed at the shortest time-interval of relevance. The continuous time picture allows us to sidestep this problem entirely, and only ever take timesteps when the we are promised that *something* has occurred.

Chapter 6

The toric code at finite temperature

10:20 Are not my days few? cease then, and let me not fall into a loop here.

??King James Programming Tumblr

The toric code was known to be a poor quantum memory since at least Nussinov and Ortiz’s study of the thermal properties of the code using Davies generators[102]. In the following Chapter I provide the first *dynamical* treatment of the error pathways of the toric code at finite temperature. This chapter is the content of the manuscript Freeman et al. [55].

6.1 Introduction

The fact that the potential power of a large scale quantum computer has not yet been realized experimentally is due largely to the fragility of quantum information. A “conventional” quantum computer stores quantum information in spatially localized qubits—consequently local noise can generate errors that destroy the locally stored quantum information. The theoretical possibility of a fault tolerant quantum computer is well understood in the literature; in general this requires building redundancy into the experimental systems such that errors can be detected and corrected. Although fault tolerance via such active error correction is theoretically feasible, the overhead required to perform active error correction has thus far kept a large scale quantum computer out of reach.

An alternative approach to fault-tolerant quantum computing is based on building *physically* robust quantum hardware with passive error correction. The notion of a topological quantum computer builds on the possibility of storing quantum information nonlocally in a robust quantum phase of matter with topological order [76, 51, 129, 95]; consequently, these phases of matter appear to hint at the potential design of a *self-correcting* quantum computer. Indeed, while in equilibrium with a zerotemperature reservoir, a topological qubit is “topologically protected,” in that errors due to local perturbations are suppressed expo-

nentially in the system size. Despite this promise, subsequent work has demonstrated that topological phases in two dimensions (2D) are thermally fragile because topological order is destroyed at any nonzero temperature [23, 102, 63, 89, 71]. While higher-dimensional topological phases are robust at finite temperature [26, 24, 4, 17, 14, 60, 5, 64], it is only in 2D that such phases can act as a universal quantum computer based on topologically protected operations [76, 52, 92, 93, 95]. This shortcoming seems to preclude the possibility of a universal topologically protected quantum computer.

Accordingly, the 2D toric code fails to be a true fault-tolerant quantum memory in 2D [76, 35, 13, 2, 3, 18, 136, 30, 82, 25]. However, while topological order is destroyed at any finite temperature in the thermodynamic limit, on a finite-size system the topological order of the toric code nevertheless persists up to a finite-size crossover temperature [23]. This suggests the possibility of operating a topological qubit in a low-temperature regime where topological order persists due to finite-size effects. While finite-size effects reduce the zero-temperature robustness to unitary perturbations [76], the existence of a low-temperature regime below the crossover temperature suggests that such finite-size effects may *increase* the thermal robustness. Consequently, characterizing how the memory lifetime of a topological qubit depends on finite-size effects, especially in the low-temperature regime, is of practical importance.

In this paper, we use real-time Monte Carlo simulations to study the relaxation dynamics of finite-size topological qubits defined by the toric code, in contact with a thermal reservoir. Previous work using related methods focused on the high-temperature scaling of decoherence times [29, 109, 111, 70]; here we focus on the dynamics at low temperatures. We find a low-temperature regime that is well described by thermal relaxation dominated by quasiparticle pairs undergoing topologically nontrivial random walks. At higher temperatures, the decoherence is dominated by local creation and annihilation of quasiparticle pairs. The transition between these two regimes allows for a *dynamical* definition of the crossover temperature T^* . We find that $T^* \sim 1/\ln N$, which agrees with the scaling of a transition temperature defined from the topological entanglement entropy at equilibrium [23]. Additionally we find that both the finite-size and finite-temperature scaling are stronger below than above T^* .

The structure of this paper is the following: in Sec. 6.2 we present the relevant background of the toric code; Sec. 7.5 introduces a microscopic master equation of the toric code interacting with a bath as well as an effective model of the low-temperature dynamics; Sec. 6.4 presents a numerical study of topologically nontrivial random walks on a torus that we use to construct the low-temperature effective model; and, finally, Sec. 6.5 presents a numerical study of the microscopic master equation for the toric code interacting with a bath and an analysis of these results in comparison with the low-temperature effective model.

6.2 The toric code

The toric code Hamiltonian

The toric code provides a simple exactly soluble model with a topologically ordered ground state that may provide topologically protected qubits at $T = 0$ [76, 35]. The toric code is defined on a square lattice, where Ising spins sit on the links of the lattice. We define the linear dimension of the lattice as L and the number of spins $N = 2L^2$. The Hamiltonian involves four-spin interactions around the plaquettes and vertices of the lattice:

$$H_{\text{TC}} = -J_e \sum_v A_v - J_m \sum_p B_p, \quad (6.1)$$

$$A_v \equiv \prod_{j \in v} \sigma_j^z, \quad B_p \equiv \prod_{j \in p} \sigma_j^x, \quad (6.2)$$

where the sums over v and p are over the vertices and plaquettes of the lattice, respectively (see Fig. 6.1). The ground states are the $+1$ eigenstate of all A_v and B_p operators, since all such operators commute. On a torus, there are four degenerate ground states that are distinguished by the expectation values of non-local winding operators $W_{1,2}^x, W_{1,2}^z$:

$$W_{1,2}^x \equiv \prod_{j \in \Gamma_{1,2}} \sigma_j^x, \quad W_{1,2}^z \equiv \prod_{j \in \tilde{\Gamma}_{1,2}} \sigma_j^z, \quad (6.3)$$

where $\Gamma_{1,2}$ and $\tilde{\Gamma}_{1,2}$ are topologically non-trivial loops along the links and plaquettes of the lattice, respectively, that wind around each of the two axes of the torus. There is a finite gap $\Delta_{e,m} = 4J_{e,m}$ to excited states that are -1 eigenstates of some A_v and/or B_p . These correspond to e -type and m -type *anyonic* quasiparticle excitations, respectively [76].

The toric code as a quantum memory

Consider the $W_{1,2}^z$ basis for the degenerate ground state subspace; we may label the four ground states by the eigenvalues of $W_{1,2}^z$:

$$\{|\Psi_0\rangle\} = \{|\Psi_0^{++}\rangle, |\Psi_0^{-+}\rangle, |\Psi_0^{+-}\rangle, |\Psi_0^{--}\rangle\}, \quad (6.4)$$

where $+/-$ represent the ± 1 eigenstates of W_1^z and W_2^z , respectively. We choose this basis to be the logical basis for a two qubit quantum memory. To simplify the discussion of errors we will consider the limit $J_m \rightarrow \infty$, so that only e -type quasiparticles have finite energy. The lowest excited eigenstates have a single pair of localized e -type quasiparticles which are connected to a ground state by operation of an error string:

$$|e_v, e_{v'}\rangle = S_x[\Gamma_{v,v'}]|\Psi_0\rangle, \quad S_x[\Gamma_{v,v'}] \equiv \prod_{j \in \Gamma_{v,v'}} \sigma_j^x. \quad (6.5)$$

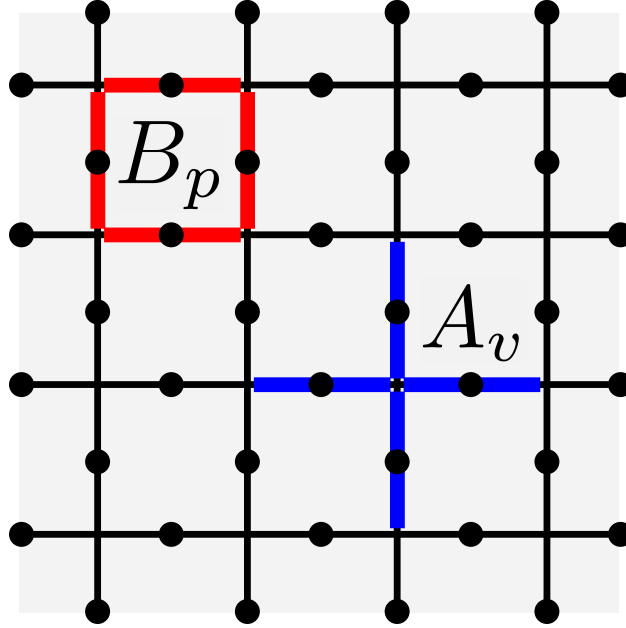


Figure 6.1: The vertex (A_v) and plaquette (B_p) operators of the toric code as defined in (8.16). Edges marked in red are operated on by σ^x while those marked in blue are operated on by σ^z .

Here, v and v' are the vertices where the quasiparticles are located and the string $\Gamma_{v,v'}$ connects v and v' . Error strings that form topologically non-trivial loops generate the $W_{1,2}^x$ operators and drive transitions between ground states; such error strings create *noncorrectable errors*, i.e., errors in the logical subspace that cannot be corrected. *Correctable errors*, or self-correcting errors, are those error strings that close without causing a change in winding number.

Under local perturbations to H_{TC} , such topologically nontrivial error strings only occur at order L in perturbation theory; consequently both the splitting of the ground state degeneracy and transitions between ground states are suppressed exponentially, and thus this ground state subspace is “topologically protected” from such unitary perturbations [77, 16, 15, 120, 122, 37, 90, 74]. The toric code can thus act as a self-correcting quantum memory at *zero* temperature.

The toric code at finite temperatures

Despite the topological protection at zero temperature, in the thermodynamic limit the topological order of the toric code is destroyed at any finite temperature [102, 23]. Consequently, a topological qubit would be thermally fragile. While topologically nontrivial error strings due to *unitary* perturbations are exponentially suppressed in the system size, non-

trivial error strings may also be generated by non-unitary perturbations, e.g., from contact with a thermal reservoir. Non-correctable errors due to non-unitary perturbations are not exponentially suppressed in the system size.

The thermal fragility of the 2D toric code can be understood from a simple picture of the dissipative dynamics generated from local interactions with an external bath. A local system-bath interaction can generate a *trivial error string* by flipping a single spin, thus creating a single pair of neighboring quasiparticles:

$$\sigma_j^x |\Psi_0\rangle = |e_v, e_{v'}\rangle \quad (6.6)$$

where v and v' are the vertices on either end of the link j . The rate of such a process is suppressed exponentially in the inverse temperature, due to the energy gap Δ to such excited states. Such a trivial error is correctable by applying another σ_j^x . However, additional trivial error strings $\sigma_{j'}^x$ with $j \neq j'$ applied to v or v' will generate a longer, *non-trivial error string* at no energy cost. Consequently, local coupling to a bath can drive a random walk of quasiparticle pairs around the lattice with a rate that is only suppressed by a single Boltzmann factor. Such random walks may generate topologically nontrivial error loops and return the system back to the ground state subspace. If an error loop has an odd winding number, this error loop has caused a non-correctable error by driving a transition between ground states. Alternatively, if the error loop has an even winding number, the error is self-correcting. Indeed, Alicki *et al.* have placed a system-size-independent upper bound on the relaxation time of a pure toric code ground state that explicitly demonstrates this thermal fragility of the toric code in the thermodynamic limit [2, 3]. Additionally, the analysis of Nussinov and Ortiz demonstrates the lack of “spontaneous topological symmetry breaking” at finite-temperature, as the autocorrelation time of the winding operators is sub-exponential in lattice dimension in the thermodynamic limit [102, 101].

Crossover temperature

While the toric code is thermally fragile at all non-zero temperatures in the thermodynamic limit, we can also consider how this fragility is affected by the finite size of a lattice. As outlined above, the dissipative error processes which lead to thermal fragility occur when there is a single quasiparticle pair present. Since the number of excitations in equilibrium is suppressed by the Boltzmann factor at low temperatures, we expect that at sufficiently low temperatures the number of quasiparticles in equilibrium will be vanishingly small. We can then define an *equilibrium crossover temperature* T_{eq}^* which distinguishes the thermally fragile regime from a low temperature regime with reduced dissipative error processes by:

$$N e^{-\Delta/T_{eq}^*} \sim 1 \Rightarrow T_{eq}^* \sim \frac{\Delta}{\ln N}. \quad (6.7)$$

Castelnovo and Chamon define an equilibrium crossover temperature T_{eq}^* above which the topological entanglement entropy vanishes [23]. They find that this equilibrium definition of

T^* scales inversely with the log of the system size, and that this becomes a zero temperature phase transition in the thermodynamic limit. Conversely, on a finite sized system the crossover temperature T^* defines a low temperature regime where topological order persists as a finite-size effect. This opens the possibility of using finite-size effects to exploit the zero temperature topological order at finite temperatures. The usefulness of this low temperature regime for quantum information processing depends on the scaling of the relaxation time in this regime. Robustness to unitary perturbations requires a sufficiently large system size to minimize the splitting of the degeneracy and the matrix elements between ground states, while the thermal fragility increases with system size. Thus one may expect that there is an optimal size for computational performance.

Below, we directly address this question of the finite-size scaling of the relaxation time of a toric code ground state in contact with a thermal reservoir. This analysis complements the growing literature concerning *active* error correction on the toric code by use of the stabilizer space and associated stabilizer operations [35, 18, 30, 72]. Usually, stabilizer error analysis is considered in the context of effectively infinite temperature thermal instability[29], in contrast to the finite-temperature dynamics presented here. A complete toolkit for understanding and controlling errors in physical implementations of the toric code would need a predictive low temperature model, as well as a recipe for understanding how the fidelity of stabilizer operations affects the *finite* temperature operation of the toric code. Here we focus on the robustness of the passive error correcting (i.e., self-correcting) dynamics under the action of the toric code Hamiltonian in contact with a thermal reservoir.

6.3 Dynamics of the Toric Code in contact with an external bath

Microscopic Quantum Master Equation

We present a microscopic model of the real-time non-equilibrium dynamics of the toric code in contact with a thermal reservoir. Due to the fact that the spectrum of H_{TC} has a finite gap to excited eigenstates with localized quasiparticle excitations, such dynamics may be described by a Lindblad master equation[2, 3]:

$$\dot{\rho} = \sum_{\omega} 2c_{\omega}\rho c_{\omega}^{\dagger} - c_{\omega}^{\dagger}c_{\omega}\rho - \rho c_{\omega}^{\dagger}c_{\omega}, \quad (6.8)$$

here ρ is the toric code system density matrix and $\{c_{\omega}\}$ is a set of Lindblad operators generated by local system-bath interactions. We will consider the limit where $J_m \gg J_e$, such that at low temperatures the system will remain in the +1 eigensector of all B_p operators and only e -type quasiparticles will be excited by the reservoir. We will only consider local system-bath couplings, for which the bath generates single spin flips in the system. The relevant Lindblad operators are:

$$\{c_{\omega}\} = \{\sqrt{\gamma_0}T_{vv}^e, \sqrt{\gamma_+}E_{vv}^{e\dagger}, \sqrt{\gamma_-}E_{vv}^e\} \quad (6.9)$$

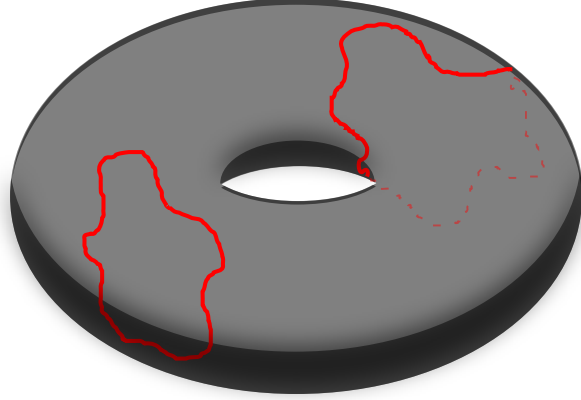


Figure 6.2: A torus with a self-correcting error string (left) and an uncorrectable error string (right).

where $E_{vv'}^{e\dagger}$ ($E_{vv'}^e$) creates (annihilates) a pair of quasiparticles at neighboring vertices and $T_{vv'}^e$ translates a quasiparticles across a link. These operators are defined by:

$$\begin{aligned} E_{vv'}^{e\dagger} &= \frac{1}{4} \sigma_{vv'}^x (1 - A_v) (1 - A_{v'}) , \\ T_{vv'}^e &= \frac{1}{4} \sigma_{vv'}^x (1 - A_v) (1 + A_{v'}) . \end{aligned} \quad (6.10)$$

Since the Lindblad form of the master equation only connects diagonal elements of the density matrix ρ to other diagonal elements, expectation values of diagonal elements will evolve independently of off-diagonal density matrix elements. Correspondingly, the time evolution of diagonal matrix elements reduces to a classical master equation:

$$\begin{aligned} \frac{dP_n}{dt} &= \gamma_0 \sum_{n_0} (P_{n_0} - P_n) + \sum_{n_+} (\gamma_- P_{n_+} - \gamma_+ P_n) \\ &\quad + \sum_{n_-} (\gamma_+ P_{n_-} - \gamma_- P_n) \end{aligned} \quad (6.11)$$

where n labels an eigenstate of H_{TC} , $P_n = \rho_{nn}$ are the diagonal matrix elements (probabilities) and $\{\gamma_0, \gamma_+, \gamma_-\}$ are the rates at which the operators $\{T^e, E^{e\dagger}, E^e\}$ act, respectively. Similarly, in (7.8), for a given n , the indices of n_0 , n_+ , and n_- label the sets of eigenstates connected to $|n\rangle$ by the operators T^e , $E^{e\dagger}$, and E^e , respectively. The ratio γ_+/γ_- is fixed by detailed balance to be

$$\frac{\gamma_+}{\gamma_-} = e^{-\Delta/T} \quad (6.12)$$

but the nature of the bath and the coupling strength determines γ_0 and the magnitude of γ_- (or equivalently γ_+).

We consider here an Ohmic, Markovian bath with a power spectrum given by:

$$J(\omega) = \omega e^{-\frac{\omega}{\omega_c}} \quad (6.13)$$

with ω_c a high frequency cutoff much larger than J_e . Taking ω_c to infinity gives rise to decay rates of the form[29]:

$$\gamma(\omega) = \xi \left| \frac{\omega}{1 - e^{-\beta\omega}} \right| \quad (6.14)$$

where ξ sets the strength of the phenomenological system-bath coupling. This leads to the following rates:

$$\gamma_0 \equiv \frac{\xi}{\beta}, \quad \gamma_+ = \frac{\xi\Delta}{e^{\beta\Delta} - 1}, \quad \gamma_- = \frac{\xi\Delta}{1 - e^{-\beta\Delta}}. \quad (6.15)$$

We are most interested in the dynamics deriving from the initial condition of a pure ground state. We characterize the relaxation from a pure ground state by considering the time evolution of the expectation value of the winding operators:

$$\langle W_{1,2}^Z(t) \rangle \equiv \text{Tr} [\rho(t) W_{1,2}^Z]. \quad (6.16)$$

The population dynamics are governed by the creation of quasiparticle pairs that undergo random walks on the torus and then annihilate. Thermal transitions between ground states occur when the quasiparticle pair undergoes a *topologically non-trivial* random walk before annihilating. The decay of the expectation values in (6.16) from their values in a pure state with eigenvalue ± 1 can be due to both topologically nontrivial random walks generating transitions between ground states, as well as transitions to excited states via propagating, open error strings. Consequently, the statistics of such topological random walks affect the scaling of the lifetime of a ground state.

Comparison to Ising Model Dynamics

Nussinov and Ortiz showed that one can take advantage of the equivalence of the partition function of the toric code and that of 1D classical Ising chains to compute equilibrium properties of the toric code [102, 101]. However, one can not readily take advantage of this mapping for the study of non-equilibrium properties. While the partition function is only a function of the spectrum of the system, non-equilibrium dynamics depend on the nature of the (local) coupling to the external reservoir. Since the mapping of the toric code to an Ising chain maps a 2D model onto a 1D model, local couplings of the toric code to an external reservoir in general can lead to non-local couplings in the corresponding Ising model. Thus, a simple model of the 1D non-equilibrium dynamics of the Ising chains *locally* coupled to an external bath cannot describe the non-equilibrium dynamics of the toric code with a *local* bath coupling. Fundamentally, the dynamics of each system at low temperatures are governed by the random walk of defects (anyons in the toric code, domain walls in the Ising

models); the defects of the Ising model undergo 1D random walks, whereas those of the toric code undergo 2D random walks. Consequently, we cannot directly compute the finite-size relaxation times of the toric code ground states from an analysis of the dynamics of the Ising chain. It is nevertheless useful to discuss the nature of thermal relaxation in a finite-sized Ising chain to help inform our discussion of such dynamics in the toric code.

Consider a periodic 1D chain of L classical Ising spins $s_i = \pm 1$ with energy

$$E = -J \sum_i s_i s_{i+1}, \quad (6.17)$$

where $J > 0$ is a ferromagnetic coupling constant. The ground state of (6.17) is a ferromagnet, but the long range order is destroyed at all nonzero temperatures. Excitations above the degenerate ground states are pairs of domain walls with energy cost $\Delta = 4J$. If a pair of domain walls undergoes a topologically non-trivial 1D random walk, this drives a thermal transition between ground states, which is we will refer to as “ground state relaxation”. At sufficiently low temperatures on a finite-sized system, we can expect that these 1D topologically nontrivial walks will dominate the relaxation time of the magnetization. Such a low temperature regime must occur when there is less than a single pair of defects in equilibrium:

$$L \cdot e^{-\Delta/T} \ll 1. \quad (6.18)$$

At low enough temperatures, there may be a separation of time scales such that $\gamma_+ \ll \gamma_0 \ll \gamma_-$. Intuitively, the domain wall production rate, γ_+ , can be tuned much less than the domain wall annihilation rate, γ_- , simply by lowering the temperature (c.f. (6.12)). For certain choices of bath model, the domain wall hopping rate, γ_0 , can be tuned between the latter two rates. On a finite size lattice the time scale for an extensive random walk of the defects can be estimated by the diffusion equation to be of the order of L^2/γ_0 . We consider the low temperature regime of a finite size lattice where $\gamma_+ \ll \gamma_0/L^2$. In this limit, the rate of topologically nontrivial walks occurring is determined by the rate of production of defect pairs and by the probability that such pairs undergo a topologically nontrivial walk before annihilating. This is because any domain wall pairs which proceed to an extensive random walk will carry out their walk and annihilate much faster than another pair of defects will be created.

We consider only the lowest order processes at first order in γ_+ . Once a defect pair is created, the probability of the pair separating instead of trivially annihilating is of the order of γ_0/γ_- . The lowest order processes will annihilate upon their first return to neighboring links. Processes for which the defect pair do not annihilate after the first return to neighboring links will occur at higher order in γ_0/γ_- . Consequently the overall order of these lowest order processes is $\gamma_+\gamma_0/\gamma_-$. We may then introduce a phenomenological form of the relaxation rate from a ferromagnetic ground state:

$$\Gamma_{\text{Ising}}(\beta, N) \sim \gamma_0 \cdot e^{-\Delta/T} \cdot L \cdot P_{1D}^\Omega(L). \quad (6.19)$$

The linear scaling in L arises from the number of locations for domain wall pairs to be created. The factor $P_{1D}^\Omega(L)$ is the probability that any given domain wall that does not

immediately annihilate eventually undergoes a topologically nontrivial random walk. A symmetry argument shows that $P_{1D}^\Omega(L) \sim L^{-1}$ (see Appendix 6.9). This linear scaling of $P_{1D}^\Omega(L)$ suggests that the relaxation rate of the Ising model is size independent: $\Gamma_{\text{Ising}} \sim \gamma_0 \cdot e^{-\Delta/T}$.

In Ref. [58] Glauber solved the exact dynamics of the Ising chain in contact with a thermal reservoir. Glauber uses a bath where γ_0 is taken to be a constant and

$$\frac{\gamma_0}{\gamma_-} = \frac{1}{2} \frac{1}{1 - e^{-\Delta/T}}. \quad (6.20)$$

The relaxation time of this model is found to be

$$\Gamma_{\text{Glauber}} = \frac{\gamma_0}{1 + e^{\Delta/T}} \quad (6.21)$$

At low temperatures, $\Gamma_{\text{Glauber}} \approx \gamma_0 e^{-\Delta/T}$, in agreement with the expected size independent form of the low temperature single defect pair model(6.19), despite the fact that $\gamma_0/\gamma_- \ll 1$ is not satisfied.

Low temperature phenomenological dynamics

We can now make an analogous argument for the toric code. At sufficiently low temperatures on a finite-sized lattice, the relaxation rate from a ground state should be dominated by the dynamics of a single quasiparticle pair. Transitions between ground states are generated by pairs of excitations that annihilate after undergoing a topologically non-trivial 2D random walk with an odd winding number. The low temperature regime dominated by single defect pairs occurs when

$$L^2 \cdot e^{-\Delta/T} \ll 1. \quad (6.22)$$

We consider a separation of time scales for the Ohmic bath defined by (6.15):

$$\gamma_0^{-1} L^2 \ll \gamma_+^{-1} \Rightarrow L \ll \sqrt{\frac{T}{\Delta}} e^{\Delta/2T} \quad (6.23)$$

$$\gamma_-^{-1} \ll \gamma_0^{-1} \Rightarrow \frac{T}{\Delta} \ll 1 \quad (6.24)$$

In this regime, the lowest-order processes are of the order of,

$$\gamma_0 \frac{\gamma_+}{\gamma_-} \sim \xi \cdot T \cdot e^{-\Delta/T} \quad (6.25)$$

and we write a phenomenological ground state relaxation rate of the form,

$$\Gamma_{\text{TC}}(\beta, L) \sim \xi \cdot T \cdot e^{-\Delta/T} \cdot L^2 \cdot P_{2D}^\Omega(L). \quad (6.26)$$

Analogous to the phenomenological relaxation rate for the Ising model (i.e., (6.19)), the term $\xi \cdot T \cdot e^{-\Delta/T} \cdot L^2$ encodes the rate at which free quasiparticle pairs are produced. The number

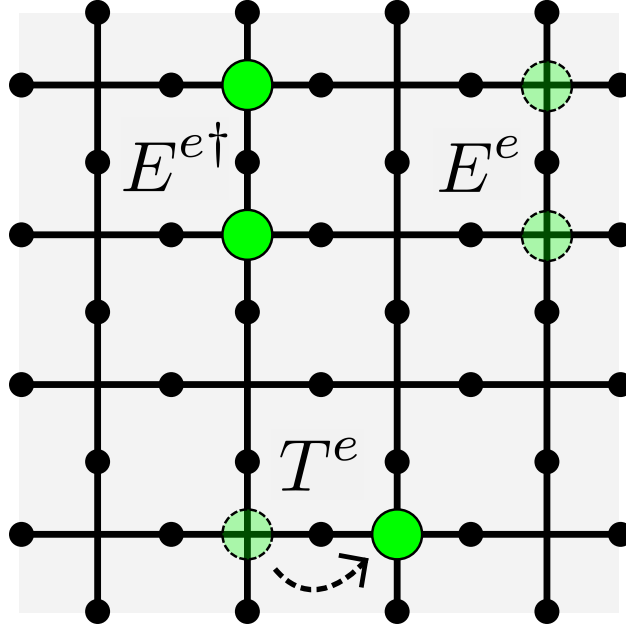


Figure 6.3: A depiction of the operation of the Lindblad operators in the master equation describing the interaction of the toric code with an external bath, as defined in (8.4).

of spins where a defect pair can be created is $N = 2L^2$. The scaling of the topological factor $P_{2D}^\Omega(L)$, which is the probability of a 2D topologically nontrivial walk (i.e., a walk with odd winding number) will control finite-size scaling of Γ_{TC} ; only if $P_{2D}^\Omega(L) \sim L^{-2}$ will the relaxation rate of the toric code be system size independent, as for the classical Ising chain. Note that $P_{2D}^\Omega(L)$ is, in general, a function of temperature (see Appendix A).

High temperature phenomenological dynamics

At high temperatures, the relaxation of a toric code ground state will be dominated by the growing population of quasiparticles, rather than the dynamics of a single quasiparticle pair. In this regime, $\gamma_+ \approx \gamma_- \approx \gamma_0$ and the relaxation rate, i.e., the rate of decay of the expectation values $\langle W_{1,2}^z \rangle$, is due to error strings created across the length of the $W_{1,2}^z$ operator. Consequently we expect the decay to be *linear* in system size:

$$\Gamma_{TH} \sim \gamma_+ L. \quad (6.27)$$

This linear scaling arises from the short time dynamics of the master equation (8.2) [124] and is independent of the topological processes that contribute to $P_{2D}^\Omega(L)$ and dominate the low temperature regime.

Low temperature effective model of ground state transitions

We will now use the form of the scaling of non-trivial annihilation probabilities discussed above to construct an effective low temperature minimal Markov model of the toric code ground state subspace. The ground state Markov model is defined by the master equation

$$\frac{d\mathbf{P}}{dt} = \mathbf{\Gamma}\mathbf{P}(t) \quad (6.28)$$

where $\mathbf{P} = (P_{++}, P_{+-}, P_{-+}, P_{--})$ is the vector of all ground state probabilities and $\mathbf{\Gamma}$ is the matrix of transition rates between ground states.

Here we assume the low temperature form of the transition rate between ground states to take the form of (6.26); correspondingly the matrix elements of $\mathbf{\Gamma}$ take the form

$$\Gamma_{ij} = \lambda P_{ij}^{\Omega} \quad (6.29)$$

where Γ_{ij} corresponds to the transition $i \rightarrow j$, λ is a rate of production of anyon pairs that undergo a nontrivial random walk, and P_{ij} is the probability that a given anyon pair will undergo a topologically nontrivial walk causing the transition $i \rightarrow j$. We take the form of λ to be:

$$\lambda = 2L^2\gamma_+ \left(1 - \frac{\gamma_-}{6\gamma_0 + (2L^2 - 7)\gamma_+ + \gamma_-} \right), \quad (6.30)$$

where $2L^2\gamma_+$ is the rate of pair creation for the entire lattice. The remaining factor in brackets is exactly the probability that an adjacent pair of quasiparticles on an otherwise empty lattice does not annihilate. The numerical factors (i.e., $6, (2L^2 - 7), 1$) simply index the number of edges that can be acted upon by the different Lindblad operators for the lattice configuration with a single pair of adjacent quasiparticles. By detailed balance, the probability of a given Lindblad operator acting on the system (e.g., E^e) is then just the ratio of the rate of that operator (e.g., γ_-) to the weighted sum of the rates of the other available operators, weighted by the number of edges available to each operator (e.g., $6\gamma_0 + (2L^2 - 7)\gamma_+ + \gamma_-$). Thus, λ accounts for the creation rate of quasiparticle pairs that do not immediately annihilate, or those pairs which can generate nontrivial random walks.

The form of P_{ij}^{Ω} is determined by whether the matrix element is relating ground states that differ by a winding on one axis of the torus or on both. We define $P_{\delta_1}^{\Omega}$ and $P_{\delta_2}^{\Omega}$ to be the probabilities of a topologically nontrivial annihilation that has an odd winding about one axis and both axes of the torus, respectively. Then the form of P_{ij} is:

$$P_{ij}^{\Omega} = \begin{cases} P_{\delta_1}^{\Omega} & \text{if } i, j \text{ differ by one winding number} \\ P_{\delta_2}^{\Omega} & \text{if } i, j \text{ differ by both winding numbers} \\ -2P_{\delta_1}^{\Omega} - P_{\delta_2}^{\Omega} & \text{if } i = j \end{cases} \quad (6.31)$$

We may solve this Markov model exactly by integrating (6.28): for $P_{++}(t=0) = 1$ we obtain:

$$\begin{aligned} P_{++}(t) &= \frac{1}{4}(1 + e^{-4tP_{\delta 1}^{\Omega}\lambda} + 2e^{-2t(P_{\delta 1}^{\Omega} + P_{\delta 2}^{\Omega})\lambda}) \\ P_{+-}(t) &= P_{-+}(t) = \frac{1}{4}\left(1 - e^{-4tP_{\delta 1}^{\Omega}\lambda}\right) \\ P_{--}(t) &= \frac{1}{4}(1 + e^{-4tP_{\delta 1}^{\Omega}\lambda} - 2e^{-2t(P_{\delta 1}^{\Omega} + P_{\delta 2}^{\Omega})\lambda}). \end{aligned} \quad (6.32)$$

When $P_{\delta 1}^{\Omega} \approx P_{\delta 2}^{\Omega} \approx P_{2D}^{\Omega}$, we see that all $P(t)$ are well described by an exponential decay with rate $4P_{2D}^{\Omega}(L)\lambda$ at a finite temperature, T (see Appendix 6.8).

6.4 Topologically non-trivial random walks on the torus

Scaling of topologically non-trivial walks

As discussed above, at sufficiently low temperatures on a finite size lattice, we expect the relaxation time of a toric code ground state to depend on the statistics of topologically nontrivial random walks on the torus. In this section we present a numerical study of discrete random walks on a square lattice on a torus using Monte Carlo simulations. Without loss of generality, we may map the processes of pair creation, two-particle random walk, and annihilation to a single random walker undergoing a random walk that starts and ends at the origin. We can compute the probability of a quasiparticle pair generating a transition between ground states after annihilation from the statistics of topologically non-trivial walks of the single walker with odd winding.

To estimate the scaling of $P_{2D}^{\Omega}(L)$, we consider a related quantity: the probability that two random walkers will annihilate after n steps $p(n)$. Topological walks must have radius of L ; given that the radius of a 2D random walk scales as \sqrt{n} , we may assume that topological walks have a minimum number of steps that scales as $n_{\text{topo}} \sim L^2$. $P_{2D}^{\Omega}(L)$ may then be estimated as

$$P_{2D}^{\Omega} \sim \int_{n_{\text{topo}}}^{\infty} dn p(n). \quad (6.33)$$

This rough estimate assumes that all walks larger than a certain length are necessarily topologically nontrivial.

Restricting to a planar square lattice with trivial topology, the annihilation probability $p^p(n)$ can be computed to give the asymptotic behavior for large n as [94]:

$$p^p(2n) \approx \frac{1}{2n(\ln 2n)^2}. \quad (6.34)$$

For small n , the exact result may be computed numerically via a recursion relation [94]. We then can estimate the scaling of P_{2D}^Ω by integrating the planar result:

$$\begin{aligned} P_{2D}^\Omega(L) &\sim \int_{n_{\text{topo}}}^{\infty} dn \frac{1}{n_{\text{topo}} (\ln n_{\text{topo}})^2} = \frac{1}{\ln n_{\text{topo}}} \\ &\sim \frac{1}{\ln L}. \end{aligned} \quad (6.35)$$

We expect then that the poly-log scaling of $p^p(n)$ will lead to an inverse logarithmic finite-size scaling of P_{2D}^Ω . Below we compute P_{2D}^Ω numerically and demonstrate this finite-size scaling empirically.

Monte Carlo study of topologically non-trivial random walks on the torus

In the low temperature limit, quasiparticle dynamics are dominated by trivial events where pairs that are created and immediately annihilate as $\gamma_- \gg \gamma_0$. Additionally, for nontrivial walks, once the anyon pair returns to occupy nearest neighbor sites on the lattice, the pair will annihilate with high probability. To model this low temperature regime, we consider an annihilation to occur as soon as a single random walker returns to a site adjacent to the origin Fig. 6.4. This can be understood as a “zero temperature” limit to the true quasiparticle statistics, as it ignores higher order processes that occur at finite temperature involving quasiparticle trajectories that meet in annihilation geometries, but then do not annihilate. Explicitly, this approximation amounts to taking $\gamma_- \rightarrow \infty$. Additionally, to improve efficiency of the Monte Carlo simulations, we start the walker at one of eight starting positions away from the origin; we account for the relative probabilities for reaching these starting positions via exact enumeration of the combinatorics of short topologically trivial walks (see Fig. 6.4). The random walker undergoes a discrete time random walk on the square lattice on a torus until it returns to one of the four vertices adjacent to the origin.

Using this approach, we compute the probability that two random walkers will annihilate after n steps, $p^t(n)$, and the average number of steps before annihilation, $\langle n \rangle$. For the true finite temperature toric code, the annihilation probability for nearest neighbor quasiparticles is less than one, as it is a function of γ_0/γ_- . The finite temperature probabilities $P_{2D}^\Omega(L)$ may be computed from the zero temperature limit via a “resummation” method described in Appendix 6.8.

Fig. 6.5 shows the annihilation probability on a torus $p^t(n)$ as a function of the number of steps n for the zero temperature model with several system sizes L , as computed via Monte Carlo. We see that $p^t(n)$ agrees with $p^p(n)$ up to a certain value of n for each lattice size. We can therefore define a characteristic “departure time” $n_d(L)$, as:

$$\left| \frac{p^p(2n_d) - p^t(2n_d)}{p^p(2n_d)} \right| = \frac{1}{4} \quad (6.36)$$

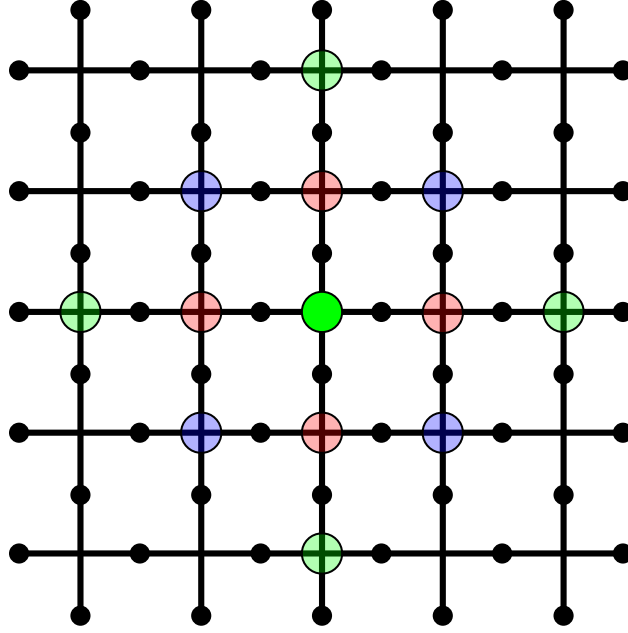


Figure 6.4: The eight starting geometries (translucent green, blue) for Monte Carlo simulations of random walks. The solid green site denotes the origin at which the “fixed” quasi-particle sits. Blue configurations were sampled twice as often as green configurations, owing to the different likelihoods of different starting geometries. The simulation was terminated when the traveling quasiparticle reached one of the translucent red vertices—i.e. an annihilation geometry.

Random walks that annihilate at small n are not sensitive to the topology of the finite-size lattice. Thus, n_d reflects the characteristic number of steps at which the random walk distribution is affected by the finite size torus topology. For $n > n_d$, we see that $p^t(n) > p^p(n)$ up to a characteristic number of steps. We therefore define the “crossing time” n_c where $p^t(n)$ crosses $p^p(n)$ and then drops significantly. Fig. 6.6 shows the scaling of both dynamical quantities, n_c and n_d as a function of system size L . Both are seen to be well described by power laws:

$$n_{c,d} \sim L^{\alpha_{c,d}} \quad (6.37)$$

with $\alpha_c = 2.343 \pm 0.001$ and $\alpha_d = 1.66 \pm 0.04$.

We also compute the initial and final topological sectors of each walk; this allows us to compute the probability of *topologically nontrivial annihilation*, $P_{2D}^\Omega(L)$, where the walk generates a topologically non-trivial path with odd winding. Fig. 6.7 shows the finite size scaling of $P_{2D}^\Omega(L)$; for larger system sizes, we find

$$P_{2D}^\Omega(L) \approx \frac{c_{2D}^\Omega}{\ln(L)} \quad (6.38)$$

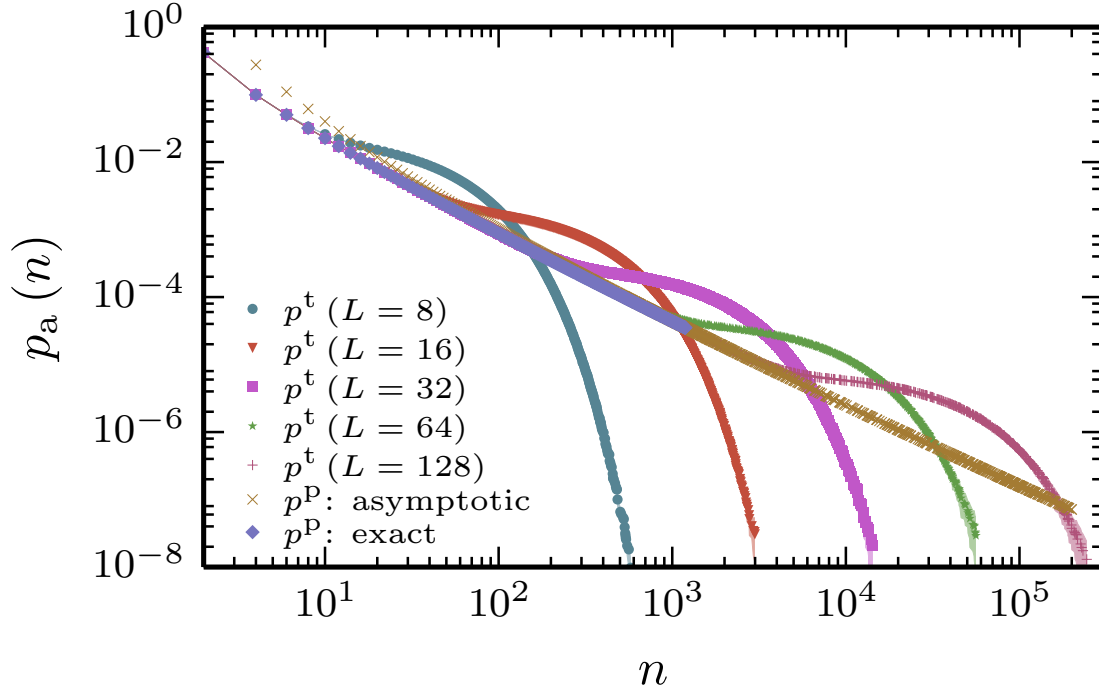


Figure 6.5: Probability of annihilation $p^t(n)$ after n steps on the torus for the model described in section 6.4, as computed via Monte Carlo. We see that the value of $p^t(n)$ agrees with the planar value $p^p(n)$ (see (6.34) and [94]) up until a characteristic value of n where it is possible for the walker to make topologically nontrivial walks on a finite size lattice.

where $c_{2D}^\Omega = 0.472 \pm 0.003$.

We may understand the scaling of $P_{2D}^\Omega(L)$ from an analysis of $p^t(n)$. Since the deviations of $p^t(n)$ from $p^p(n)$ for $n > n_d$ are due to topologically nontrivial walks which contribute to $P_{2D}^\Omega(L)$, we can place approximate bounds on $P_{2D}^\Omega(L)$ from $p^t(n)$. As an upper bound to $P_{2D}^\Omega(L)$, we assume that all walks for $n < n_d$ generate topologically nontrivial windings that contribute to $P_{2D}^\Omega(L)$; therefore we define the integrated probability:

$$P_d \equiv \int_{n_d}^{\infty} dn p^p(n). \quad (6.39)$$

As both p^p and p^t integrate to unity, P_d is approximately equal to the same integral over p^t . P_d includes both topologically trivial walks and topologically nontrivial walks that have even winding; consequently we expect P_d to provide an upper bound to $P_{2D}^\Omega(L)$.

Alternately, we can make the approximation that topologically nontrivial walks are only the *excess* probability for $n_d \leq n \leq n_c$. By making the assumption that $p^t(n > n_c) \approx 0$ (see

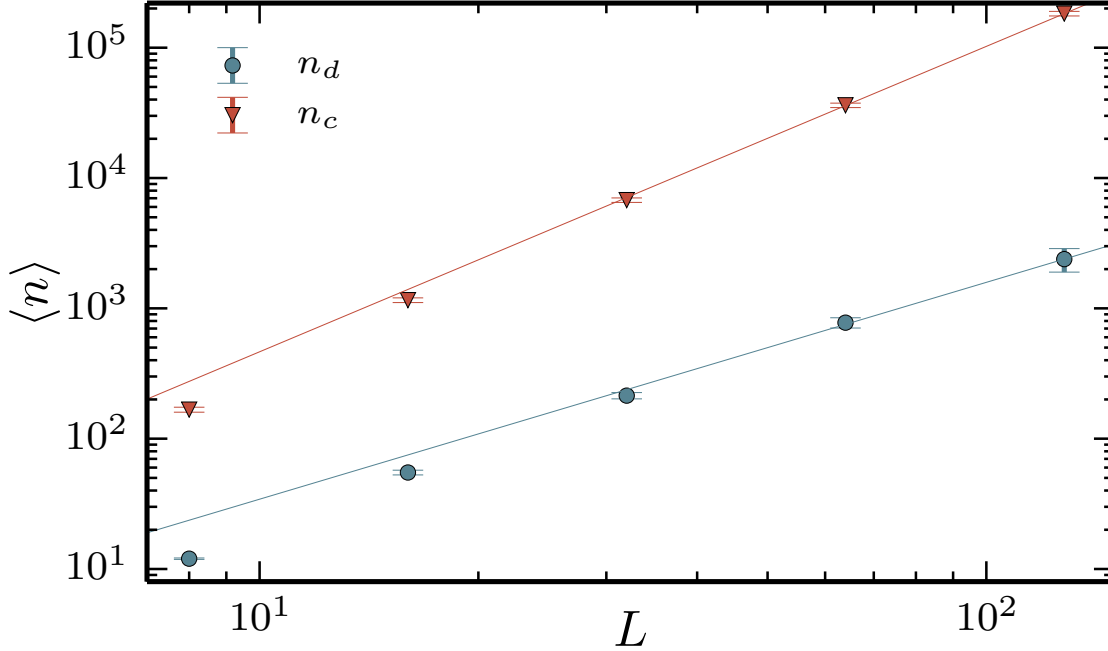


Figure 6.6: The finite-size scaling of the characteristic departure (n_d) and crossing (n_c) times from an analysis of the Monte Carlo data shown in Fig. 6.5. The lines represent the best fit power laws to the three largest system sizes.

Fig. 6.5), we may approximate this excess by:

$$P_c \equiv \int_{n_c}^{\infty} dn p^p(n), \quad (6.40)$$

again relying on the normalization of p^t and p^p . While P_c should over-count the events that contribute to $P_{2D}^{\Omega}(L)$ in the region $n_d \leq n \leq n_c$ (as only odd winding topological events contribute), the approximation $p^t(n > n_c) \approx 0$ leads to an underestimation of $P_{2D}^{\Omega}(L)$, i.e. P_c provides a lower bound on $P_{2D}^{\Omega}(L)$.

Fig. 6.7 shows the finite size scaling of $P_c(L)$ and $P_d(L)$ and confirms that these provide a lower and upper bound to $P_{2D}^{\Omega}(L)$, respectively. Thus as with $P_{2D}^{\Omega}(L)$, we find that $P_c(L)$ and $P_d(L)$ scale as $(\ln L)^{-1}$; the lines in Fig. 6.7 represent a fit to $(\ln L)^{-1}$.

To see the origin of this $(\ln L)^{-1}$ scaling, we can use the asymptotic form of $p^p(n)$ and

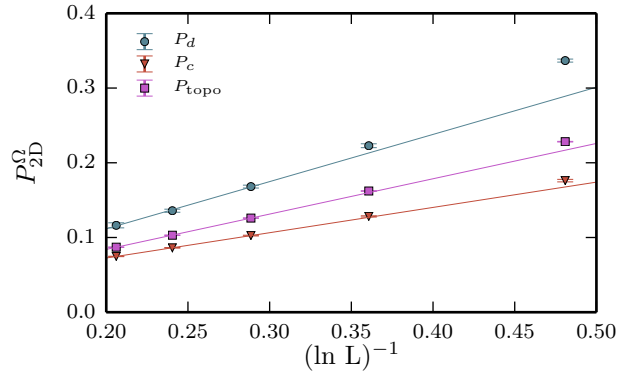


Figure 6.7: The probability of topologically non-trivial annihilations on a torus, $P_{2D}^{\Omega}(L)$ as a function of system size, as computed by Monte Carlo simulations. Also shown are the bounds P_c and P_d , which are computed from $p^t(n)$ and $p^p(n)$, as described in the text. The lines represent fits to $(\ln L)^{-1}$ of the three largest system sizes.

the power law scaling of $n_{c,d}$ from (6.37), to approximate both P_c and P_d :

$$\begin{aligned}
 P_{c,d}(L) &\approx \int_{n_{c,d}}^{\infty} dn \frac{1}{n_{c,d} (\ln n_{c,d})^2} = \frac{1}{\ln n_{c,d}} \\
 &\approx \frac{1}{\alpha_{c,d} \ln L}.
 \end{aligned} \tag{6.41}$$

Consequently, we can understand the origin of the $(\ln L)^{-1}$ scaling which we predicted for $P_{2D}^{\Omega}(L)$ to fundamentally be due to the particular form of the polylog scaling of $p^p(n)$. We note that this inverse logarithmic scaling of $P_{2D}^{\Omega}(L)$ implies a nontrivial finite-size scaling of Γ_{TC} ; indeed for the phenomenological form from Eq. (6.26) we have $\Gamma_{\text{TC}} \sim L^2 / \ln L$.

6.5 Real time Monte Carlo Simulation of the toric code dynamics

Numerical method

We now present Monte Carlo simulations of the real time dynamics of the toric code in contact with an Ohmic bath, as described in section 6.3. We use a continuous real time Monte Carlo method [30] to numerically solve the master equation given in (7.8). We focus on the relaxation dynamics of the system when prepared initially in a pure ground state. We define the operator:

$$\Pi_{++} \equiv \frac{1}{4} (W_1^z + 1) (W_2^z + 1); \tag{6.42}$$

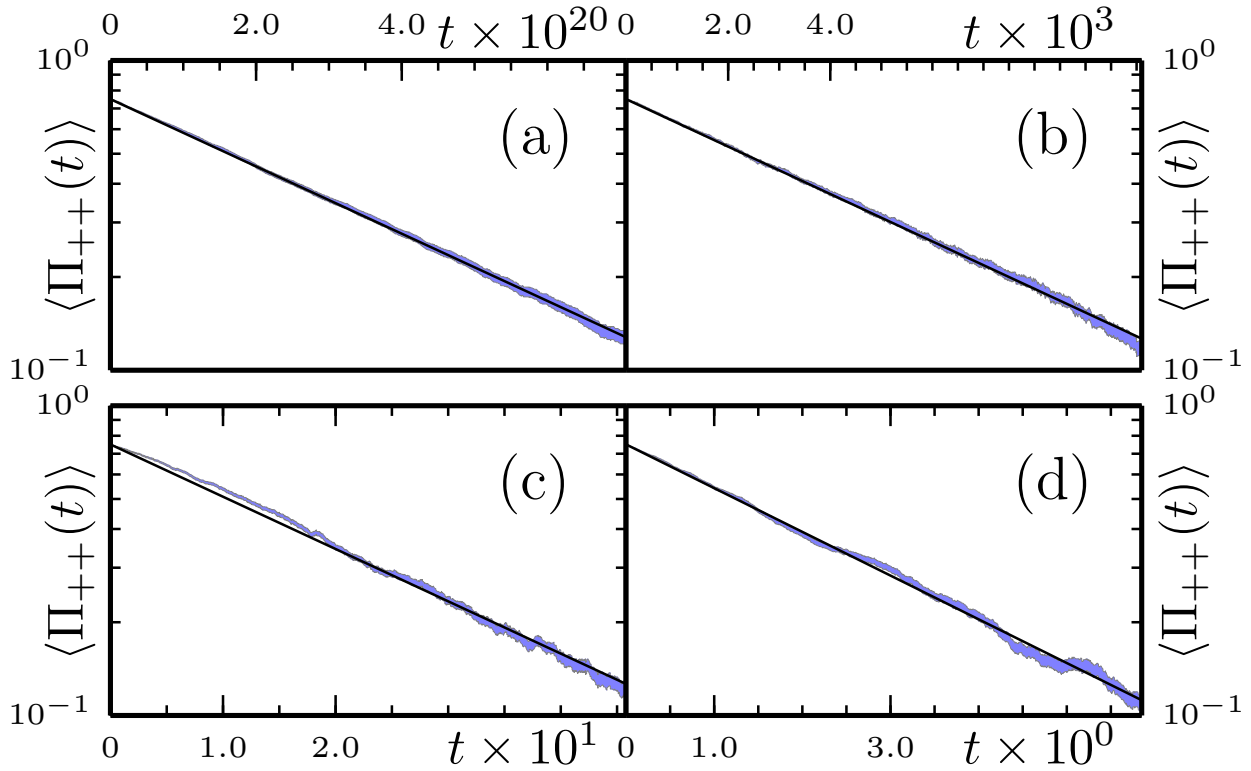


Figure 6.8: Time evolution of the expectation value of the Π_{++} operator (defined in section 6.5) computed via Monte Carlo, where $1/4$ has been subtracted to reveal the exponential decay. These simulations were initialized to a pure ground state with $L = 128$ with $T = \{0.02, 0.08, 0.14, 0.2\}$ respectively for subfigures (a)-(d). The black lines represent exponential fits to the Monte Carlo data. Note the stretched exponential behavior for early times in (c). γ_0 has been set to 1.

which is one for the $|\Psi_0^{++}\rangle$ ground state and vanishes for all other ground states. We then compute the expectation value $\langle \Pi_{++}(t) \rangle$ to study the decay from $|\Psi_0^{++}\rangle$.

The exponential nature of the decay of $\Pi_{++}(t)$ is displayed in Fig. 6.8, where the lines represent exponential fits to the Monte Carlo data. We find such decays are well described by exponential decays at all but intermediate temperatures (see Fig. 6.8 c.), where short time deviations lead to a stretched exponential decay. We fit $\Pi_{++}(t)$ to an exponential:

$$\Pi_{++}(t) = \frac{1}{4} (1 + 3e^{-\Gamma_{++}t}) \quad (6.43)$$

to extract the relaxation rate Γ_{++} . The additional systematic uncertainty of the rate Γ_{++} due to the stretched exponential behavior in intermediate regimes does not appreciably affect the analysis. Fig. 6.9 shows $\Gamma_{++}/e^{-\Delta/T}$ computed for four system sizes over a range of tem-

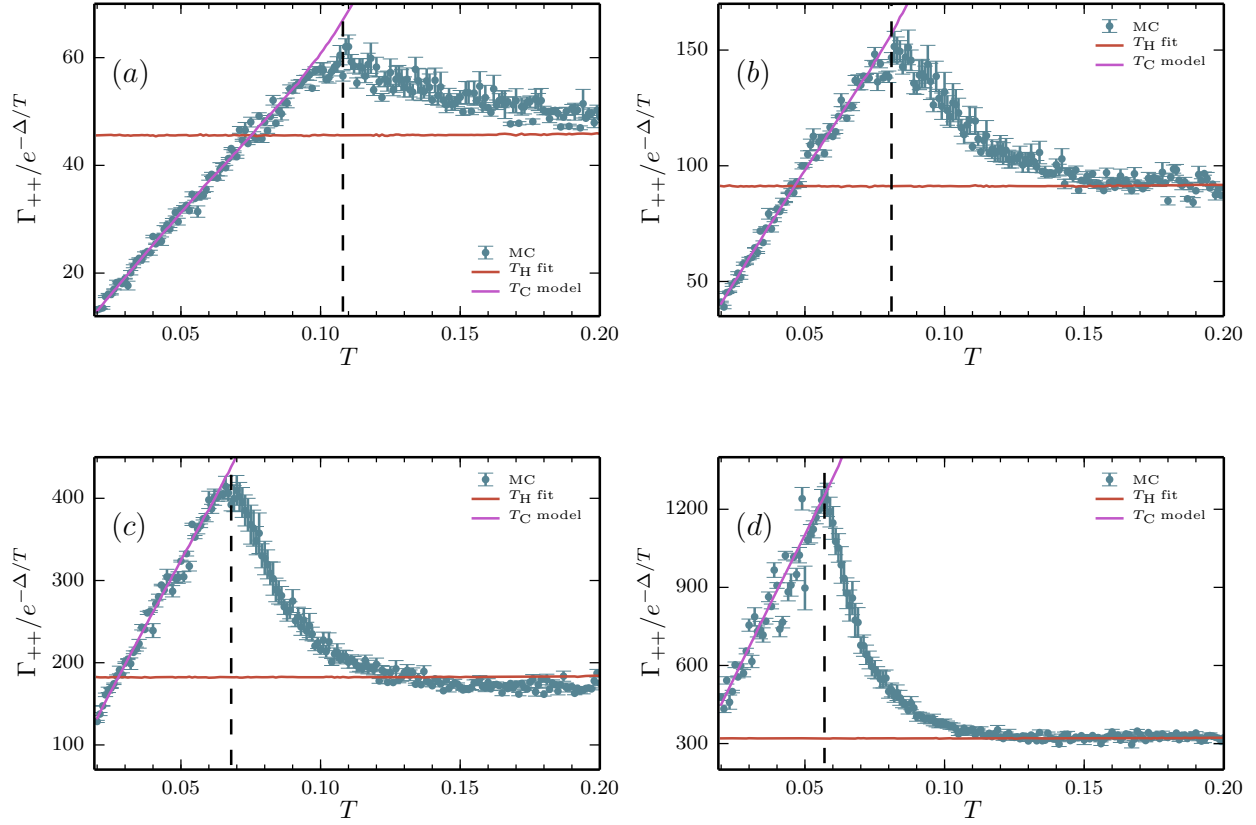


Figure 6.9: Ground state relaxation rates as a function of temperature for system size $L = \{16, 32, 64, 128\}$ corresponding to (a)-(d) respectively. The solid lines are the low temperature phenomenological model, (T_C), (6.26), and the high temperature fit, T_H , (6.27). The vertical dashed line represents the dynamical crossover temperature T_{dyn}^* . Note that $\Gamma_{++}(T)$ is a monotonic, increasing function of T ; the rescaling by $\exp(-\Delta/T)$ generates the nonmonotonicity. Δ has been set to 1.

peratures. We see three temperature regimes for each system size: a low temperature regime where $\Gamma_{++} \sim T e^{-\Delta/T}$, a high temperature regime where $\Gamma_{++} \sim e^{-\Delta/T}$ and an intermediate temperature regime smoothly connecting these two forms of the temperature scaling.

Low Temperature regime

Fig. 6.9 shows the low temperature model predictions of (6.26) as well as the Monte Carlo data. The regime of linear behavior of $\Gamma_{++}/e^{-\Delta/T}$ and corresponding agreement with the effective model at low temperatures allows us to identify this regime as the low temperature regime where the finite-size scaling of the relaxation time is determined by the scaling of

topologically non-trivial random walks. Fig. 6.10 shows the finite-size scaling of Γ_{++} in this low temperature regime, where we have performed a data collapse to remove the leading temperature dependence of (6.28). We find good agreement with the *parameter free* low temperature model (6.28) (with temperature dependent $P_{2D}^\Omega(L)$ obtained by the resummation procedure in appendix A) which has an approximate $L^2/\ln L$ scaling. This non-trivial finite-size scaling is a key feature of this low temperature regime.

On increasing temperature, we can understand the transition out of this low temperature regime as follows. The separation of timescales breaks down as the ratio $L^2 \cdot \gamma_+/\gamma_0$ grows larger. As the temperature increases, the decay rate is significantly affected by multi-pair processes which are not accounted for in (6.28). Interactions between multiple pairs of quasiparticles modify the annihilation probability distributions used in the low temperature model. Additionally, at higher temperatures, as the lifetime of quasiparticle pairs increases, the decay of Π_{++} is sensitive to trivial error strings (i.e., single applications of E^{et}) acting across the edges shared with the winding operators. At higher temperatures, these trivial error strings dominate the decay rate.

High temperature regime

At higher temperatures in Fig. 6.9, we see that $\Gamma_{++} \sim e^{-\Delta/T}$; this is the high temperature regime described by (6.27) where we expect a linear scaling in system size. In Fig. 6.11 we show a fit to the linear finite-size scaling for several different temperatures, where we have scaled the Γ_{++} by the rate of formation of quasiparticle pairs, γ_+ . This one parameter linear fit to the scaled data gives a single functional form for all system sizes and temperatures

$$\Gamma_{++} = c_{T_H} \gamma_+ L, \quad (6.44)$$

where we find the constant $c_{T_H} = 2.5 \pm 0.1$. If only the lowest order process trivial anyon pairs contributed to the decay across both winding operators, we would have $c_{T_H} = 2$. Obtaining a fit to the finite-size scaling with $c_{T_H} > 2$ suggests that higher order processes are also providing significant contributions. The solid red line in Fig. 6.9 shows that this one parameter high temperature fit is in good agreement with the Monte Carlo data in the high temperature regime. We note that the linear finite-size scaling of Γ_{++} is distinct from the $L^2/\ln L$ scaling in the low temperature regime (see Fig. 6.10).

Dynamical Crossover Temperature

Analysis of the results shown in Fig. 6.9 strongly suggests that we can identify two distinct regimes where the relaxation rate is dominated by distinct physical processes. We can therefore define a *dynamical crossover temperature* T_{dyn}^* which signifies the crossover between these regimes. We define T_{dyn}^* as the local maxima on Fig. 6.9 where the linear temperature scaling breaks down; this does not correspond to a maximum of Γ_{++} itself, which is monotonically increasing as a function of temperature, since we have removed the temperature

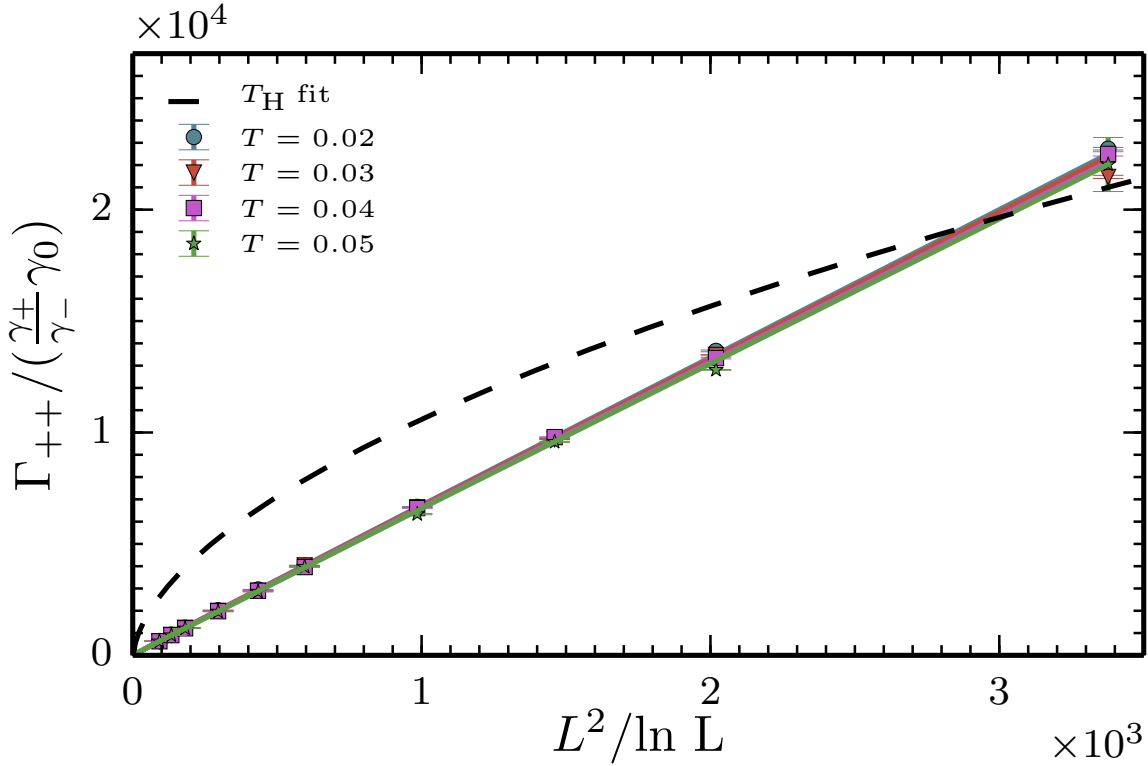


Figure 6.10: Low temperature regime: Finite-size scaling of Γ_{++} in the low temperature regime where we have collapsed the temperature dependence of the data according to (6.26). The solid lines are the low temperature model (T) predictions as described in (6.28); these lines nearly completely overlap due to the weak residual temperature dependence in (6.26). The dotted line indicates the best fit to purely linear scaling in L which is expected in the high temperature regime (T_H).

dependence of the Boltzmann factor by rescaling. Clearly T_{dyn}^* is a function of system size, since the low-temperature regime shrinks as L increases; Fig. 6.12 displays the finite-size scaling of T_{dyn}^* as well as the equilibrium crossover temperature T_{eq}^* computed in [[23]] from the topological entanglement entropy. We find an inverse logarithmic scaling of T_{dyn}^* with system-size, in agreement with the scaling of T_{eq}^* .

6.6 Discussion

We have demonstrated the non-trivial finite size scaling of the relaxation time of the toric code in contact with a thermal reservoir, using numerical simulations of real time dynamics of quasiparticles. We have identified a low temperature regime in which the relaxation dynamics are dominated by topologically non-trivial random walks of quasiparticle pairs;

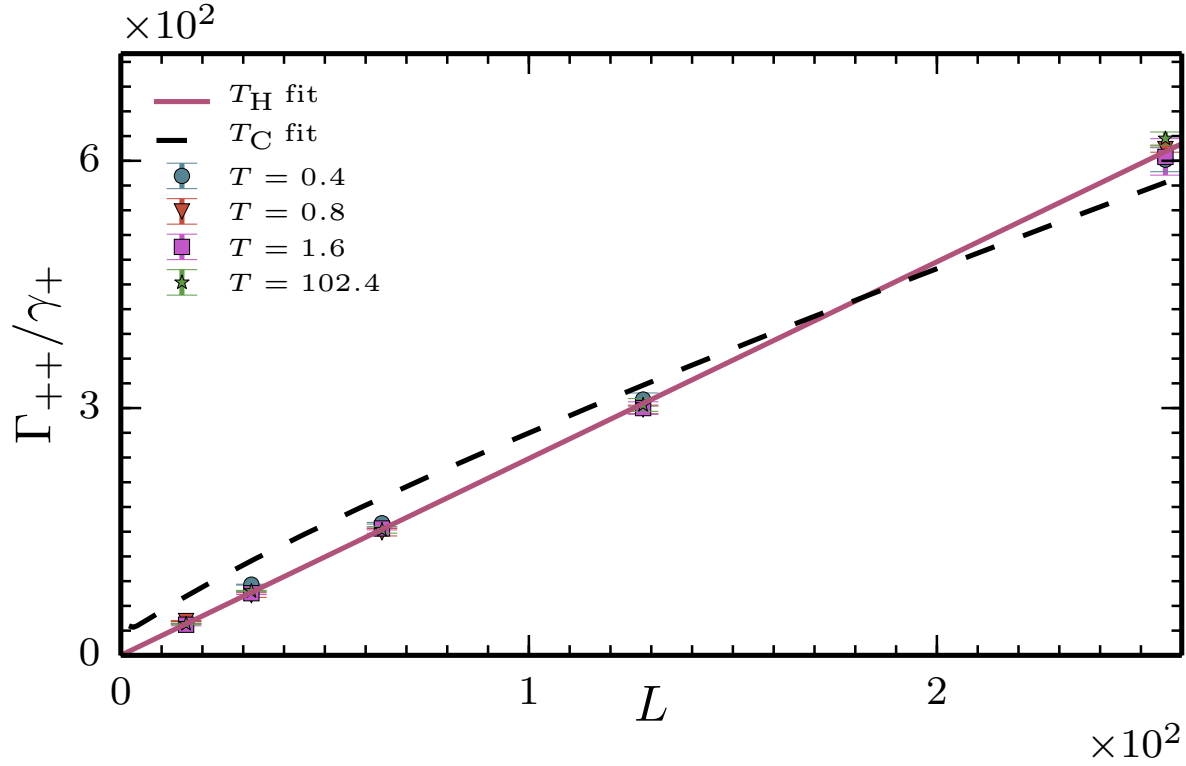


Figure 6.11: High temperature regime: finite-size scaling of Γ_{++} in the high temperature regime. The solid line represents a fit of all high temperature data (T_H) to a linear scaling in L . The dotted line indicates a best fit to the poly-log scaling $L^2/\log L$ which is expected in the low temperature regime (T_C).

consequently the finite-size and temperature scaling of this regime are distinct from the high temperature regime above the crossover temperature. We find that both the finite-size and finite temperature scaling are *stronger* in this low temperature regime than at higher temperatures where the behavior coincides with the expected scaling in the thermodynamic limit[124].

In the low temperature regime, we find the relaxation rate to scale as $L^2/\ln L$, in contrast to the scaling as L above T_{dyn}^* . Consequently, the lifetime of topological qubits will increase faster in this regime as the system size is decreased, than above T_{dyn}^* . We also find that the relaxation rate is suppressed by an additional factor of T in the low temperature regime; the memory lifetime will increase with inverse temperature $\beta = 1/T$ as $\beta e^{\Delta\beta}$, faster than the $e^{\Delta\beta}$ scaling of the lifetime above T^* . We note that the particular form of the additional crossover suppression is dependent on the nature of the bath, since it arises from the temperature scaling of the diffusion rate for the ohmic bath studied here, $\gamma_0 = \xi kT$. In contrast, for a super-Ohmic bath $\gamma_0 = 0$; however, the effective diffusion rate will scale as $e^{-2\Delta/T}$, due to

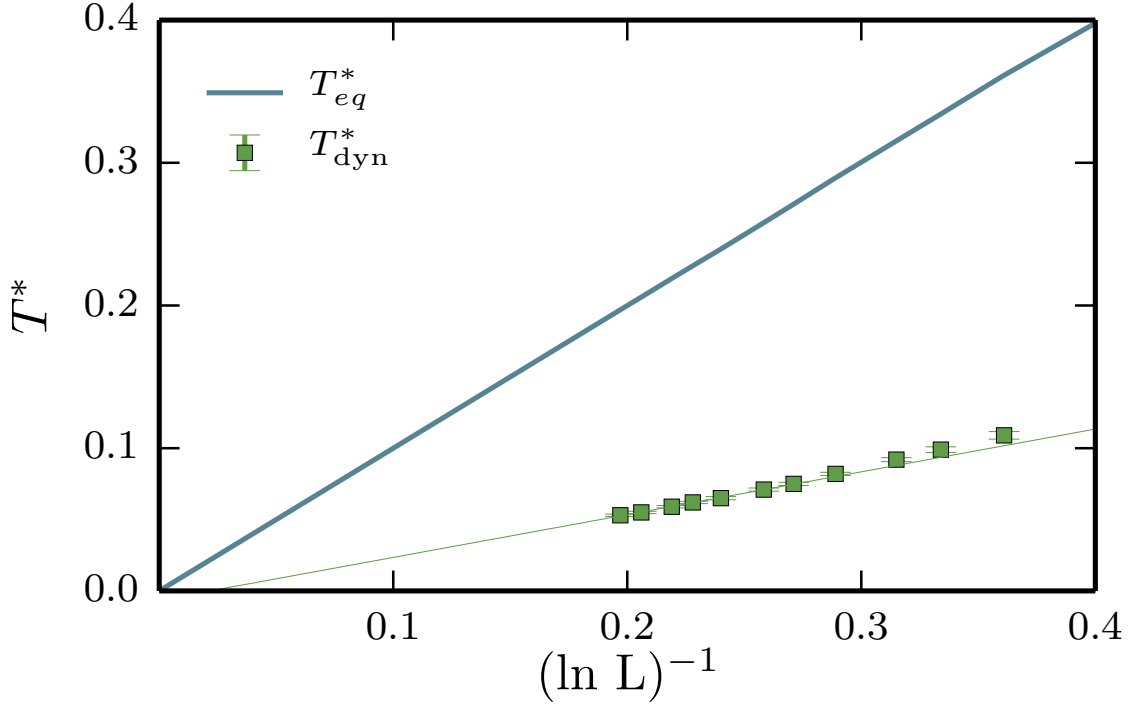


Figure 6.12: The dynamical crossover temperature T_{dyn}^* as a function of system size L . The line represents the fit to $\ln(L)^{-1}$ scaling for the largest system sizes. Also shown is the equilibrium crossover temperature T_{eq}^* as defined in Ref. [23].

indirect hopping of quasiparticles from 2nd order pair creation events [29]. Consequently the low temperature suppression will be even stronger for a super-Ohmic bath.

This work may help guide the design of ground state relaxation of optimal topological qubits. While it is now evident that true topological protection is not achievable for the 2D toric code in the thermodynamic limit, as a practical matter in a finite size realization, one may wish to balance robustness to unitary perturbations, which is maximized by using the largest possible system, against thermal robustness, which decreases with system size. The stronger finite-size and temperature scaling of the relaxation time (corresponding to the quantum memory lifetime) in the low temperature regime suggests that the optimal balance will be achieved below T_{dyn}^* . The corresponding optimal size will of course depend on the prefactors of the scaling relations and will therefore be dependent on both the microscopic form of the coupling to the bath and the unitary perturbations.

Thus, although the topological order required for topological protection of quantum information processing is destroyed at all temperatures in the thermodynamic limit, we have identified a dynamical low temperature regime for finite size systems which may prove prac-

tically useful for quantum information processing.

6.7 Acknowledgments

This material is based upon work supported by DARPA under Grant No. 3854-UCB-AFOSR-0041 and by NSF under Grant PIF-0803429. CDF was supported by the NSF Graduate Research Fellowship under Grant DGE-1106400. CDF also thanks Stefano Chesi for helpful correspondence regarding bath models.

6.8 Resummation method

The probabilities of return were calculated by numerically tabulating the fraction of random walks that arrived at the annihilation geometries depicted in Fig. 6.4. At finite temperature, quasiparticles have a nonzero probability of not annihilating after reaching these positions, and continuing a random walk. Only in the ‘zero temperature’ limit do these quantities represent the true annihilation statistics for the quasiparticles. To distinguish between these, we define $\overline{P}_{\delta 1}^{\Omega}$ and $\overline{P}_{\delta 2}^{\Omega}$ as the “zero temperature” probabilities of return.

To calculate this temperature dependent annihilation probability, we define:

$$\overline{P}_{ij}^{\Omega} = \begin{cases} \overline{P}_{\delta 1}^{\Omega} & \text{if } i, j \text{ differ by one axis} \\ \overline{P}_{\delta 2}^{\Omega} & \text{if } i, j \text{ differ by both axes} \\ 1 - 2\overline{P}_{\delta 1}^{\Omega} - \overline{P}_{\delta 2}^{\Omega} & \text{if } i = j \end{cases} \quad (6.45)$$

$\overline{P}_{ij}^{\Omega}$ represents the transition matrix for a *discrete* Markov chain. This matrix encodes the zero temperature transit probabilities for a quasiparticle pair to meet in an annihilation geometry. To account for the possibility of both annihilation and continued traversal, we define:

$$\Sigma = \begin{pmatrix} (1 - \tau)\overline{P}_{ij}^{\Omega} & \mathbf{0} \\ \tau\overline{P}_{ij}^{\Omega} & \mathbb{I} \end{pmatrix}, \quad (6.46)$$

$$\tau = \left(\frac{\gamma_-}{6\gamma_0 + (2L^2 - 7)\gamma_+ + \gamma_-} \right), \quad (6.47)$$

where τ is the probability that an adjacent pair of quasiparticles annihilates, \mathbb{I} is the 4×4 identity matrix, and where $\mathbf{0}$ represents a 4×4 zero matrix. The initial state vector for this Markov chain represents a single pair of quasiparticles initialized to one of the starting configurations in a given sector. By convention, these are the $++$, $+-$, $-+$, $--$ sectors for the first four entries of the state vector. The latter four entries encode the probabilities of a pair of walkers annihilating in a given sector after some number of steps. The long time steady state solution of this larger Markov chain then determines the temperature dependent probabilities that a given quasiparticle pair causes a transition.

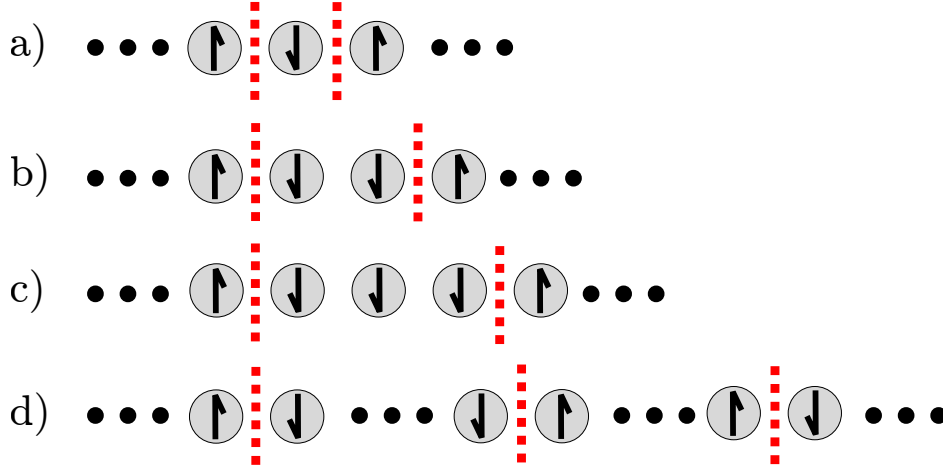


Figure 6.13: Various configurations of domain walls in 1 dimension. a) An “annihilation geometry”, where the domain walls (in red) are separated by a single spin. b) A “free” pair of domain walls. This configuration will annihilate trivially with probability $1/2$. c) Domain walls are further separated. d) Domain walls are separated by half the system size. The rightmost domain wall is the same as the leftmost domain wall due to periodic boundary conditions.

For example, consider a pair initialized to the $++$ sector, with an initial state vector $(1, 0, 0, 0, 0, 0, 0)^T$. The state in the long time limit is

$$\lim_{k \rightarrow \infty} \Sigma^k \cdot (1, 0, 0, 0, 0, 0, 0)^T = (0, 0, 0, 0, 1 - 2P_{\delta 1}^\Omega - P_{\delta 2}^\Omega, P_{\delta 1}^\Omega, P_{\delta 1}^\Omega, P_{\delta 2}^\Omega)^T \quad (6.48)$$

In the higher temperature limit all transition probabilities tend towards $1/4$. The zero temperature limit corresponds to the bare probabilities $\overline{P_{\delta 1}^\Omega}$ and $\overline{P_{\delta 2}^\Omega}$. The temperature dependent $P_{\delta 1,2}^\Omega$ are used in the manuscript in sections III and V. The “zero temperature” limits are used exclusively in section IV.

6.9 Finite size scaling in the 1D Ising model

Here we demonstrate the scaling of P_{1D}^Ω discussed in 6.3. Consider a pair of domain walls on a 1D periodic chain of L classical Ising spins separated by two spins (i.e., configuration (b) in Fig. 6.13). In such a configuration, it is equally likely for the domain walls to move one unit to the left or right. If either of the domain walls is separated from the other by only a single spin, they are in an “annihilation geometry” (i.e., configuration (a) in Fig. 6.13), and in the $T \rightarrow 0$ limit will annihilate with unit probability.

Without loss of generality, fix one domain wall as an “origin”. A pair initially in configuration (b) from Fig. 6.13 will either annihilate with probability $1/2$, or become separated

by at least $2+1$ spins with probability $1/2$. In random walks for which the domain walls become separated by $2+1$ spins (i.e., configuration (c) in Fig. 6.13), the domain walls will either annihilate trivially with conditional probability $1/2$ or become separated by at least $4+1$ spins with conditional probability $1/2$.

In this way, the set of random walks available to a domain wall pair separated by $d+1$ spins can always be partitioned into those that return to the annihilation geometry, and those that separate the domain wall pair by an additional d spins, because the inverse process that results in an annihilation event is a random walk that separates the domain walls by an additional d spins. Once $d+1$ is exactly half the system size, the probability of the domain walls separating by an additional d spins is equivalent to annihilating nontrivially, as the free domain wall “wraps around” and annihilates from the opposite side of the fixed domain wall.

For simplicity, if we suppose the system size is of the form $L = 2^n + 2$ for some positive integer n , then domain walls which are separated by $L/2$ spins (equivalently, $2^{n-1} + 1$ spins) have a conditional probability of $1/2$ of annihilating either trivially or nontrivially. The total probability of the domain walls reaching this configuration is just the product of the conditional probabilities of the domain walls reaching $2+1$, $4+1$, ..., $2^{n-1}+1$ spins of separation. Thus: $P_{1D}^\Omega = 1/2^n$, or by rearrangement: $P_{1D}^\Omega = 1/(L-2)$.

Chapter 7

Autonomous algorithms without measurements

the control structure of a programming language is more than raiment.

King James Programming Tumblr

After tracing the breakdown of the toric code’s resilience to the random movements of quasiparticle excitations, I constructed error correction algorithms that specifically targeted these sorts of error processes. In the spirit of trying to retain as much of the “quantum memory” properties of the toric code as possible, algorithms considered in this chapter do not make use of measurements of stabilizers in any conditional way. The text of this chapter is the manuscript Freeman, Herdman, and Whaley [54].

7.1 Introduction

For the past two decades, significant effort has gone into devising schemes for encoding quantum information in reliable and retrievable forms. Stabilizer error correcting codes are thought to be an effective strategy for performing this encoding, because they allow an efficient means of detecting and correcting errors. Among these, topological stabilizer codes (or topological quantum memories), are particularly promising strategies for storing quantum information due to their intrinsic robustness to errors at zero temperature, their ability to be efficiently implemented via a local Hamiltonian[49], as well as the existence of efficient strategies for performing error detection and correction[125, 36, 57] which have been demonstrated in recent experiments[99]. Several exhaustive studies have been performed on calculating error thresholds for these topological codes, like Kitaev’s toric code, both in the presence and absence of error correcting protocols[13, 125, 50, 36, 135].

However, these topological codes are well known to be poor *passive* quantum memories at finite temperature in less than four spatial dimensions[102, 101, 23, 3, 29, 30, 136, 124, 126, 55] (for a thorough review, see Ref. [22]). For physically realistic coupling to an environment,

local noise processes drive the creation of localized defects. In the absence of an error correcting protocol, the propagation of these defects can then lead to decoherence of the memory. For the case of the toric code, these error strings are particularly pathological, and cause the maximum lifetime of an encoded qubit to decay exponentially with temperature with a timescale independent of system size[3]. While experimentally intractable, fault tolerant topological quantum memories are known to exist in four and six dimensions[35, 9].

On the other hand, a variety of active error correction protocols exist for efficient detection and correction of errors. As long as error rates and the temperature are low or, alternatively, as long as detection and correction are fast enough, the lifetime of these codes can in principle be extended indefinitely. But these decoding strategies implicitly rely on resources that may not always be available or efficiently physically implementable. For example, performing a measurement on a quantum system requires a fresh ancilla qubit for each measurement. Thus, continuously measuring any quantum system requires continuously recycling ancilla qubits for measurement—a procedure which will necessarily be rate limiting for near term quantum architectures[127].

An error correcting strategy for topological codes *without* the need for stabilizer measurement is desirable. At face, ignoring the power of the stabilizer group will assuredly provide a suboptimal strategy. But given limited resources and rates of measurement it is worthwhile to understand the limits of strategies which do not require syndrome measurements, and to determine if such strategies can augment known decoding schemes.

We provide here a new protocol for error correction of pairs of localized defects which modifies an existing dissipative protocol. We achieve this by applying a specially designed periodic sequence of unitary operators to a code of choice. This pattern of operators is designed to encourage defects in the system to dissipate more quickly. In this work, we explicitly treat the theory for the 1D Ising model at finite temperature, and describe how this approach may be extended to other stabilizer codes, such as the toric code. While dissipative protocols have previously been employed to generate hamiltonians[65, 128], to prepare encoded ground states[34], to mediate long range interactions[69, 106, 73], and to “trap” defects[6, 57], a measurement-free protocol that explicitly targets string-like error processes has not been proposed to date. While it does not completely eliminate errors, the protocol presented here provides a significant enhancement of the lifetime of a finite-size system.

It is known that stabilizer Hamiltonians at finite temperature in dimension less than three have a system-size independent upper bound to their lifetime[18, 3, 30, 136, 63, 82, 116, 117]. These “no-go” theorems necessarily limit the extent to which the method proposed here can be carried out. In fact, a size-independent constant enhancement of a system’s lifetime may be the best one can get with a purely local unitary protocol like the one presented here. Thus, this scheme, by itself, will not generate a topologically protected quantum memory at finite temperature for one or two dimensions. It is nonetheless worthwhile to understand how far purely local protocols can be pushed, because a large constant increase in the lifetime of a quantum architecture could mean the difference between a physically realistic architecture that can be fault tolerantly operated versus one that cannot, as discussed in Sec. 7.5.

The rest of the paper is structured as follows: in Sec. 7.2, we briefly review the theory of stabilizer codes, discuss where our strategy falls on the continuum of different active and passive strategies used in the literature for implementing these codes, and sketch how they can be modeled at finite temperature. In Sec. 7.3, we describe how the 1D Ising model can be treated as a stabilizer code and discuss the low temperature dynamics of the model. In Sec. 7.4, we construct our autonomous protocol, built out of local unitary operators, and discuss the scaling behavior of the protocol. We also demonstrate evidence for the enhancement of the lifetime of the 1D Ising model. In Sec. 7.5, we sketch how our protocol generalizes to higher dimensions and to other stabilizer codes, including the toric code.

7.2 Stabilizer Codes

Definitions

In this section, we briefly review the theory of stabilizer error correcting codes[59, 87]. Given n qubits, a collection of operators S_i , and 2^k states $|\psi\rangle_i, i = 1, \dots, 2^k$ which span a subspace in which k encoded qubits are defined, let,

$$S_i|\psi\rangle_i = +1|\psi\rangle_i \quad (7.1)$$

$$[S_i, S_j] = 0 \quad (7.2)$$

for all i, j . Furthermore, suppose there are m error operators $E_j, j = 1, \dots, m$, and that for each of them, there exists some not necessarily unique operator S_j such that

$$\{E_i, S_j\} = 0 \quad (7.3)$$

Stabilizer codes are those collections of states $|\psi\rangle_i$ and operators $\{S_j\}$ which satisfy the above conditions for error operators belonging to some subset of the Pauli group—tensor products of Pauli operators with the identity.

For example, given three qubits, let $|\psi\rangle_1 = |\uparrow\uparrow\uparrow\rangle$ and $|\psi\rangle_2 = |\downarrow\downarrow\downarrow\rangle$. Then the set of operators S_i satisfying (7.1) and (7.2) is $\{\sigma_z\sigma_z I, I\sigma_z\sigma_z\}$. One can easily determine that the set of error operators E_i corresponding to these two *stabilizer* operators is: $\{\sigma_x I I, I\sigma_x I, I I\sigma_x, \sigma_x\sigma_x I, \sigma_x I\sigma_x, I\sigma_x\sigma_x\}$.

More transparently, this 3-qubit stabilizer code encodes two protected states. If some noise source were to apply any single qubit σ_x operator, or any two-qubit $\sigma_x^i\sigma_x^j$ operator, measurement of the set of stabilizer operators would indicate the presence of an error. Furthermore, the code can actually detect *and* correct single σ_x errors. For example, a measurement result of $-1, +1$ of the stabilizers $\sigma_z\sigma_z I$ and $I\sigma_z\sigma_z$, respectively, indicates either an error on the first qubit or two errors on the latter two qubits. In a sufficiently noisy environment, these two errors would be indistinguishable—i.e., degenerate—but for many noise models, the single error situation is much more likely, thus a single σ_x operator applied to the first qubit will more often than not return the qubit back into the protected subspace.

Active State Preparation versus Dissipative Hamiltonian Engineering

Here we will refine our discussion by broadly classifying error correcting approaches into (1) state preparation strategies and (2) Hamiltonian engineering strategies.

The target of both strategies is the same: the generation of an encoded stabilizer state. In state preparation, a stabilizer encoded state is prepared by the application of a sequence of unitaries. However, ignoring noise sources, the natural Hamiltonian which describes the system is $H = 0$. The target of such a strategy is generation of the stabilizer state itself. Implicitly, some sort of active error measurement and correction needs to be performed once the target state is reached.

In contrast, in Hamiltonian engineering approaches[137], the encoded state is reached by implementing a Hamiltonian on a set of qubits which has a stabilizer encoded state as its ground state. The stabilizer state is then preserved by keeping a quantum system at a sufficiently low temperature to suppress errors.

Mixtures of these strategies exist. For example, one could use a Hamiltonian engineering approach to generate a stabilizer encoded state, and then immediately turn off the Hamiltonian once the desired state was reached, preserving the state at further times with active error correction. Alternatively, one could use Hamiltonian engineering to prepare the state, and then use a combination of dissipation with an additional protocol to detect or correct errors. We will focus here on this latter strategy. Specifically, we will be concerned with systems being dissipatively driven towards the ground state of a Hamiltonian which encodes a stabilizer state, and we will build an autonomous error correction protocol to mitigate the ways in which dissipation alone fails to protect the encoded state. Furthermore, we are not explicitly concerned with the process of state creation, and always assume that our systems are initialized to an encoded state.

Error Correcting Master Equation

To dissipatively generate a stabilizer code, one forms the system Hamiltonian as the sum of the stabilizer operators for the code of interest, i.e., $H = -\sum_i S_i$. This guarantees that the ground state of that hamiltonian will be the encoded subspace. Furthermore, this ensures that configurations of the system with errors present are excited states.

To model dissipation in such a code, we employ here a Lindblad master equation. Without loss of generality, but to simplify analysis, we assume that the bath only operates on the system with purely local errors, and that these local errors correspond to the E_i (see (7.3)) of the stabilizer code of interest. The dynamics may be described by the Lindblad equation:

$$\dot{\rho} = \sum_{\omega} 2c_{\omega}\rho c_{\omega}^{\dagger} - c_{\omega}^{\dagger}c_{\omega}\rho - \rho c_{\omega}^{\dagger}c_{\omega}, \quad (7.4)$$

Here ρ is the system density matrix for some candidate system and $\{c_\omega\} = \{\sqrt{\gamma_\omega}L_\omega\}$ are Lindblad operators arising from interactions with a bath, where the L_ω act on the system with characteristic rates γ_ω . Error processes can then be represented by products of the Lindblad operators: $\{c_\omega^1 c_\omega^2 \cdots c_\omega^n\}$.

A necessary condition for error correction to occur to n^{th} order in the error processes is to apply the inverses of the error processes sufficiently rapidly. If we restrict ourselves to stabilizer codes on lattices, then the recipe for error correction is straightforward: measure the stabilizers of the code and apply correction operations conditioned on the results of the stabilizer measurements.

While it is in principle possible to measure all of the stabilizers of a given system simultaneously because they all commute, it will be convenient to decompose a given correction protocol into groups of terms involving operators only acting within a characteristic length scale λ . This is useful because it provides a natural scale for treating stabilizer codes with fixed resources, and it allows the interpretation of different protocols as the implementation of a certain kind of effective long-range interaction.

Error Correction Thresholds and Scaling

Much of the power of stabilizer codes arises from the existence of error thresholds. Specifically, as the stabilizer code is made sufficiently large, the probability of returning to the original code state goes to 1 as long as measurement/correction cycles occur sufficiently fast compared to the threshold error rate. This can give rise to a competition between the resources necessary to perform error correction/detection for stabilizer codes involving many qubits, versus the scaling of the error rate of the code with system size.

For concreteness, consider a stabilizer code, with correction/detection steps idealized by operators O acting over a length scale λ as in Fig. 7.1. How these operators scale with system size depends crucially on the particular code and the specific error correction scheme. For example, it is known[62] that a purely local scheme with λ independent of system size can achieve a threshold in the toric code via a cellular automata construction at the expense of a reduction in the error threshold. This scheme requires measurements and classical decoding. Contrariwise, the naive implementation of minimum weight perfect matching using Edmond’s algorithm to error correct the toric code requires measurements as well as global classical processing—that is, the lengthscale λ of the error correcting operator O which implements Edmond’s algorithm grows with system size, but with the a larger resulting threshold. In this way, global resources can be converted to local resources at the expense of the magnitude of the fault tolerance threshold.

For protocols that both do not yield a threshold and do not use measurements, like ours, the tradeoff between resources and system lifetime is less obvious. In anticipation of the following sections, one might expect a larger λ protocol would dominate over a smaller λ protocol, because the larger λ protocol corrects “more” errors. However, this is not generically the case, and the success of a given error correcting procedure depends nontrivially on λ , L , and the fundamental noise rates of the system.

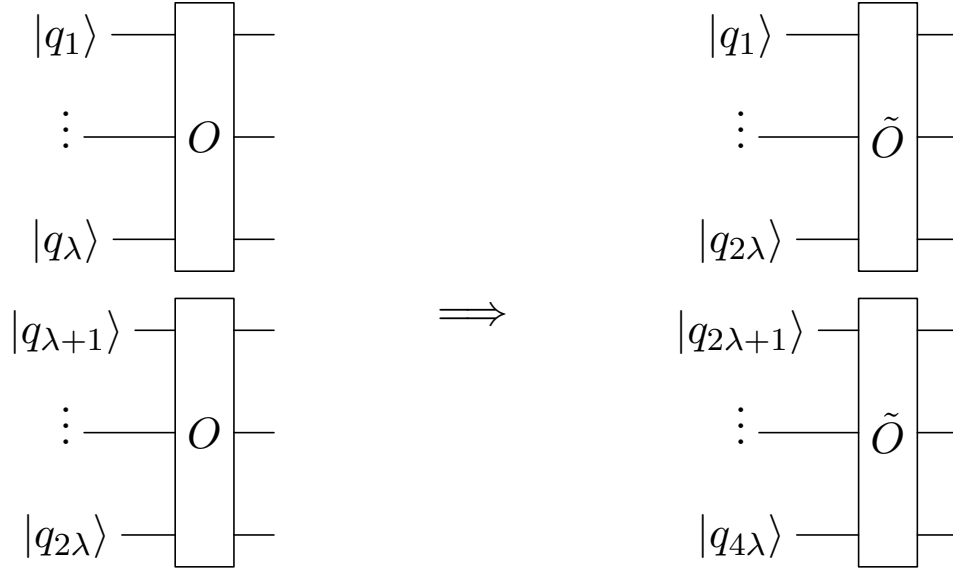


Figure 7.1: A snapshot of a linear stabilizer code with detection/correction operators O . The optimal error correction operator has nontrivial system size scaling, and in general depends on the particular code being used.

We spend a large fraction of the remainder of the manuscript elaborating on our protocol's scaling properties. Because we seek a protocol *without* measurements, we first motivate our construction with the conventional error correction operator for the 1D Ising model in Sec. 7.4, and then detail our measurement-free construction in Sec. 7.4. Next, we explicitly derive how the lifetime of the 1D Ising model is improved by our measurement-free construction, and connect the scaling of the enhanced lifetime to the scaling properties of the protocol with λ in Sec. 7.4. Finally, we elaborate on how our protocol can be generalized to higher dimensions in Sec. 7.5, including its expected scaling properties.

7.3 1D Ising Model

1D Ising Model as a Stabilizer Code

The choice of 3-qubit stabilizer code introduced in 7.2 was deliberate, because it can naturally be extended and interpreted as the ground state of a 1-dimensional Ising model.

$$H_{1DIsing} = -\Delta \sum_{i=1}^L \sigma_z^i \sigma_z^{i+1} \quad (7.5)$$

The ground state subspace of this model is two-fold degenerate and is comprised of the states $|\uparrow \cdots \uparrow\rangle$ and $|\downarrow \cdots \downarrow\rangle$. These ground states are exactly the L -qubit analogues of the 3-qubit code treated previously. These states are stabilized by the set of all adjacent pairwise σ_z operators $\{II \cdots I \sigma_z^i \sigma_z^{i+1} I \cdots I\}$, where i runs from 1 to L . These are precisely the operators appearing in the Hamiltonian of the 1D Ising model. For the remainder of our analysis, we assume without loss of generality that $\Delta = 1$.

Furthermore, σ_x errors are equivalent to excited states. In the simplest case, errors can be corrected by resorting to a simple majority rule—if most spins point in a particular direction, the correction protocol returns the state to the encoded ground state corresponding to that direction.

1D Ising Model at Finite Temperature

By coupling the Ising model to an external reservoir, one might hope to dissipatively drive the 1D Ising model into one of these encoded states. However, the 1D Ising model has no finite temperature ordered phase, so at all finite temperatures, the system evolves towards the unique thermal state. Furthermore, this timescale over which the system relaxes to a thermal state is known to be independent of the size of the chain, given modest bath assumptions[58]. Thus, dissipation by itself cannot protect the 1D Ising model, and an additional protocol needs to be implemented in order to correct thermal errors. While dissipation cannot protect the 1D Ising Model at finite temperature, it is instructive to understand the details of how thermal fluctuations lead to instability in this simple case, because very similar processes are responsible for the instability of many other stabilizer codes at finite temperature. In previous work, we examined the dynamics of this model, as well as of the toric code, at finite temperature[55]. In particular, we identified a low temperature regime where the dynamics are well described by a simple random walk model. We briefly summarize the analysis below.

When studying the error dynamics, it is convenient to consider the dual lattice of the Ising model: we imagine a new 1D lattice with sites interleaved between the sites of (7.5) and associate auxiliary spin values b_i with them. The auxiliary site's spin values are uniquely determined by the products $b_i = \sigma_z^i \sigma_z^{i+1}$, where site b_i defined by this equation sits between site i and $i + 1$. We can identify these extra variables with *domain walls*. If adjacent spin variables disagree, then the auxiliary site sitting between them will have $b_i = -1$. If all but a contiguous block of spins disagree, then all auxiliary sites will have $b_i = 1$ except for those two sites which sit at the two boundaries of the contiguous blocks of spins. Describing the dynamics of these domain walls is equivalent to describing the spin dynamics, because if one knows all the auxiliary variables plus any single spin value, σ_z^i , one can reconstruct all of the remaining spin variables σ_z^j .

For simplicity, we assume a bath that operates on the system only by creating, destroying, or translating domain walls. Then, for sufficiently low temperatures, occasionally the bath will cause an adjacent domain wall pair to appear in the system. Bath fluctuations will cause this pair of domain walls to fluctuate across the system, effectively causing the domain walls to undergo a 1D random walk. When domain walls are adjacent, it is energetically favorable for them to be dissipated. If domain walls fuse *before* traversing the length of the system, the encoded state will be preserved. But if domain walls undergo a random walk such that one winds entirely around the system, this effectively performs an uncorrectable error on the encoded qubit because the system will have transitioned from one encoded ground state to the other encoded ground state[55].

Microscopic Master Equation

When the bath operates on the system with purely local spin flip errors which only create, destroy, and translate domain walls, the Lindblad operators are of the form:

$$\{c_\omega\} = \left\{ \sqrt{\gamma_0} T_b, \sqrt{\gamma_+} D_b^\dagger, \sqrt{\gamma_-} D_b \right\} \quad (7.6)$$

When resolved in the Pauli basis and factored, these operators take a simple form:

$$\begin{aligned} D_b^\dagger &= \frac{1}{4} (I\sigma_x I) (1 + I\sigma_z\sigma_z) (1 + \sigma_z\sigma_z I) \\ D_b &= \frac{1}{4} (I\sigma_x I) (1 - I\sigma_z\sigma_z) (1 - \sigma_z\sigma_z I) \\ T_b &= \frac{1}{4} (I\sigma_x I) (1 - I\sigma_z\sigma_z) (1 + \sigma_z\sigma_z I) \end{aligned} \quad (7.7)$$

A short calculation verifies $\sum_i L_i L_i^\dagger = I$. Physically, these operators represent the creation of a domain wall pair at dual lattice sites b and $b + 1$ (D_b^\dagger), annihilation of a pair of domain walls at dual lattice sites b and $b + 1$ (D_b), and the translation of a domain wall from b to $b + 1$ (the adjoint translates $b + 1$ to b) (T_b). Intuitively, these operators are built from projectors $(1 \pm I\sigma_z\sigma_z)$ which pick out those configurations relevant to each process. For example, the combination $(1 + I\sigma_z\sigma_z) (1 + \sigma_z\sigma_z I)$ projects onto the subspace with no domain walls between the first and second or second and third spins. The leftmost ($I\sigma_x I$) term then performs the spin flip, effectively creating a domain wall.

Additionally, these operators only connect diagonal elements of the density matrix to other diagonal elements. This reduces the time evolution of the diagonal matrix elements to a classical master equation:

$$\begin{aligned} \frac{dP_n}{dt} = & \gamma_0 \sum_{n_0} (P_{n_0} - P_n) + \sum_{n_+} (\gamma_- P_{n_+} - \gamma_+ P_n) \\ & + \sum_{n_-} (\gamma_+ P_{n_-} - \gamma_- P_n) \end{aligned} \quad (7.8)$$

Where $P_n := \rho_{nn}$, and the indices n_0, n_+ , and n_- run over those eigenstates connected to n^{th} state by a single application of the operator T_b, D_b^\dagger , or D_b , respectively. The rates with which these operators are applied, i.e. γ_0, γ_+ , and γ_- , are set by the specific choice of bath model. For simplicity, we consider here a Markovian bath. The rates of such a bath are determined by:

$$\gamma(\omega) = \xi \left| \frac{\omega^n}{1 - e^{-\beta\omega}} \right| \quad (7.9)$$

Where ω is the amount of energy exchanged when a particular Lindblad operator acts on the system. The relevant rates for our study are γ_- , γ_+ , and γ_0 , corresponding to domain wall pair annihilation, pair creation, and translation, with energy scales of -4Δ , 4Δ , and 0, respectively. Different n correspond to different types of baths—for $n = 1$ the bath is Ohmic, and for $n \geq 2$, the bath is Superohmic[29]. For our purposes, it will be more convenient to treat γ_0 as a tunable parameter to study the scaling behavior of our protocol. Qualitatively, γ_0 scales linearly with T for Ohmic baths and equals zero for Superohmic baths. For simplicity, we work in units where $\xi = 1$.

For more details of the master equation approach used to study this model, see Ref. [55].

7.4 The Protocol

Protocol Considerations for the 1D Ising Model With Measurements

While we are interested in protocols that do not use measurements, we first review a conventional measurement-based protocol for performing error correction in the 1D Ising model.

In the absence of resource constraints, it is straightforward to construct the operators which correct errors in the 1D Ising model. According to the schematic shown in Fig. 7.1, the $\lambda = 2$ analogue of O is simply the domain-wall annihilation operator, D_b from (8.4). In more generality, for larger λ the corresponding O is the operator which, given an even number of domain walls, annihilates all domain wall pairs in the region being operated upon. For example, the circuit for the $\lambda = 3$ version of this operator is depicted in Fig. 7.2. Note that for an odd number of domain walls, there is not an unambiguous choice for how to annihilate domain walls because a free, unpaired domain wall is always left over.

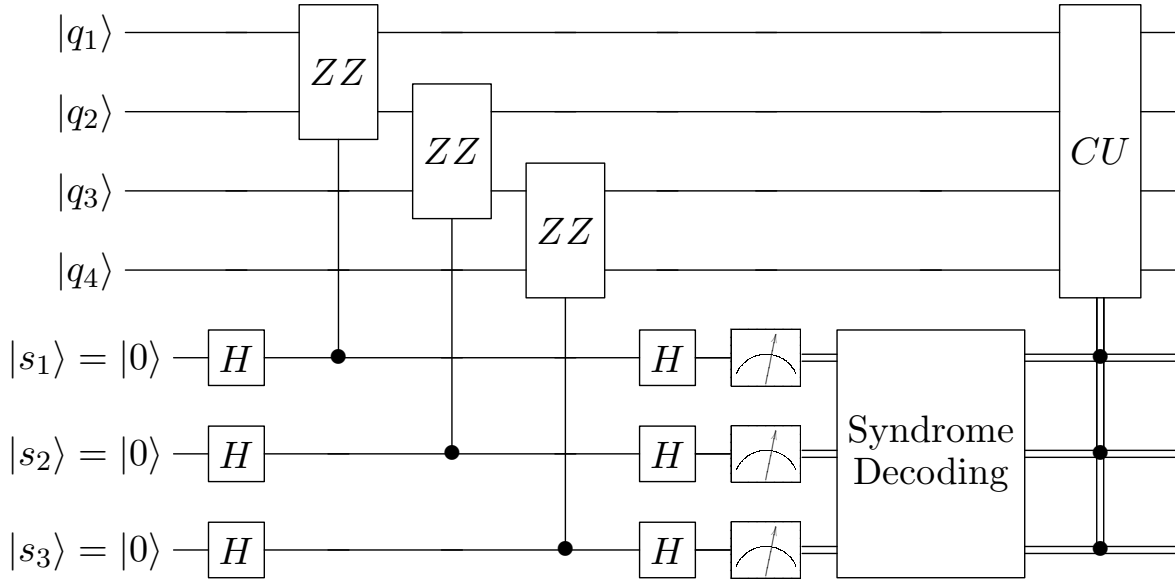


Figure 7.2: Circuit for performing error suppression for a subregion of the 1D Ising model. Qubits q_1 through q_4 are lattice sites on an Ising chain. s_1 through s_3 are ancillas used to read out the syndrome measurements of ZZ on the nearest-neighbor Ising lattice sites. Based on the results of the syndrome measurements, the conditional unitary operator CU corrects the errors present. A table which defines CU is provided in Appendix 7.8. This entire circuit represents a possible realization of one such operator O from Fig. 7.1.

Thus, the most straightforward conventional error correcting protocol is simply to measure the system's stabilizers often enough that one can unambiguously locate pairs of domain walls and then perform correction operations, as indicated in Fig. 7.2. This can be represented by a sequence of measurement operators, the stabilizer for the 1D Ising model, S_i , interleaved by conditional application of corrective unitaries: DSWAP and DWALL. These operators have the following representation in the Pauli basis:

$$\text{DWALL} = \frac{1}{2} (III + I\sigma_x I - \sigma_z I\sigma_z + \sigma_z \sigma_x \sigma_z) \quad (7.10)$$

$$\text{DSWAP} = \frac{1}{2} (III + I\sigma_x I + \sigma_z I\sigma_z - \sigma_z \sigma_x \sigma_z) \quad (7.11)$$

where there is a pair of these operators for each triple of lattice sites. In Fig. 7.2, the details of the 4-qubit operator O are abstracted away (see Appendix 7.8), but it can be decomposed into applications of DSWAPs and DWALLs, conditioned on syndrome measurements. A simple calculation shows $[H_{\text{Ising}}, \text{DSWAP}] = 0$ and $[H_{\text{Ising}}, \text{DWALL}] = +4\Delta$.

Intuitively, DWALL destroys (creates) a domain wall pair at the dual lattice site in-between the three qubits being operated on if and only if a domain wall pair is present (or, all of the spins are aligned), respectively. DSWAP translates a domain wall, either left or right if and only if a single domain wall exists between the 3 spins being operated on.

Error correction protocols for the 1D Ising model can thus be understood as procedures for efficiently dissipating domain walls before they propagate too far. We leverage this intuition in the following sections to build a procedure that dissipates domain walls *without* explicit knowledge of the locations of those domain walls.

A Protocol Without Measurements

Because we seek a protocol *without* explicit measurements, the natural operators for such a procedure are DWALL and DSWAP. DWALL is inconvenient, both because the bath already acts to dissipate excitations and because it can lead to the generation of extra, uncontrolled domain walls more easily than the DSWAP operator. Consequently, we only use DSWAPs in our protocols.

If we restrict our attention to the low temperature regime with the rate assumption $L\gamma_+ \ll 1$, then the lifetime of the Ising chain is governed by the dynamics of single pairs of defects. For error correcting purposes, it is convenient to classify the common geometries of pairs of domain walls. First, *correctable errors* are those errors for which the pair of domain walls is not yet separated by $L/2$ or more. *Non-correctable errors* are those domain wall configurations in the complement of this set. In the language of error correction, the distance for this code is $\lfloor L/2 \rfloor$ —more transparently, correctable errors are those errors which will be correctly matched by a perfect decoder. Furthermore, we need to distinguish between *trivial* and *nontrivial* defect pairs. A domain wall pair is trivial if two domain walls sit on neighboring dual lattice sites. Again, assuming we operate in the low temperature regime, these trivial defect pairs annihilate with rate γ_- —that is, much faster than other time scales of the problem. Nontrivial pairs are those pairs which are not on neighboring dual lattice sites.

Designing a successful protocol for the Ising model amounts to designing a sequence of DSWAPs that efficiently dissipates nontrivial, correctable defect pairs. If we let χ be the rate at which DSWAPs can be applied, then we expect an enhanced lifetime given the following rate assumptions:

$$\gamma_- \gg \chi/O(\text{poly}(L)) \sim \gamma_0 > \gamma_+. \quad (7.12)$$

To wit, DSWAPs are applied at a rate much slower than the inherent annihilation rate of the system—this is so DSWAPs do not turn trivial defect pairs into nontrivial defect pairs. Furthermore, χ is chosen to be close to the inherent translation rate so that correctable, nontrivial defect pairs can be brought adjacent to one another and then be dissipated by the bath before they have time to translate out of the correctable range of the protocol. The $O(\text{poly}(L))$ factor multiplying χ accounts for the fact that different protocols require

some polynomial in L number of swaps to sweep across the entire lattice. For a proof of the polynomial scaling in L , see Appendix 7.9.

In the absence of a corrective protocol, this intrinsic hopping rate of the Ising model gives rise to a simple, background error rate[58, 55],

$$\Gamma_0 = \frac{\gamma_0}{1 + e^{\Delta/T}} \quad (7.13)$$

Protocol Construction

In this section we construct an autonomous error correction protocol for the 1D Ising model with a variable length-scale λ . The design of the protocol reduces to attempting to perform a sequence of DSWAPs that will necessarily cause any arbitrarily placed pair of domain walls within a region of length 2λ to become neighbors. We refer the reader to Appendix 7.9 for a more complete discussion of this strategy.

There are a variety of ways to construct protocols which achieve this in a number of DSWAPs that scales polynomially in the length of the system. Here we focus on protocols which we call λ -mixing. By definition, these are protocols which, in the absence of errors, never translate domain walls a distance λ or greater. For an Ising model of length L , λ runs from 1 to $\lfloor L/2 \rfloor$. In the language of error correction, the protocol can be designed to correct errors of distance 1 to distance $\lfloor L/2 \rfloor$.

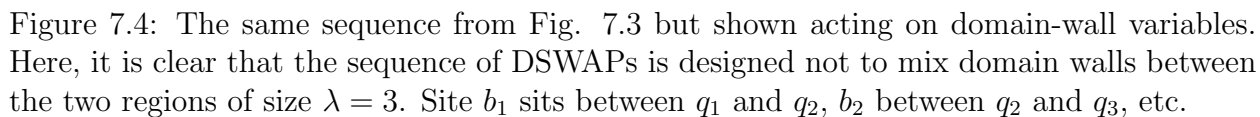
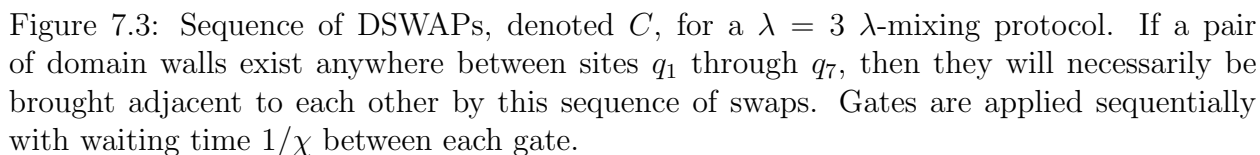
First, the dual lattice is subdivided into non-intersecting subregions of length λ . Then, two adjacent regions are chosen, and a λ -mixing protocol is applied over that subregion of total length 2λ . DSWAPs are chosen to move defects towards the shared boundary of the two regions, but not to mix defects between the boundaries. The non-intersection of the two regions is crucial: if the protocol did not have this feature, it would actually *increase* the error rate, effectively increasing the inherent translation rate, and thus diffusion rate of defects in the system. Fig. 7.3 depicts a circuit for this protocol for $\lambda = 3$ and Fig. 7.4 depicts the same circuit acting on the domain wall variables. Fig. 7.5 illustrates a snapshot of this entire procedure for a representative error process involving two domain walls sitting in neighboring λ -domains.

This circuit should be reminiscent of the cartoon sketched in Fig. 7.1. For our purposes, the operator O from Fig. 7.1 is the full sequence of DSWAPs in Fig. 7.3 dressed by the probabilistic action of creation/annihilation/translation operators by the bath on the system.

We provide code for this algorithm in the Appendix 7.10, including how the λ -mixing subprotocols are constructed.

Error Modes and Scaling

In this section, we examine how uncontrollable thermal errors lead to loss of the qubit in the presence of the protocol.



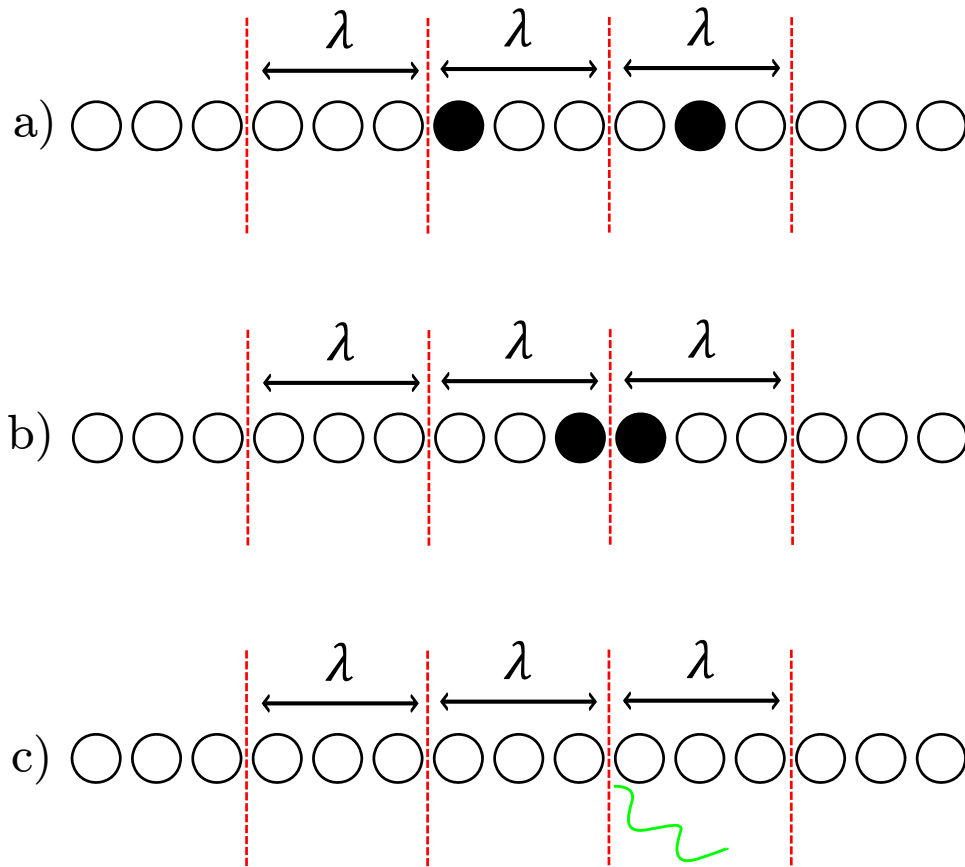


Figure 7.5: One possible snapshot of the error correction process. *a)* depicts a system with two domain walls present, each sitting in adjacent λ -domains. *b)* depicts the state of the system after the protocol has been applied—domain walls have been shuttled to the shared boundary. In *c)*, the bath dissipates the domain walls, and the system returns to the ground state.

In the presence of a corrective protocol, and assuming the correction rate χ is close to the translation rate of the system but still much less than the annihilation rate, the rate of the lowest order error process is given by,

$$\Gamma_{\text{Cyc}} = L\gamma_+ \frac{\gamma_0}{\lambda\chi} \frac{\gamma_0}{\gamma_-} \frac{1}{L-2-2\lambda} \frac{f(\lambda)}{L} \quad (7.14)$$

This rate is the product of (i) the baseline production rate of defect pairs, $L\gamma_+$, (ii) the probability of a defect pair not immediately annihilating, $\frac{\gamma_0}{\gamma_-}$, (iii) the probability of a defect exiting a corrective region, $\frac{\gamma_0}{\lambda\chi}$, (iv) the probability of a nontrivial random walk across the chain $\frac{1}{L-2-2\lambda}$, divided by a factor proportional to the number of correcting regions on the lattice. Thus, for fully parallel application $f(\lambda) \propto \lambda$. Without the protocol, the probability that a pair of domain walls undergoes a random walk that winds around the entire system scales like $\frac{1}{L-2}$, but when the protocol is implemented, the effective lattice size is slightly reduced: the particle need only come within approximately a distance 2λ of its partner for the protocol to fuse them. Thus, Γ_{Cyc} is the leading order estimate for the rate at which domain wall pairs are “missed” by the protocol, and cause uncontrolled transitions between the two ground states of the model.

This effective rate is valid as long as χ is fast enough to compete with γ_0 , but not so fast as to compete with pair annihilation, γ_- , and other higher order processes in $\frac{\gamma_0}{\gamma_-}$ and $\frac{\gamma_0}{\chi}$. Further, we assume that random walks occur much faster than the intrinsic creation rate, or $\gamma_0/L^2 \gg \gamma_+$, and that we can model the random walk as occurring instantaneously—for sufficiently large lattice sizes, the breakdown of this assumption would introduce an additional γ_0 dependence into this error rate to account for the nonzero amount of time it takes defects to traverse the lattice. For regimes studied here, γ_0 and χ run from 10,000 to 100,000 times faster than the intrinsic pair creation rate, γ_+ .

It might be tempting to examine the form of (7.14) and expect that errors vanish in the limit of $\gamma_0 \rightarrow 0$, but a new effective translation rate appears once $\gamma_0 \ll \gamma_+$. In this regime, two pairs of domain walls can appear next to one another, and a consecutive annihilation event produces a lone of pair of domain walls separated by two dual-lattice sites. In this way, an effective translation rate is set by the rate at which these doubled-pair creation events occur. We do not consider this limit further, but it is the natural error process for superohmic baths at low temperature.

To model the breakdown of (7.14) as χ is varied, we can approximate the lifetime, $\frac{1}{\Gamma_{\text{Cyc}}}$, as being effectively reduced by some factor proportional to χ :

$$\frac{1}{\Gamma_{\text{Cyc}}} \rightarrow \frac{1}{\Gamma_{\text{Cyc}}} (1 - \chi g(\lambda, L, \gamma_0, \gamma_-, \gamma_+) + O((\frac{\chi}{\gamma_-})^2 + (\frac{\chi}{\gamma_0})^2)), \quad (7.15)$$

with $g(\lambda, L)$ a protocol-dependent scaling function. Heuristically, for fixed λ , one expects that g should scale linearly with the number of parallel domains of size λ because, for twice as many domains, twice as many pairs will be pulled apart by the protocol that would have otherwise fused. At the same time, for a fixed number of domains, i.e. fixed $\frac{L}{\lambda}$, any given

pair of lattice sites is only ever operated on by a DSWAP for a fraction of the corrective cycle. So, for fixed χ and fixed $\frac{L}{\lambda}$, as λ is increased, domain walls may spend a longer amount of time sitting on a boundary before being caught by the protocol. For the protocol used in this paper, this is cubic in λ . Thus,

$$g(\lambda, L, \gamma_0, \gamma_-, \gamma_+) \propto g(\gamma_0, \gamma_-, \gamma_+) \lambda^3 \frac{L}{\lambda} = g(\gamma_0, \gamma_-, \gamma_+) \lambda^2 L \quad (7.16)$$

This scaling behavior suggests a critical cycling rate, χ_c , at which the lifetime is maximally improved by the protocol. Differentiating (7.15) with respect to λ yields the critical rate, up to the rate function g ,

$$\chi_c = \frac{1}{2\lambda^2 L g(\gamma_0, \gamma_-, \gamma_+)}, \quad (7.17)$$

where any residual prefactors and terms involving γ_0 , γ_+ , and γ_- have been absorbed into g .

Memory Enhancement and Scaling

We now present numerical results demonstrating the enhanced lifetime of the Ising Model when subjected to λ -mixing protocols in serial and in parallel. For serial application, only a single corrective operation was applied every $1/\chi$ units of time. For parallel application, $L/(2\lambda)$ simultaneous corrective operations were applied every $1/\chi$, where each operation acted on a nonintersecting region of length (2λ) .

For the following analysis, we define the *lifetime* as the average time it takes a 1D Ising model initialized to the spin up state to transition to the spin down state. In the absence of the protocol, that is, in the low- χ limit, this lifetime asymptotes to approximately the lifetime given by (7.13).

For the details of the Monte Carlo algorithm, see Ref. [55]. The only nontrivial choice required at the level of simulation is how to treat the competition between the application DSWAPs and bath operators. For simplicity, we assume if a bath operator takes longer than $1/\chi$ to occur, that the DSWAP occurs unhindered. Likewise, if a bath operator takes less than $1/\chi$ to occur, the transformation associated with that bath operator occurs unhindered, be that a pair creation, pair annihilation, or single translation. More complicated choices could be made, like choosing a probabilistic failure rate of a DSWAP as a function of the ratio of the competing timescales, but we do not expect the result of a such a treatment to greatly affect our analysis.

Fig. 7.6 depicts the scaling of the 1D Ising model's lifetime with λ at fixed L , where a smaller λ results in more domains being operated on in parallel. Specifically, for parallel simulations, the protocol was performed simultaneously on $L/(2\lambda)$ domains. These domains were chosen such that DSWAPs were only being applied on non-overlapping regions of characteristic size λ . Here, increasing parallelization manifestly increases the lifetime of the model. For small χ , the protocol does nothing, and the memory converges to the value of

the memory in the absence of any corrective protocol, i.e. (7.13). For χ approaching γ_- , the protocol begins to compete with the process of pair annihilation, and begins turning trivial defect pairs into nontrivial pairs. This actually reduces the lifetime below that of the protocol-free value. In the intermediate regime, the optimal lifetime grows linearly with the number of parallel blocks employed in the algorithm. For this particular protocol, the number of parallel blocks was $48/\lambda$.

Fig. 7.7 depicts the scaling of lifetime with λ , as in Fig. 7.6, but for a serial application of the protocol. For serial application, only a single DSWAP operator ever operates on the system over a timescale χ^{-1} . Decreasing λ also manifestly increases the maximum enhanced lifetime of the protocol. Thus, for fixed-resource architectures, smaller λ necessarily outperforms larger λ implementations.

Fig. 7.8 and Fig. 7.9 depict the scaling of the lifetime with L at fixed λ for parallel application. Remarkably, the L dependence of the models can be completely removed by rescaling the data by (7.14), and rescaling χ to χL as depicted in Fig. 7.9. This rescaling reveals the turnaround in the scaling of the lifetime for $\chi L/\gamma_0 = 54.2 \pm 2.8$, whereafter it transitions from linear scaling in χ to a power law decay.

Fig. 7.10 depicts the scaling of the critical cycling rate χ_c versus the inverse system length $1/L$. This scaling agrees with the error ansatz of (7.17).

We note that substituting χ_c from (7.17) into (7.14) yields $1/\Gamma_{\text{Cyc}} \sim \frac{L-2-2\lambda}{L}$ —or that the maximum enhanced lifetime is asymptotically independent of system size. Thus, even for much larger system sizes, the maximum achievable lifetime will not greatly exceed the maximum lifetime for the $L = 192$ result in Fig. 7.8.

7.5 Higher Dimensions and Generalization

The Toric Code

The argument and construction from the previous section immediately generalizes to any higher dimensional stabilizer codes with stringlike error operators. The immediate analogue is Kitaev's Toric Code, whose Hamiltonian is defined as a sum over vertex and plaquette operators acting on the edges of a square lattice,

$$H_{\text{TC}} = -J_e \sum_v A_v - J_m \sum_p B_p, \quad (7.18)$$

$$A_v \equiv \prod_{j \in v} \sigma_j^z, \quad B_p \equiv \prod_{j \in p} \sigma_j^x, \quad (7.19)$$

The low temperature dynamics of the toric code are governed by the proliferation of localized excitations that are created by string-like error operators, with dynamics similar to those of the 1D Ising model. However, toric code dynamics differ in two ways: first, there are now two types of defects in the toric code—defined as -1 eigenstates of the A_v and B_p

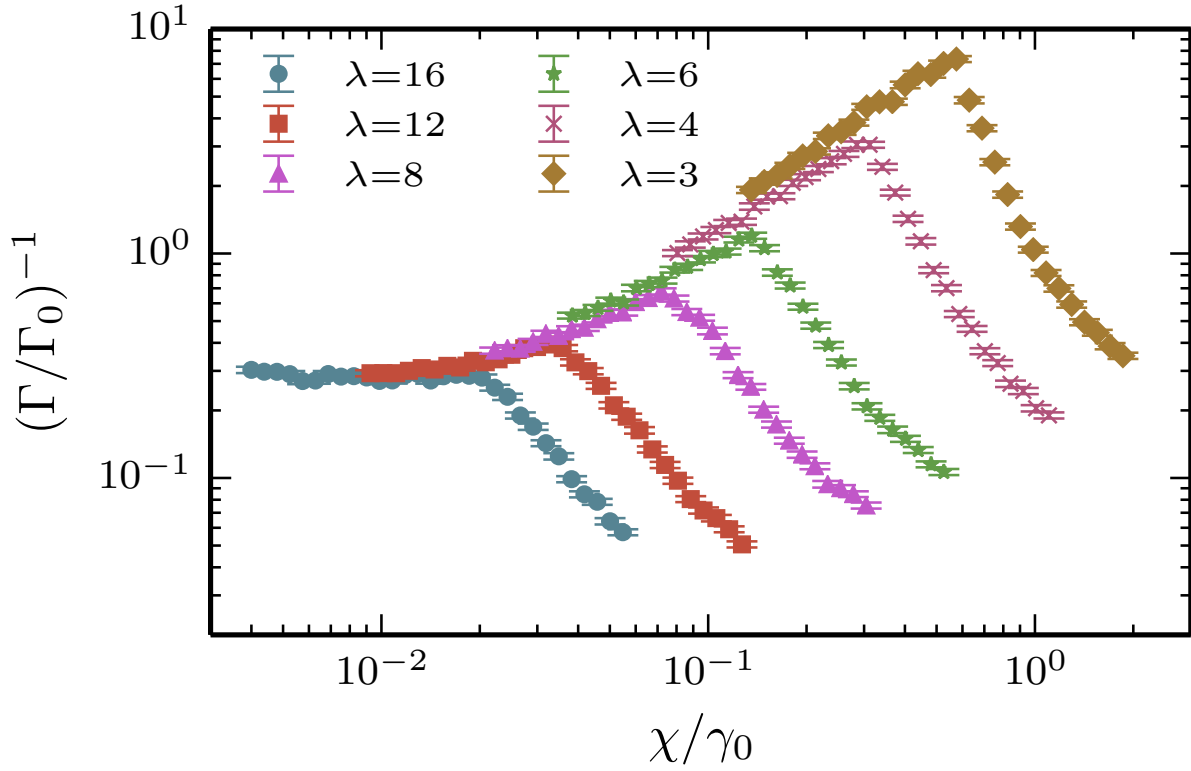


Figure 7.6: Lifetime of the Ising model, Γ^{-1} , expressed in terms of inverse units of Γ_0 (i.e., (7.13)) as a function of the correction rate, χ , rescaled by the translation rate, γ_0 , for different values of λ and for fixed system size $L = 96$, temperature, $T = .07$, and translation rate, $\gamma_0 = .0007$. Protocols were implemented in parallel on $48/\lambda$ blocks (see text). In the absence of the protocol, the lifetime of the Ising model for these parameters corresponds to approximately Γ_0^{-1} , i.e. (7.13). This is the value which all three protocols converge towards in the limit of $\chi/\gamma_0 \ll 1$. Note the decrease in lifetime for $\chi/\gamma_0 \approx 1$, as well as the universal scaling in the lifetime up until the λ -dependent cutoff (i.e., (7.17)), which increases as λ decreases. The only remaining mismatch between the curves is due to the residual λ dependence in the $1/(L - 2 - 2\lambda)$ term of (7.15).

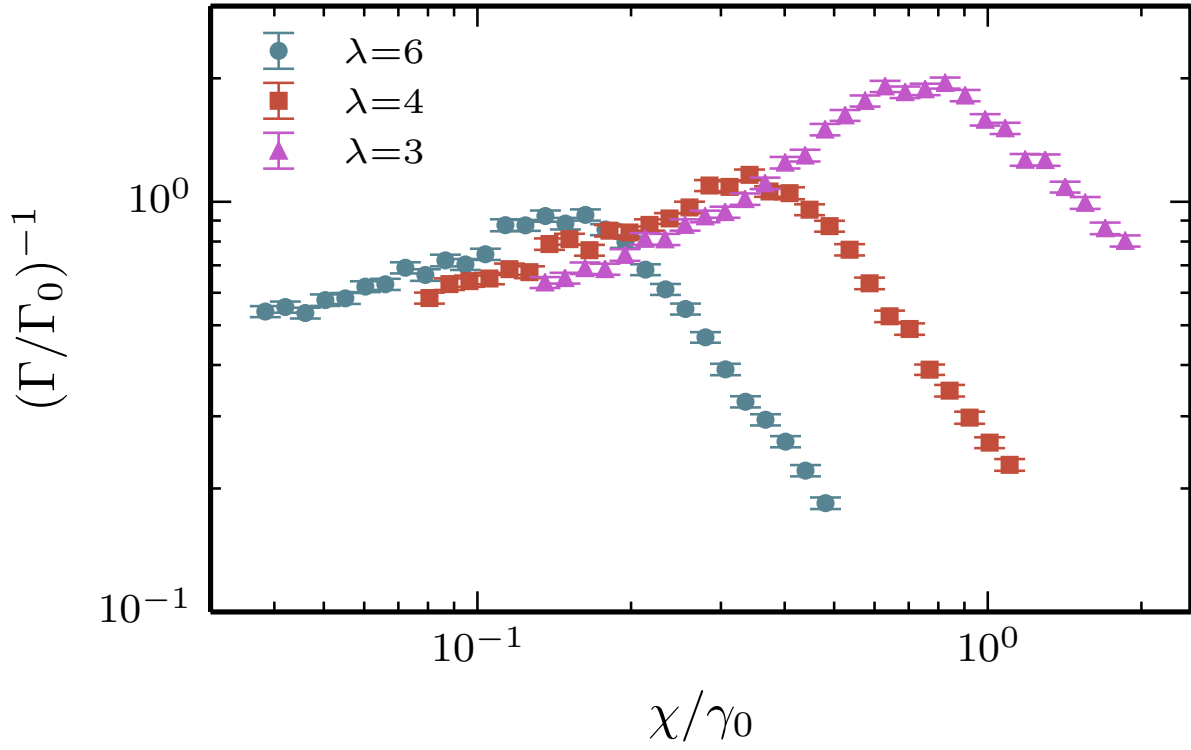


Figure 7.7: Lifetime of the Ising model as a function of the correction rate, χ , rescaled by the translation rate, γ_0 , for different values of λ , and for fixed system size, $L = 96$, temperature, $T = .07$, and translation rate, $\gamma_0 = .0007$. Protocols were implemented serially (see text). The scaling of lifetime with χ is characteristically similar to the parallel case; however, the maximal lifetime is correspondingly smaller for the serial implementation. Note that smaller λ still yields a larger enhanced lifetime.

operators, located on the vertices and plaquettes of the square lattice, respectively. Because of the non-trivial braiding statistics of these two defects, a nontrivial winding resulting from the interaction with the bath gives rise to uncontrolled errors that can be mapped onto logical Z *and* logical X operations, depending on which type of defect incurs the nontrivial winding. These erroneous operations can be suppressed by operating at low temperature and by tuning the relative strength of the plaquette and star terms in the Hamiltonian. Secondly, both of these defects undergo two-dimensional random walks rather than one-dimensional random walks. This difference in dimension gives rise to a modified form of the toric code's finite temperature error rate, due to the differing nontrivial topological random walk probability for two dimensions versus one.

Operationally, these differences only require small modifications of the autonomous pro-

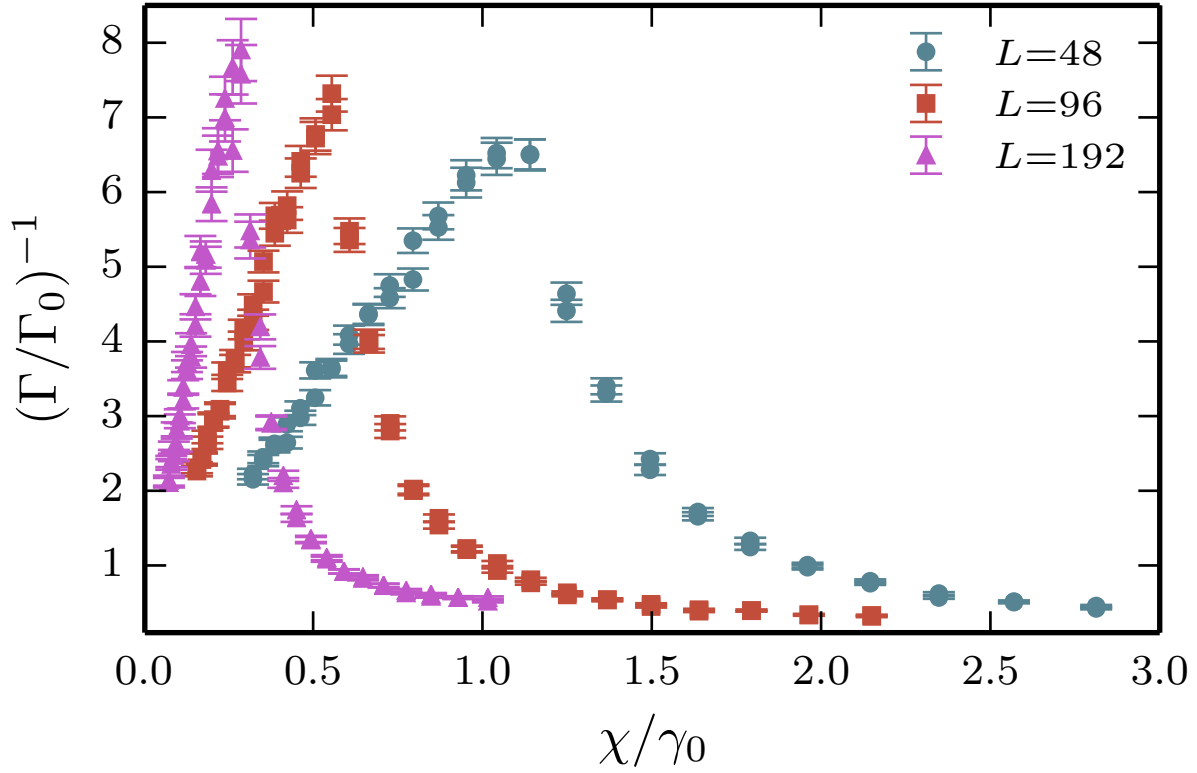


Figure 7.8: Lifetime of the Ising model as a function of the correction rate, χ , rescaled by the translation rate, γ_0 , for different values of system size, L , and for fixed $\lambda = 3$, $T = .07$, $\gamma_0 = .0007$. Protocols were implemented in parallel on $L/2\lambda$ blocks (see text). Note the linear scaling in χ for small values, as well as the shift in the maximum of the lifetime as a function of L .

tol. Namely, there need be two DSWAP operators:

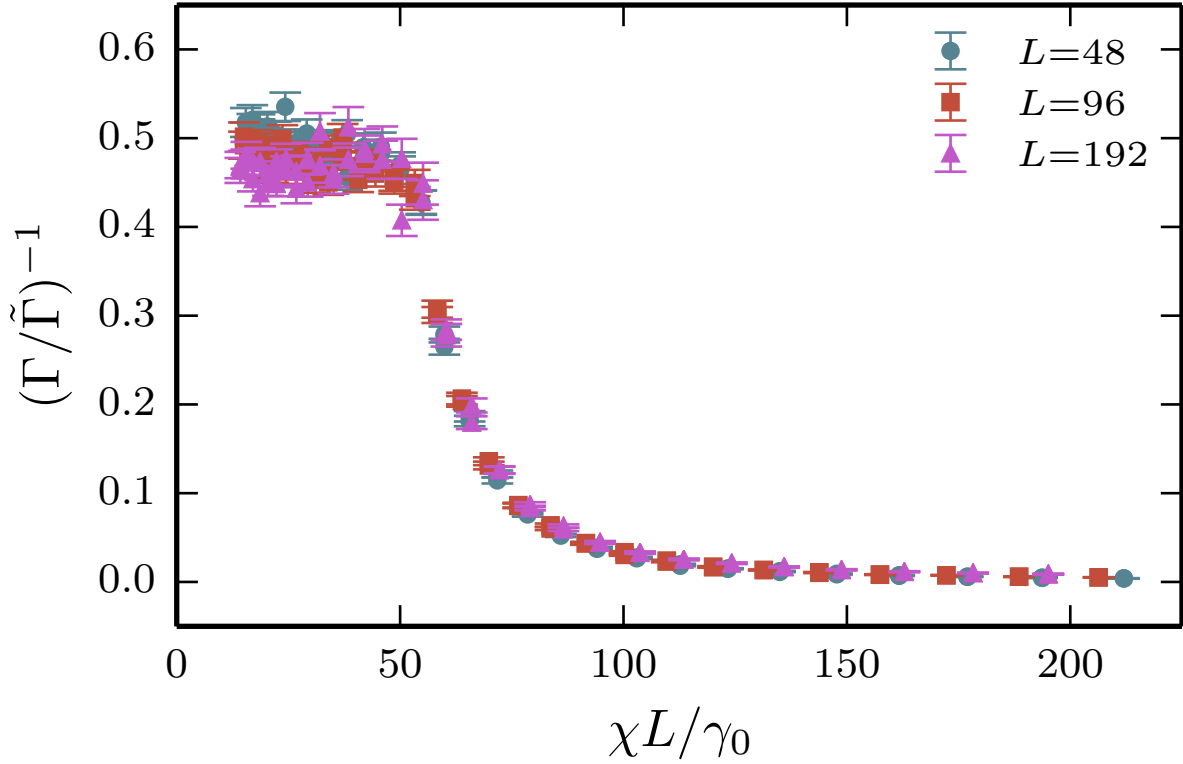


Figure 7.9: This figure contains the same data as Fig. 7.8, but with the χ axis rescaled to χL , and the $1/\Gamma_{\text{Cyc}}$ axis rescaled by (7.14). Hence, the linear scaling in χ , and the slight residual system size dependence have been removed. Note the steep, sudden dropoff in lifetime after $\chi L \sim 55$.

$$\text{DSWAP}_{vv'}^c = \frac{1}{4} \sigma_{vv'}^x (1 - A_v) (1 + A_{v'}) \quad (7.20)$$

$$+ \frac{1}{4} \sigma_{vv'}^x (1 + A_v) (1 - A_{v'}) \quad (7.21)$$

$$+ \frac{1}{4} (1 - A_v) (1 - A_{v'}) \quad (7.22)$$

$$+ \frac{1}{4} (1 + A_v) (1 + A_{v'}) \quad (7.23)$$

$$\text{DSWAP}_{pp'}^m = \frac{1}{4} \sigma_{pp'}^z (1 - B_p) (1 + B_{p'}) \quad (7.24)$$

$$+ \frac{1}{4} \sigma_{pp'}^z (1 + B_p) (1 - B_{p'}) \quad (7.25)$$

$$+ \frac{1}{4} (1 + B_p) (1 + B_{p'}) \quad (7.26)$$

$$+ \frac{1}{4} (1 - B_p) (1 - B_{p'}) \quad (7.27)$$

$$(7.28)$$

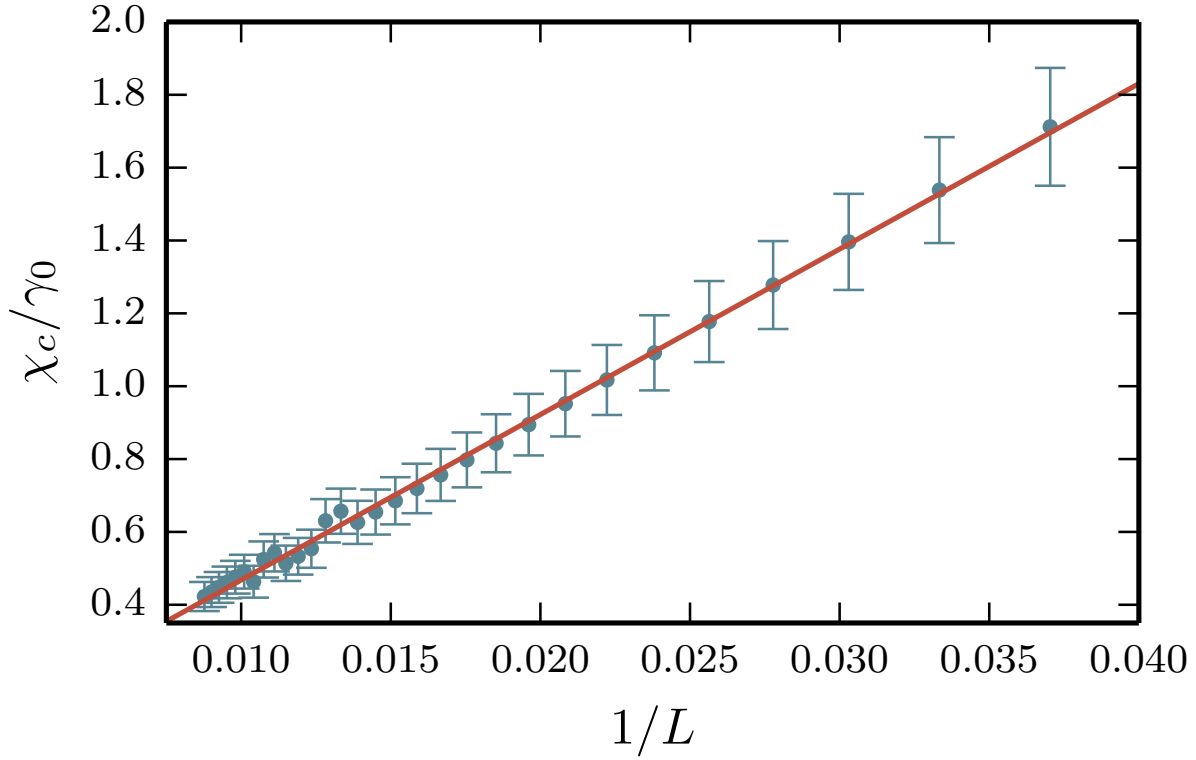


Figure 7.10: The critical cycling rate, χ_c , rescaled by the translation rate, γ_0 , as a function of $\frac{1}{L}$ for $\lambda = 3$, $T = .07$, $\gamma_0 = .0007$. Protocols were implemented in parallel on $L/2\lambda$ blocks. This scaling is consistent with the error model ansatz in (7.17). Fit to $1/L$ in red. Errors are dominated by systematic effects, not sampling error.

These unitary operators translate an A -type (B -type) excitation from a vertex v (plaquette p) to an adjacent vertex v' (plaquette p'). Second, the λ -mixing protocol shuttles defects towards a shared boundary of length λ between subdomains of characteristic area λ^2 .

Because subregions share a boundary of length λ rather than a single site, as in the one-dimensional case, the cycling protocols require at most a factor of λ more swaps to complete a cycle. The protocol then takes the following simple form:

1. Choose a species of quasiparticle
2. Divide the lattice into domains of characteristic area λ^2
3. Pick two λ -domains which share a boundary
4. Pick two candidate defect locations within these two λ -domains.
5. If these defect locations are within the same λ -domain, apply DSWAPs until they would be nearest neighbors. If they are in different λ -domains, apply DSWAPs until they meet at the shared boundary.

6. Repeat (5) until all pairs of defect locations are exhausted.
7. Repeat (3) through (6) until all pairs of λ domains which share a boundary are exhausted.
8. Repeat (1) through (7) until all species of quasiparticle are exhausted.

This protocol is also highly parallelizable, both by operating on multiple pairs of λ -domains, and by acting on simultaneous pairs of defect sites within pairs of λ -domains.

The General Problem

We can always divide a d -dimensional lattice into $N \equiv L^d/\lambda^d$ domains and try to devise an algorithm that fuses defects between adjacent domains. For our protocol, defects are shuttled towards $d - 1$ -dimensional boundaries between adjacent domains of volume λ^d . From this, we can generalize the low temperature dynamics of equation (7.14) to the d -dimensional case as follows:

$$\Gamma_{\text{Cyc}} \propto L^d \gamma_+ \frac{\gamma_0}{\lambda \chi} \frac{\gamma_0}{\gamma_-} P_{\Omega}^d(L, \lambda) \frac{f(\lambda^d)}{L^d}, \quad (7.29)$$

where L is the edge length of the d -dimensional volume enclosed by the system, $P_{\Omega}^d(L, \lambda)$ encodes the probability of a nontrivial topological random walk of a pair of defects in d dimensions with system size L and domain lengthscale λ , and $f(\lambda^d) \propto \lambda^d$ is a protocol dependent function which depends on the implementation details of the algorithm.

Theorem (1) guarantees that an algorithm exists which can perform the cycling in a number of steps polynomial in the dimension of the lattice. However, this Theorem does not guarantee that a λ -mixing protocol exists which solves the problem. In general, for higher dimensions, there are always defect patterns of distance $O(\lambda)$ which are uncorrectable by our λ -mixing protocol. The design strategy is then to try and maximize this minimum uncorrectable distance by careful tiling of the graph of interest.

To be more explicit, if a single pair of defects appears on the graph, uncorrectable errors are generated only when one of the defects escapes to an adjoining region which does not share a boundary with its pair. A cartoon of this process is depicted in figure Fig. 7.11. For one dimension, this cannot happen in one step after an adjacent pair of defects appears. More specifically, if a single pair of adjacent defects appears on the lattice, no single DSWAP will cause such an error to occur, by design, and no single bath operation will cause an adjacent pair of defects to be in nonadjacent regions of size λ . In one dimension, at worst a pair will be created, shuttled around by the protocol, and *then* translate by a bath operator across a boundary, resulting in an uncorrectable error.

This distinction is important because poor choice of tiling in higher dimension *can* result in uncorrectable errors that occur in a single step after pair creation. For example, compare the single hop in the upper half of Fig. 7.12 to the lower half. A defect pair appearing at a corner can transition to an uncorrectable configuration in a single step, whereas in the lower tiling, this is not possible for any initial configuration of adjacent defect pairs. This can be checked by simple enumeration of the possible defect locations and single-hop geometries.

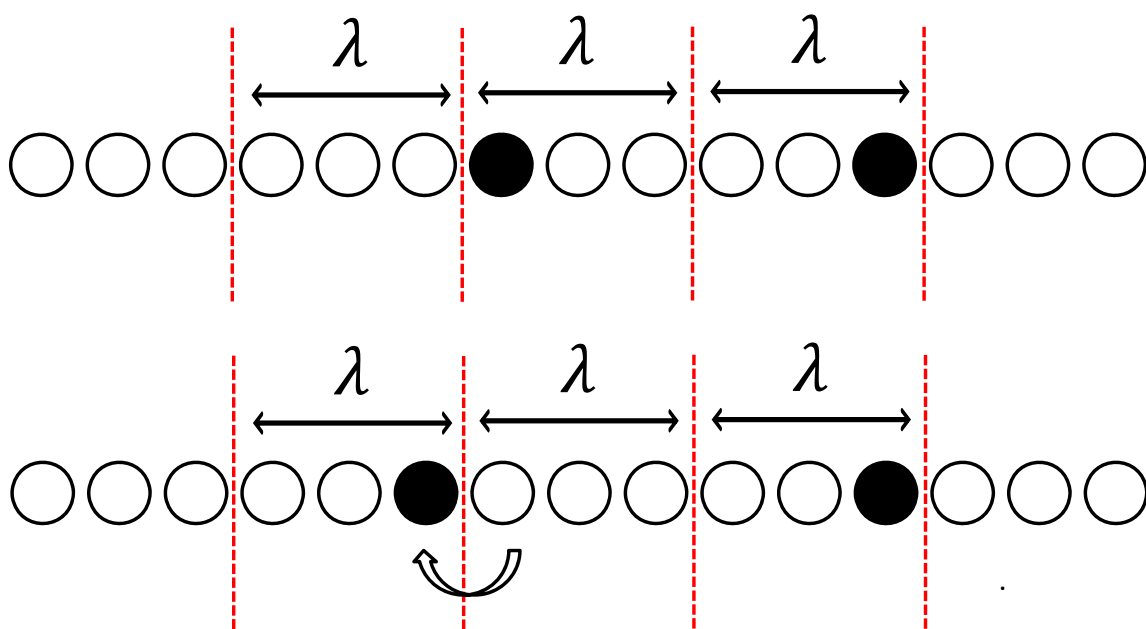


Figure 7.11: A single random application of the bath hopping operator causes a defect to move between adjacent regions of size λ . Once a pair of defects are separated by this distance, the protocol will not be able to correct them with certainty.

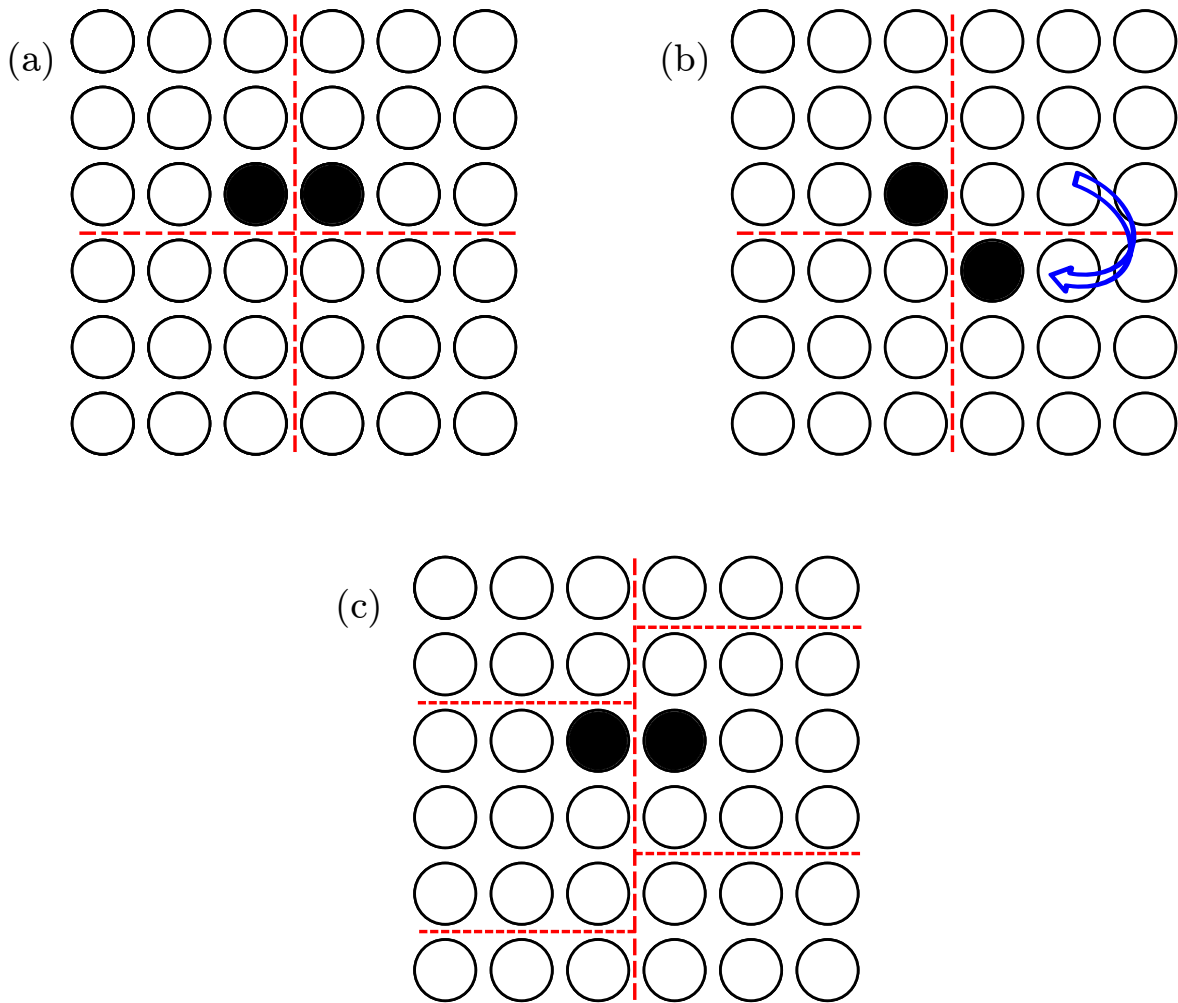


Figure 7.12: A single random application of the bath hopping operator causes a defect to move between nonintersecting regions of size λ , depicted in (a) and (b). (c) depicts a new lattice tiling where no single swap can move defects into two λ regions which do not share a boundary.

This shifted square lattice tiling depicted in the lower half of Figure 7.12 generalizes to three dimensions, and is necessary for equation (7.29) to describe the leading order error process.

Hybrid DSWAP-Stabilizer Codes

While we have demonstrated that our protocol gives rise to an enhanced lifetime for a topological code with string-like error operators, it is also possible and desirable, to use our DSWAP cycling protocol with a more traditional stabilizer detection-correction scheme simultaneously. We postpone numerical analysis of such a scheme for future work, but we sketch such a protocol in this section.

For concreteness, we specialize here to the case of the toric code. Error detection and correction in the toric code requires (1) measurement of all stabilizer syndrome operators and (2) application of a perfect-matching algorithm to determine which pairs of defects to fuse. Whether or not such an algorithm will be successful depends on the density of errors at the time of measurement. Given a stabilizer measurement rate γ , in the “infinite temperature” limit (i.e., $T \rightarrow \infty$ in (8.5)), each edge is acted upon by an error operator with an error probability p . Operationally, the bath is equally likely to create a pair of defects as it is to dissipate a pair. The resulting dynamics are analogous to an uncorrelated “white noise” model.

It is well known that if p is below some critical value, p_c , it is possible to correct the errors in the toric code with certainty. Equivalently, p_c sets the minimum rate at which measurement must occur so that detection is possible in principle. Call this rate γ_c .

For the protocol to have an effect, we must operate in a regime where pair annihilation is favored over pair creation. For simplicity, we work in the low temperature regime where single defect pairs dominate. In this regime, an uncorrectable error has occurred when a single pair of defects becomes separated by more than half the linear lattice dimension. In the presence of the DSWAP cycling protocol, the rate associated with such an event occurring is modified by some constant factor:

$$\frac{1}{\Gamma_{\text{Toric Code Cycling}}} = g \frac{1}{\Gamma_{\text{Toric Code}}} \quad (7.30)$$

where $\Gamma_{\text{Toric Code Cycling}}$ is the error rate of the toric code in the presence of a cycling protocol, and $\Gamma_{\text{Toric Code}}$ is the error rate in the absence of the protocol.

For $g > 1$, i.e., when our protocol actually enhances the lifetime of the code, this effectively reduces the critical detection rate γ_c by the same factor. This is because the protocol effectively reduces the rate at which undetectable pairs are created.

Thus, if a physical realization of a stabilizer error detection/correction cycle is rate limited due to hardware or fundamental noise constraints, the DSWAP cycling protocol provides one avenue towards reducing the critical measurement/detection rate purely by application of local unitaries.

7.6 Discussion

In summary, we have provided a dissipative error correction protocol that enhances the lifetime for models with string-like defects (see Sec. 7.4). In particular, we have combined a local dissipative thermal protocol with carefully designed unitaries, in order to further encourage defect dissipation. Furthermore, we have derived an enhanced lifetime for the one-dimensional Ising model in the presence of our protocol, i.e. equation (7.14), determined the scaling behavior of the optimal protocols, i.e. equation (7.17), and provided numerical evidence for the enhanced lifetime of the Ising model in the presence of our protocol given certain rate assumptions (see Figs. 7.6 through 7.10). Practically, this algorithm increases the lifetime of the system linearly with system size up to a system-size independent cutoff, illustrated in Fig. 7.8, as anticipated from No-Go theorems. Furthermore, we have sketched how this protocol can be generalized to higher dimensional models like the toric code, and used in conjunction with traditional stabilizer error detection/correction schemes (see Sec. 7.5 and 7.5).

The efficacy of these sorts of protocols is intimately related to the scaling of the protocol with system size and protocol parameters, as we have demonstrated. Notably, the best performing versions of our protocol have small λ , and, in a sense, only correct the shortest distance errors. This may seem counterintuitive from the perspective of designing protocols which correct as many errors as possible. For example, suppose we wish to compare a $\lambda = 3$ protocol with total cycle time τ to a $\lambda = 4$ protocol with the same cycle time τ . Note that by fixing total cycle time, we are implicitly requiring that the $\lambda = 4$ protocol be performed more quickly at the level of individual application of DSWAPs (because there are more DSWAPs in a complete $\lambda = 4$ cycle than a $\lambda = 3$ cycle), but we require that the complete error correcting cycle of each protocol is completed in the same amount of time. Naively, we would expect the $\lambda = 4$ protocol to do better, because it is dissipating errors over a longer length scale, but in the same amount of time. Fig. 7.7 indicates a narrow region where this is the case, but, generically, this is not the case.

This can be traced to the poor scaling of the maximal lifetime with λ , as represented by equation (7.17). Essentially, the protocols which correct larger distance errors—i.e., large λ -fixing protocols—employ so many gates that all of the gains of correcting longer distance errors are erased by the time it takes to actually perform the protocol, even when implementing the protocol in parallel.

Additionally, the form of equation (7.14) suggests that larger systems manifestly have lower error rates, because $\Gamma_{\text{Cyc}} \propto \frac{1}{\chi L}$ for low temperature and $L \gg \lambda$. For sufficiently large systems this once again breaks down due to equation (7.17) (see also, Fig. 7.10). Namely, for a fixed cycling rate χ_0 , making the system larger only increases the lifetime so long as $\chi_0 < \chi_c$. Thus, the more favorable scaling of the lifetime with χ at larger system sizes is precisely offset by the poor scaling of χ_c . This can be seen immediately by inserting χ_c from (7.17) into (7.14). This replacement yields the expected scaling of our “best” measurement-free protocol, which scales asymptotically as $\sim \frac{L - \text{constant}}{L}$ —i.e., upper bounded by a constant.

These shortfalls could be circumvented by allowing for longer range unitaries. For example, the DSWAP operator could be replaced by a generalized operator DSWAP_λ which transports domain walls over longer distances. Our insistence on building the protocol entirely out of local DSWAP gates was to perform as honest an analysis as possible with respect to the power of this type of protocol. But if a particular architecture could exchange defects over long distances just as easily as short ones, this would immediately allow for algorithms with better scaling. We hope to examine the optimality of these sorts of protocols in future work.

In the long term, this program is meant to identify the simplest possible set of ingredients necessary to provide protection for a stabilizer code based quantum memory. Many partial ingredients are known, like the No-Go theorems mentioned in Sec. 8.1. Practically, the goal is a protocol designed around the dynamics of the excitations of the stabilizer codes of interest with minimal usage of resources, but which still results in an error threshold so that a state can be preserved indefinitely. With this work, we have demonstrated a constant factor improvement with only local unitaries dressing the system.

7.7 Acknowledgments

This material is based upon work supported by DARPA under Grant No. 3854-UCB-AFOSR-0041 and by the National Science Foundation under Grants No. PIF-0803429 and No. CHE-1213141. CDF was supported by the NSF Graduate Research Fellowship under Grant DGE-1106400.

7.8 Syndrome Decoding for 1D Ising Model

The corrective operator O in Fig. 7.2 can be written as a collection of conditional applications of DSWAP and DWALL, where the applications of the operators are conditioned on the measurements of the stabilizers. We adopt the notation O_{123} to indicate the application of the operator O on qubits 1, 2 and 3. The operator O is given explicitly in Table 1.

7.9 MATCHSEQ and Error Correction

Polynomial Scaling

Define the game *MATCHSEQ* as follows: two nonadjacent vertices on a simply connected graph G are colored black, called defects, the rest white. The player is allowed to perform a conditional swap, or DSWAP, on any two adjacent vertices, which exchanges black vertices and white vertices, and does nothing to pairs of white vertices. If black vertices become adjacent, they immediately *fuse* and become white vertices. Crucially, the player does not know which vertices are colored black.

s_1	s_2	s_3	O
1	1	1	I
1	1	-1	I
1	-1	1	I
1	-1	-1	DWALL_{234}
-1	1	1	I
-1	1	-1	$\text{DSWAP}_{123}\text{DWALL}_{234}$
-1	-1	1	DWALL_{123}
-1	-1	-1	I

Table 7.1: Corrective operations given certain measurements of the stabilizers s_1 through s_3 in Fig. 7.2. If an odd number of domain walls are detected, the identity is applied.

$ M(3) $	1
$ M(4) $	3
$ M(5) $	6
$ M(6) $	10
$ M(7) $	18

Table 7.2: Minimum number of DSWAPs required to necessarily fuse any two defects on a linear chain with open boundary conditions. Computed via breadth first search.

“Winning” *MATCHSEQ* amounts to performing a sequence of moves which guarantees that a pair of arbitrarily placed vertices fuses.

Define the pairing sequence $M(G_v)$ to be the sequence of conditional swaps necessary to bring any configurations of two defects adjacent to one another at least once on a graph G with v vertices. Define the pairing number $|M(G_v)|$ to be the pairing sequence with minimal length. Table 1 tabulates the first few nontrivial pairing numbers for the special case of G equal to a linear chain of length L .

Theorem 1. *The number of DSWAPs necessary to win MATCHSEQ for an arbitrary finite, connected graph G is polynomial in the number of vertices in the graph G .*

Proof: Let $M^*(G)$ be a winning strategy on an arbitrary graph G . Suppose an arbitrary vertex is added to G , called v^* , with up to $|G|$ edges. Call this modified graph G' . Then, performing $M^*(G)$ on G' either fuses two arbitrarily placed defects, or there’s a single defect on the new vertex, and the remaining vertex has just been permuted around in G . A candidate $M(G')$ is then:

1. Perform M^* on G .
2. Pick a vertex, v' , on G . Supposing a defect is on v' , perform a sequence of DSWAPs that

brings that defect adjacent to the new vertex, v^* .

3. Perform the reverse of the sequence of DSWAPs in (2), and repeat (2) with a new v' .
4. Repeat (2) and (3) until all vertices in G are exhausted.

The number of DSWAPs needed for step (2) is at most $|v|$, i.e., the number of vertices in G . Thus, the total complexity of steps 2 through 4 is $O(|v|^2)$. This admits a recurrence relation:

$$|M(G_{v+1})| \leq |M(G_v)| + b * |v|^2, \quad (7.31)$$

Where b is a constant ≤ 1 . $b = 1$ corresponds to the case that the new vertex is only connected to one vertex in the original graph G . Solving this recurrence relation in the limit that the inequality is always saturated yields $|M(G_v)| \leq O(|v|^3)$. It is worth emphasizing that this is not the minimal such solution to MATCHSEQ, just one that is easily provably polynomial in $|G|$. The likely graph structures of interest to an experimentalist, i.e., linear chains, square lattices, admit more favorable algorithms with softer polynomial scaling.

Strategies

For a given winning strategy, $M^*(G)$, it will be convenient to classify the strategy based on the maximum distance that any given defect is moved. In the sequel, we will construct $M^*(G)$ out of a concatenation of $M^*(G_i)$, where G_i are subgraphs of G . Thus, if only a single defect happens to be in the subgraph G_i , we would like to bound the maximum displacement of that defect by the strategy.

Let $d(M)$ be the maximum distance any given defect is moved by a given strategy. For winning strategies, $d(M)$ is at least half of the maximum distance between defects and at most permutes defects around the entire graph, so $|v| \geq d(M) \geq |v|/2$. Define a strategy $M^*(G)$ to be k -mixing if $d(M) = k$.

We introduce this terminology because most physical realizations of MATCHSEQ will have a background rate of uncontrollable DSWAPs, driven by coupling to a bath. k -mixing strategies are necessary in such cases to be partially resilient to these random “error” DSWAPs.

Mapping onto 1D Ising Model

Vertices in the problem setup for MATCHSEQ correspond to the dual lattice of the Ising chain, and the process of fusion is simply dissipation of adjacent domain wall pairs by the bath.

However, we caution that the mapping onto MATCHSEQ is only partial: defects on the Ising chain hop in the absence of any experimental intervention, so the Ising chain is more akin to a game of MATCHSEQ with a random, background DSWAP rate. Further, there can be more than two pair of excitations on the Ising chain, but for low temperature, the regime

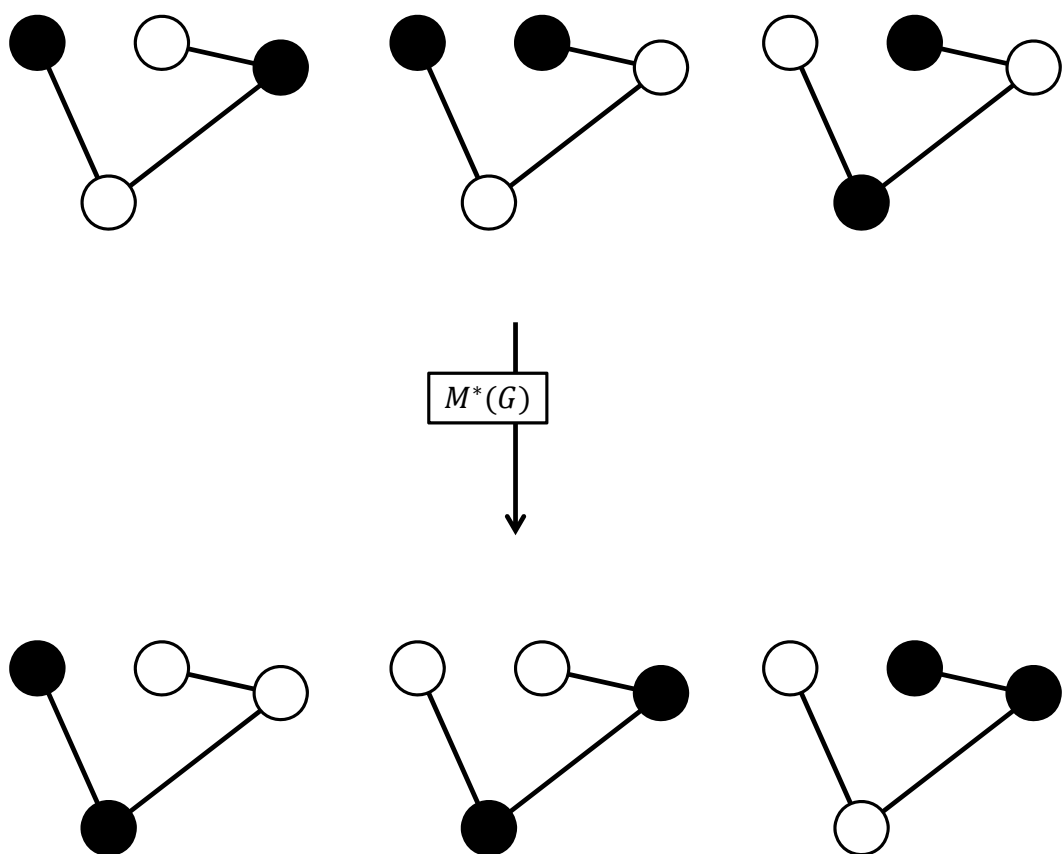


Figure 7.13: An example of the result of applying a winning sequence $M(G)$ to a graph with defect pairs present. No matter where the defects are, the sequence of DSWAPs brings pairs adjacent, whereupon they immediately fuse.

where the protocol works best, this is exceedingly rare. Lastly, defect pairs don't necessarily fuse immediately—fusion happens at the timescale set by the system-bath coupling, the type of bath model, and the temperature, so rate at which DSWAPs are applied must be chosen carefully for optimal lifetime enhancement.

7.10 Algorithm for 1-D Ising Model

Here we provide python code for a λ -mixing algorithm for the Ising chain. The output of the algorithm is a sequence of locations. Our convention is such that location i indicates a DSWAP should be applied that exchanges defects between sites i and $i + 1$. Heuristically, the algorithm attempts to shuffle defects towards the shared boundary of the disjoint sites $0, 1, 2, \dots, \lambda - 1$ and $\lambda, \lambda + 1, \dots, 2\lambda - 1$. That is, it attempts to translate defects so that they are adjacent to each other at sites $\lambda - 1$ and λ . After completing this cycle, the algorithm repeats for the next two adjacent domains, $\lambda, \dots, 2\lambda - 1$ and $2\lambda, \dots, 3\lambda - 1$. This continues until the lattice has been exhausted.

The following python code generates a complete sequence of DSWAPs given a lattice size and λ length.

```
def SwapProtocol(L, lamb):
    prot = []
    numofdomains = L / lamb
    for d in xrange(numofdomains):
        for k in xrange(lamb):
            for m in xrange(k):
                prot.append((lamb-1-k+m+d*lamb)%L)
            for i in xrange(lamb):
                for j in xrange(i):
                    prot.append((lamb+i-j-1+d*lamb)%L)
    return prot
```

The protocol is parallelized by operating simultaneously on specific pairs of domains. To be more explicit: denote the first λ sites as λ_1 , the next λ sites λ_2 and so on. The algorithm can be naturally partitioned into a sequence of DSWAPs that translates defects to the shared boundary of λ_1 and λ_2 (call this sequence (λ_1, λ_2)), followed by a sequence that translates defects to the shared boundary between λ_2 and λ_3 , (call this sequence (λ_2, λ_3)), etc. To parallelize, apply the sequence (λ_1, λ_2) simultaneously with (λ_3, λ_4) , (λ_5, λ_6) , etc. When complete, apply the sequence (λ_2, λ_3) with (λ_4, λ_5) etc. This exhausts the protocol.

Chapter 8

Autonomous algorithms with measurements

The representation described here, although conceptually simple, is not efficient and would not have condemned the guiltless.

King James Programming Tumblr

After developing algorithms that did not use measurements and finding that those algorithms could not restore the exponential scaling of the toric code lifetime with size, we turned to algorithms that used asymptotically as few measurements as possible. In this chapter, we demonstrate the existence of a finite temperature threshold for a 1D stabilizer code under an error correcting protocol that requires only a fraction of the syndrome measurements. Below the threshold temperature, encoded states have exponentially long lifetimes, as demonstrated by numerical and analytical arguments. We sketch how this algorithm generalizes to higher dimensional stabilizer codes with string-like excitations, like the toric code. The text of this chapter is the recent manuscript Freeman et al. [56].

8.1 Introduction

Quantum memories are an essential component for many quantum technologies, including quantum computing and quantum repeaters. In analogy to modern classical memories, one ideally wants a stable quantum memory that requires little or no active intervention and error correction. Unfortunately, no physical system that passively preserves quantum information indefinitely at finite temperatures and in an experimentally accessible number of dimensions is known [118]. Instead, the operation of all known practical quantum memories require a combination of passive elements (i.e. dissipative cooling) and active measurement and correction cycles to keep quantum information protected. In this work, we study the degree to which the amount of active measurement and correction can be reduced while maintaining quantum memory stability (our notion of stability, to be quantified later, corresponds to

exponentially long lifetime for encoded states below some finite threshold temperature). We develop a new decoding and correction protocol that enables one to trim the number of measurements to a fraction of the complete set of measurements normally considered, and still maintain quantum memory stability.

We restrict our attention to quantum memories defined through stabilizer codes. For near term architectures, stabilizer codes[59] have emerged as the leading candidate for encoding quantum information and subsequent active error correction in quantum hardware, with small scale architectures actively being developed and deployed[78, 115, 31, 32]. A tremendous amount of effort has gone into developing novel decoding and correction schemes for stabilizer codes, particularly the toric code. Different schemes often emphasize different decoding features, like efficient decoding[80, 125, 36], locality[20, 119, 62, 66], robustness to particular sorts of noise[100, 10, 134], or use of dissipation[54, 69, 68, 57, 106, 105, 29, 30, 61, 65, 137, 128, 34, 73, 6].

In previous work [55], we analyzed the finite temperature dynamics of the toric code, verifying the well-known no-go theorems for the upper bound to the lifetime of the toric code at finite temperature[18, 3, 30, 136, 63, 82, 116, 117]. Using this analysis, we were able to construct a measurement-free protocol for protecting the encoded qubits of the toric code [54], but these protocols again were limited by the no-go theorems, and only provided a multiplicative constant increase to the lifetime.

Building off this previous work, here we examine the extent to which a limited amount of measurement can increase the lifetime of stabilizer codes with string-like excitations. In sum, we demonstrate an algorithm that, for any constant density of measurements for a stabilizer code with stringlike excitations undergoing dissipation at a fixed temperature, exhibits a threshold temperature, below which exponentially long lifetimes can be achieved in the encoded space. The threshold temperature scales with the amount of measurement used—fewer measurements result in a smaller threshold temperature, whereas more complete measurement raises the threshold temperature. This tradeoff is commensurate with and complements what is known about decoding the stabilizer codes in the presence of *noisy*, but complete measurements[97].

The remainder of the manuscript is structured as follows: Section 8.2 briefly reviews the theoretical tools used for performing simulation of stabilizer codes at finite temperature. The content of this section is also expanded upon in refs [55, 54]. Section 8.3 includes the full description of our limited measurement algorithm, including a discussion of the expected low temperature error processes that cause the algorithm to fail, and a heuristic justification for the expectation of a threshold temperature below which a stable quantum memory is feasible. Section 8.4 details our numerical investigations of our algorithm for the 1D Ising model. Finally, Sec. 8.5 sketches how this algorithm could be generalized to higher dimensions, and Sec. 8.6 provides some concluding analysis and discussion.

Repetition Stabilizer Code	1d Ising Stabilizer Hamiltonian
Encoded States	Ground states
Bit flip errors	Excited states
Decoding and error correction	Identical or by cooling

Table 8.1: A short summary of the similarities and differences between the Ising model considered as a code (left panel) versus as a hamiltonian (right panel).

8.2 Stabilizer codes at Finite Temperature

Definitions

In this section, we briefly review the theory of the 1D Ising model, as well as the Markovian open quantum systems formalism for evaluating its finite temperature dynamics. The Hamiltonian for the 1D Ising model is

$$H_{\text{Ising}} = -\Delta \sum_i \sigma_z^i \sigma_z^{i+1} \quad (8.1)$$

where, for the remainder of the manuscript, unless explicitly stated otherwise, we assume $\Delta = 1$. This is exactly the Hamiltonian version of the repetition stabilizer *code*[54]. Note that the terms $\sigma_z^i \sigma_z^{i+1}$ correspond exactly to the parity check stabilizer operators of the repetition code (see Table 8.1).

In the parlance of the 1D Ising model, bit flip errors are often also classified via the dual variables called *domain walls* or defects. Defects are simply locations on the 1D Ising chain where a stabilizer operator yields a measurement of -1 —i.e., locations where neighboring spins point in different directions. With periodic boundary, the number of these locations is always even, and a single bit flip event either creates a pair of such defects, deletes a pair of defects, or causes a defect to translate by one unit.

As long as less than half the system has had errors, a majority rule decoder that has access to measurements of the full set of stabilizers $\sigma_z^i \sigma_z^{i+1}$ will reliably be able to correctly identify and remove errors. When errors are completely independent (i.e., at very high temperature), we can define random variables $x_i = 1$ when an error occurs on site i , and 0 otherwise. If these errors occur with probability p on each spin, independently at random every error detection cycle, then Chernoff's bound gives an upper bound to the probability of an error in the encoded space, $P(\sum_i x_i \geq L/2) \leq \exp[-Lp \frac{\delta^2}{2+\delta}]$ for $\delta = 1/2p - 1$. Thus, for complete measurement, errors in the encoded subspace are exponentially suppressed in system size, so long as the error rate is sufficiently small.

For much of the remainder of the manuscript, we consider how the decoding scheme changes when one does *not* have access to the full set of stabilizer measurements.

Following Ref. [54], we consider a simple local Ohmic, Markovian bath to model finite temperature effects. This is modeled by the following master equation in Lindblad form:

$$\dot{\rho} = \sum_i 2c_i \rho c_i^\dagger - c_i^\dagger c_i \rho - \rho c_i^\dagger c_i, \quad (8.2)$$

Here ρ is the density matrix, with Lindblad operators c_i chosen to take the form:

$$\{c_i(\Delta)\} = \left\{ \sqrt{\gamma(0)}T_i, \sqrt{\gamma(\Delta)}D_i^\dagger, \sqrt{\gamma(-\Delta)}D_i \right\} \quad (8.3)$$

where T_b translates a defect by one unit, D_b^\dagger creates a pair of defects, D_b dissipates a pair of defects, and $\gamma(\cdot)$ is a rate function dependent on the details of the bath. This bath is chosen to model the dynamics of local, single bit-flip errors. In the Pauli basis, these operators take the following form.

$$\begin{aligned} D_i^\dagger &= \frac{1}{4} (I\sigma_x I) (1 + I\sigma_z \sigma_z) (1 + \sigma_z \sigma_z I) \\ D_i &= \frac{1}{4} (I\sigma_x I) (1 - I\sigma_z \sigma_z) (1 - \sigma_z \sigma_z I) \\ T_i &= \frac{1}{4} (I\sigma_x I) (1 - I\sigma_z \sigma_z) (1 + \sigma_z \sigma_z I), \end{aligned} \quad (8.4)$$

By convention, we define i to index the first qubit in these operators.

Finally, the remaining details of the bath are specified by the spectral density, which determines the rates with which the different Lindblad operators act:

$$\gamma(\omega) = \xi \left| \frac{\omega^n}{1 - e^{-\beta\omega}} \right| \quad (8.5)$$

where $n = 1$ corresponds to an Ohmic spectral density, which is the choice we make for the remainder of the manuscript. With this choice, in the absence of any error correcting protocol, it can be shown that the 1D Ising model has a system size independent thermal logical error rate given by[54]

$$\Gamma_0 = \frac{\gamma(0)}{1 + e^{1/T}} \quad (8.6)$$

We define the bare lifetime of qubits evolving under the 1D Ising model Hamiltonian in contact with an Ohmic thermal bath to be Γ_0^{-1} .

Finite Temperature vs. Infinite Temperature

The majority of the error correction literature assumes an error model akin to an “infinite temperature limit”. More precisely, an array of physical qubits receives errors from some

set of error operators E_i independently at random with some probability p during every error correction cycle. The threshold theorems state that there exists some critical error probability p_c below which it is possible to return an error correcting code to its encoded state with unit probability for asymptotically large systems; e.g., for the toric code, $p_c \approx .109$ [125].

In contrast, thresholds at finite temperature are usually quoted in terms of a critical *temperature*. That is, there must exist some critical temperature T_c below which codes can be reliably corrected. Unfortunately, this definition obscures a great deal of physics—different choices of bath model can greatly affect the dynamics of the error processes, to the extent that a quoted “critical temperature” often implicitly specifies a choice of bath model. Because different bath interactions can give rise to different system dynamics, the choice of bath also directly affects the strategy used for error correction. For example, it is known that the toric code’s threshold temperature is altered by considering a space-correlated bath rather than an uncorrelated one [100].

The main consequence of choosing an Ohmic bath is that it sets the amplitude of the excitation hopping process. That is, $\gamma(0)$ is determined by the $\omega \rightarrow 0$ limit of the spectral density of the bath, and for the Ohmic bath taking the $\omega \rightarrow 0$ limit of Eq. 8.5 yields $\gamma(0) \sim T$. Ultimately, this means that the hopping rate of domain walls is controlled by this choice of bath model. At finite temperatures, this introduces correlations into the patterns of errors that effect the system, and so it is no longer possible to talk about an “independent error probability per site”. In contrast to the behavior of $\gamma(0)$, the other operationally important feature of the bath, the ratio of defect creation and annihilation rates, is set by detailed balance to Boltzmann-like scaling (i.e., $\gamma(\Delta)/\gamma(-\Delta) = \exp(-\Delta/T)$), and is independent of the choice of bath spectrum.

In the most extreme case, at sufficiently low temperature, pairs of neighboring defects are most often immediately dissipated by the bath upon creation via a D_b operator. However, if a pair creation is followed by a pair hopping event – i.e., a D_b^\dagger followed by a T_b – the error can no longer be immediately dissipated by the local action of the bath. Subsequently the defects will undergo a one-dimensional random walk, and topologically non-trivial random walks will cause uncorrectable logical errors.

Thus, error correcting the 1D Ising model at low temperature with this sort of bath dynamics reduces to attempting to identify these randomly-migrating rare pairs of defects. While a majority-rule decoding scheme works in both low and high temperature limits for the Ising model, if the number of measurement resources is restricted, the standard majority rule scheme breaks down because of the intrinsic uncertainty regarding unmeasured defects.

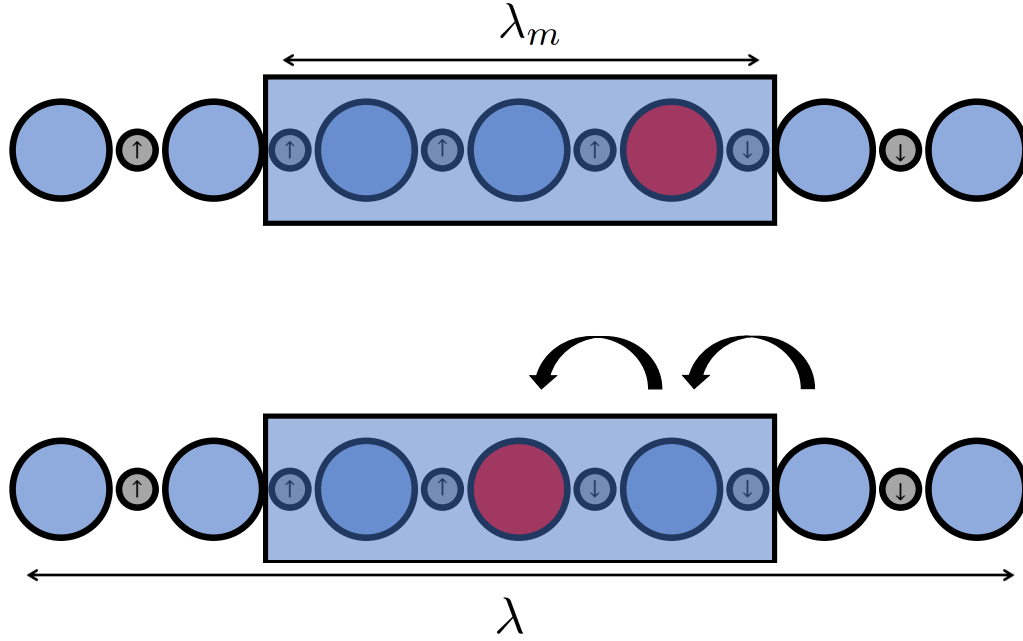


Figure 8.1: This cartoon illustrates the “centering” procedure for detected defects on a unit cell. Spin variables are in gray, and domain wall variables are in blue (no defect present) and red (defect present). When a defect is detected on a measurement patch (blue box), it is swapped to the center of the measurement patch via the DSWAP operator (black arrows). The defect immediately adjacent to it is also swapped onto the measurement patch so as not to pull apart defect pairs that would have otherwise dissipated. Measurement patch lengthscale λ_m indicated by arrow on top, and unit cell lengthscale λ indicated by arrow on bottom. The measurement fraction is defined as $m \equiv \lambda_m/\lambda$.

8.3 Few Measurement Error Correction Algorithm

The Algorithm

In this section, we sketch a new algorithm which reliably removes errors in the 1D Ising model below a threshold temperature, which we determine numerically. The primary technical innovation of this algorithm, and its generalization to quantum memories based on any stabilizer Hamiltonian, is that it does not require measurement of the complete set of stabilizer operators for a given stabilizer code—only a fixed subset. We assume *i*) that the system is subject to periodic measurements on periodically spaced measurement “patches”, *ii*) that measurement readout and processing occurs much faster than any system timescale, and *iii*) that the system is subject to a thermal bath as described in Sec. 8.2.

The algorithm can be summarized in five steps:

1. Measure stabilizers on patches, keeping record of the age of defects that are already on patches—i.e., the amount of time a defect is continuously detected on a patch—as well as defect locations.

2. Perform “centering” on patches with defects (see Fig. 8.1), based on centering protocol introduced in Ref. [54].

3. Calculate probability of fusion (explicitly given in Eq. 8.18) for all pairs of measured defects residing on the measurement sites. This probability serves as an estimate for whether two defects should be paired or not for the purposes of error correction.

4. Probabilistically perform error correction based on probabilities calculated in step 3.

5. Repeat steps 1 – 4.

Step 2 encourages defects to remain localized at measurement patches. This centering protocol can be performed entirely unitarily by the DSWAP operator, which takes the following form in the Pauli basis,

$$\text{DSWAP}_i = \frac{1}{2} (III + I\sigma_x I + \sigma_z I\sigma_z - \sigma_z \sigma_x \sigma_z) \quad (8.7)$$

where i indexes the location of the first qubit being acted upon by the operator by convention.

If a domain wall exists either between the first and second qubit or the second and third qubit, then the DSWAP operator exchanges those domain walls. If there are no domain walls, it acts as the identity. By concatenating a sequence of DSWAPs, i.e. $\text{DSWAP}_i \text{DSWAP}_{i+1} \text{DSWAP}_{i+2} \dots$, domain walls can be shuttled to the center of the measurement patch for efficient tracking.

The centering process, illustrated in Fig. 8.1, aids the probability of fusion calculation by ensuring that the coordinates and measurement times are representative of when and where defects are actually created. If defects escape from measurement patches, then upon being measured again, the time recorded by the measurement patch now underestimates how old the defect actually is, biasing the probability estimate. This centering operation greatly reduces the probability of defect escape. Any remaining underestimate of defect lifetimes can be fixed by a more elaborate record keeping protocol (See Appendix 8.9).

Note that the pattern of DSWAPs used in Fig. 8.1 also swaps the neighboring, *unmeasured* defect onto the measurement patch. This is to ensure that the protocol does not inadvertently create a new separated pair of defects in the system by shifting only one defect in a potentially adjacent pair.

Fusion Probability Calculation

To perform error correction properly, we need to be able to estimate the probability that two given measured defects are a pair, given that they have been measured at two particular measurement patches at two different times. For notational convenience, we define:

$$d_1 : d_2 \equiv \text{defect } d_1 \text{ and } d_2 \text{ are a pair} \quad (8.8)$$

and

$$d_i^{x_1, t_1} \equiv \text{defect } d_i \text{ measured at time } t_1 \text{ at patch } x_1 \quad (8.9)$$

Then, we aim to calculate the fusion probability:

$$P(d_1 : d_2 | d_1^{x_1, t_1} \wedge d_2^{x_2, t_2}), \quad (8.10)$$

i.e., the probability that two defects measured at spacetime coordinates (x_1, t_1) and (x_2, t_2) are part of the same defect pair, and therefore should be fused in a correction step.

To calculate this probability, we proceed via Bayes rule:

$$\begin{aligned} P(d_1 : d_2 | d_1^{x_1, t_1} \wedge d_2^{x_2, t_2}) = \\ \frac{P(d_1^{x_1, t_1} \wedge d_2^{x_2, t_2} | d_1 : d_2) P(d_1 : d_2)}{P(d_1^{x_1, t_1} \wedge d_2^{x_2, t_2})} \end{aligned} \quad (8.11)$$

The individual terms on the right hand side of equation 8.11 are straightforward to interpret. $d_i^{x_i, t_i}$ indicates a defect residing on a measurement patch centered on spacetime coordinate x_i, t_i . $P(d_1^{x_1, t_1} \wedge d_2^{x_2, t_2} | d_1 : d_2)$ represents the probability that two measured defects would be at (x_1, t_1) and (x_2, t_2) given that they are indeed a pair. $P(d_1 : d_2)$ represents the probability that two measured defects, d_1 and d_2 , are in fact a pair. Finally, $P(d_1^{x_1, t_1} \wedge d_2^{x_2, t_2})$ is the probability that two defects are measured, one at (x_1, t_1) , and the other at (x_2, t_2) .

$P(d_1^{x_1, t_1} \wedge d_2^{x_2, t_2} | d_1 : d_2)$ can be related to the probability that a one dimensional diffusion process with diffusion constant D will perform an excursion with a displacement $|x_2 - x_1|$ or greater in a time $t_2 - t_1$, i.e., will perform an excursion that can reach measurement patches at x_1 and x_2 . Explicitly,

$$\begin{aligned} P(d_1^{x_1, t_1} \wedge d_2^{x_2, t_2} | d_1 : d_2) \\ = 1 - 2 \int_0^{|x_2 - x_1|} dx \frac{1}{2\pi D |t_2 - t_1|} \exp\left(-\frac{|x_2 - x_1|^2}{2D |t_2 - t_1|}\right) \\ = 1 - \text{erf}\left(\frac{|x_2 - x_1|}{2\sqrt{D |t_2 - t_1|}}\right) \end{aligned} \quad (8.12)$$

For our analysis, we will choose $D \propto \gamma_0$. The exact correspondence between D and γ_0 depends on the details of the error correction algorithm itself, so, in practice, we treat the constant of proportionality as an empirically tuned parameter. Furthermore, we approximate any detected defects as arising from a pair that was created an equal distance between the measurement patches at locations x_1 and x_2 for the purposes of calculating the probability in Eq. 8.12.

As we discuss in Appendix 8.8, the remaining two factors are not as important for the decoding scheme as the likelihood term in Eq. 8.12. In practice, we find that using the expression from Eq. 8.12 alone is sufficient to provide resilient error correction. We defer further discussion to the appendix.

Error dynamics

In this section, we discuss parameter regime in which we expect the error correcting algorithm to perform well. We then derive the logical error rate for a simple error model, under some simplifying assumptions about the error dynamics. While this error model does not account for the complete error dynamics of the full 1D Ising model in the presence of our protocol, we argue how it nonetheless serves as a worst-case approximation to the true error dynamics. Finally, we describe how this protocol provides a threshold for any finite density of measurements.

Correspondence between model and full error dynamics

In this section, we detail the approximations and rate assumptions that are necessary for the error correcting algorithm to perform well. The primary approximations made are concerning (1) fast defect detection, (2) accurate pairing, (3) defects escaping measurement patches, and (4) defect interactions.

(1). If defects are produced in between measurement patches faster than they are detected, then this algorithm cannot in principle correct errors. Thus, we require that the characteristic diffusion time for defects in the bulk to migrate to a measurement patch, $\gamma_0^{-1}(\lambda_b)^2$, where $\lambda_b = \lambda - \lambda_m$, to be much shorter than the characteristic timescale over which a pair of defects is created in the bulk, $\gamma_0^{-1} \exp \Delta/T$. Thus, working at low temperature ensures the validity of this approximation.

(2). If defects are paired incorrectly more often than they are paired correctly, then the algorithm will fail. Let τ_e^{-1} be the rate of the error process and τ_d^{-1} be the rate of a non-erroneous error correction operation. To ensure that $(\tau_e^{-1}/\tau_d^{-1}) \leq 1$ (see Eq. 8.14), we must work in the diffuse limit, where the average number of defects per unit cell is much less than 1. This is equivalent to $\lambda\gamma_+ \ll 1$. This ensures that, when defects are being processed by the algorithm, that more often than not, defects will be correctly paired simply because it's unlikely there are any other defects nearby. Thus, assuming condition (1)—that defects are detected quickly—defect pairs satisfying Eq. 8.12 are more likely than not to be genuine pairs.

(3). While the simple model does not account for defects escaping measurement patches, this can occur in the real system when a series of translation events occurs between measurements. For a measurement rate χ , these processes are of $O(\gamma_+(\gamma_0/\gamma_-)(\gamma_0/\chi)^{\lambda_m/2})$, for measurement patches of size λ_m , assuming $\chi > \gamma_0$. Thus, this process can be suppressed by working with a larger sized measurement patch, or with a measurement rate suitably larger than the intrinsic translation rate of the system, γ_0 . At worst, the age of defects that escape measurement patches but that are subsequently recaptured may be underestimated by the algorithm, because the “age” of the defect would be erroneously reset to zero. This would then erroneously underestimate the distance the algorithm would plausibly search for a pairing defect—i.e., the denominator of the error function in Eq. 8.12 could be artificially small because the “real” defect age is actually older. While these sorts of errors can poten-

tially spoil the error correcting protocol at very long distances—much larger than considered in this manuscript—these errors can be corrected with a modified version of our algorithm without any additional measurement resources, detailed in Appendix 8.9.

(4). In reality, defects can annihilate without the protocol intentionally pairing them. To leading order, at low temperature, these processes are “self-correcting”. That is, a pair of neighboring defects enters the system, and then is subsequently annihilated. In principle, it is possible for a sequence of k free pairs of defects to appear in the bulk—one pair per unit cell—of which, $k - 1$ are then subsequently erroneously “corrected”, resulting in two defects separated by a distance $k\lambda$, but this process is exponentially slow in the average defect unit cell density, which we already choose to be small via condition (2). That is, the error rate due to the erroneous separation of defects by a distance λk is $\propto (\lambda\gamma_+)^k$.

A simple error model

To bound the error rate of the Ising model in the presence of our protocol, we study a simple error model for “spurious error correction” events. A representative example of one of these events is when two pairs of defects are detected in the system (four defects total on four distinct measurement patches), and the protocol erroneously pairs one defect from each distinct pair. Because the density of defects is low at low temperature, this error process is similar to an error process that occasionally randomly translates one defect of a pair some distance. The distance one of the pair becomes separated depends on the age of the defect, as well as whether the erroneously paired defect was to the left or the right of the original pair of defects.

Thus, the simplified error model is defined as follows: suppose that a single pair of defects is in the system, and that no new pairs will be introduced. One of the pair is fixed on a measurement patch, and the other is, at time $t = 0$, undetected and residing somewhere in the bulk between measurement patches. We will model spurious error correction by an error process that translates the unmeasured defect by a distance $2\sqrt{\gamma_0\delta t}$. As time increases, the characteristic distance over which this error process can occur also increases, in accordance with the typical pair-wise separation between two defects performing a random walk. This typical distance is exactly the factor used by the error correction algorithm to determine if a pair of defects should be corrected or not.

An uncorrectable error will occur if the bulk defect remains undetected up until it crosses half the system. For unit cells of size λ , the probability that this occurs is roughly $(\tau_\epsilon^{-1}/\tau_d^{-1})^k$, where k is the number of times the error process must occur for the error process to have separated the defects a distance $(L/2\lambda)$, τ_ϵ^{-1} is the rate of the error process and τ_d^{-1} is the rate of a non-erroneous error correction operation. After a timescale $q \cdot \tau_d$, the defect pairs will have been separated a distance equal to

$$\sqrt{q \cdot \gamma_0 \tau_d} \tag{8.13}$$

assuming that they are never correctly paired. This grows as $q^{1/2}$, thus, k scales approximately as $(L/\lambda)^2$. Finally, assuming there are L/λ such simultaneous independent error

processes in the system—one for each measurement patch—then the total error probability scales as

$$P(\text{error}) \leq (L/\lambda)(\tau_\epsilon^{-1}/\tau_d^{-1})^{(L/\lambda)^2}. \quad (8.14)$$

For sufficiently low temperatures (see Sec. 8.3), τ_ϵ^{-1} is much smaller than τ_d^{-1} , thus the full probability of erroneous corrective operations is exponentially small in system size.

While our toy error model is “non-interacting”—that is, it assumes L/λ independent error processes which, in sum, take the form described in Eq. 8.14—a more careful treatment of the error process, including interactions between defects, as in the real model, would result in an error probability *smaller* than the one calculated here. In Sec. 8.4, we provide numerical evidence that the lifetime of the Ising model in the presence of the protocol scales exponentially with the number of measurement patches, as anticipated by the upper bound in Eq.8.14.

Error correction at any measurement density

A key feature of the protocol is the ability to provide an error correcting threshold temperature at any finite measurement density. In particular, for a fixed measurement density m , and fixed measurement and bulk length scales λ_m and λ_b , respectively, it is still possible to satisfy the rate assumptions of Sec. 8.3 by tuning temperature sufficiently low. Each rate assumption does not explicitly depend on total system size L , only unit cell size λ .

In practice, larger λ_b (alternatively, smaller m) will result in lower threshold temperatures simply because the temperature must be lower to satisfy the rate assumptions of conditions (1) and (2). We provide explicit evidence of this scaling in Fig. 8.5.

8.4 Finite Temperature Simulations

In this section, we present the numerical simulations of the protocol on finite-size systems of length L . We consider systems with unit cells of size λ with $\lambda_m = 3$ measured sites in each unit cell, and a measurement fraction of $m \equiv \lambda_m/\lambda$.

Fig. 8.2 depicts the scaling of the system lifetime enhancement with temperature for several system sizes. Below a certain temperature, the system lifetime increases exponentially with system size. Due to finite size effects, it is difficult to extract an unambiguous threshold temperature, but below $T \approx 0.16$, the lifetime increases exponentially with larger system size. We estimate the threshold by fitting $\Gamma(T)^{-1}$ to $1 + \exp(-a * (T - T_{\text{th}}))$. The inset of Fig. 8.2 shows the finite-size scaling of T_{th} , which suggests that T_{th} remains non-zero in the limit of $L \rightarrow \infty$; this demonstrates that this protocol has a finite-temperature threshold in the thermodynamic limit. In this limit, we find $T_{\text{th}} = .155(6)$.

Fig. 8.3 (top) depicts the finite-size scaling of the lifetime enhancement for temperatures below and above the threshold. Note that for systems above $T \approx 0.16$, larger system sizes asymptote to a constant lifetime enhancement, whereas for models below $T \approx 0.16$, the

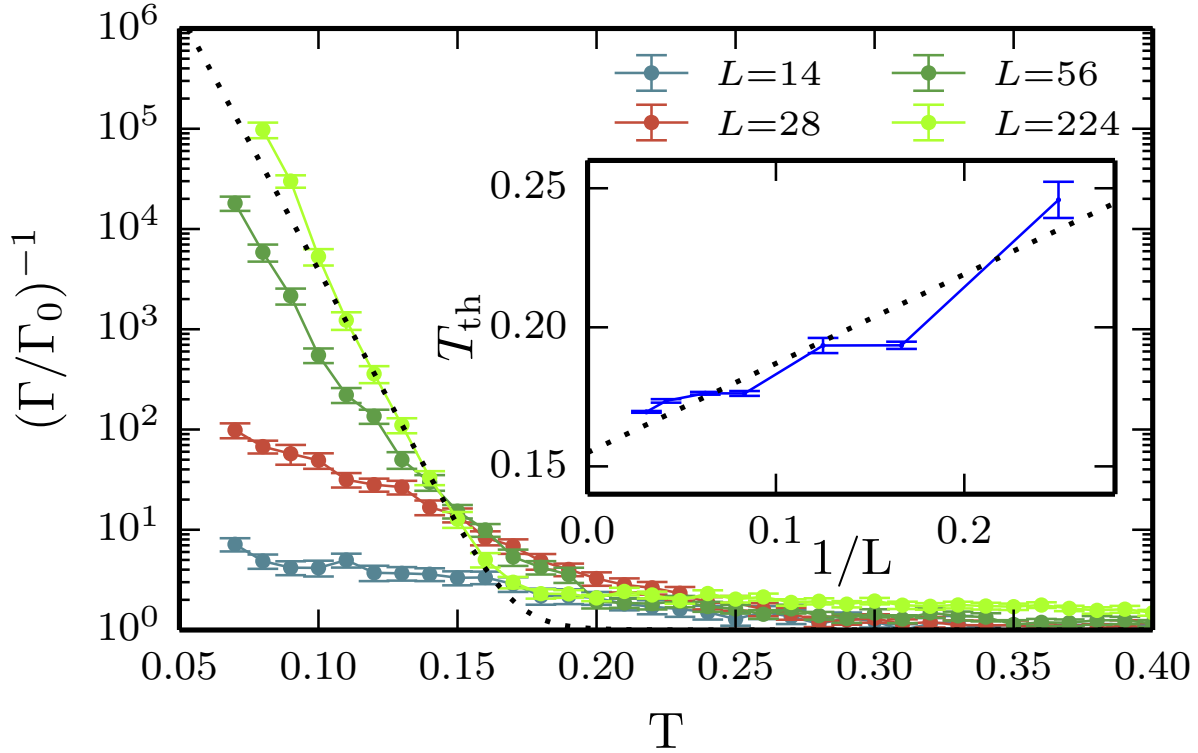


Figure 8.2: Lifetime *enhancement* of several system sizes L as a function of temperature T using a measurement fraction of $m = 3/7$. Dotted line indicates example fit for threshold temperature T_{th} extraction for $L = 224$ data. Below $T \approx 0.16$, we find the lifetime to grow exponentially with L , indicative of a finite-temperature threshold. *Inset*: finite size scaling of T_{th} with inverse length $1/L$. Extrapolation to the infinite system limit yields a threshold temperature of $T_{th} = .155(6)$.

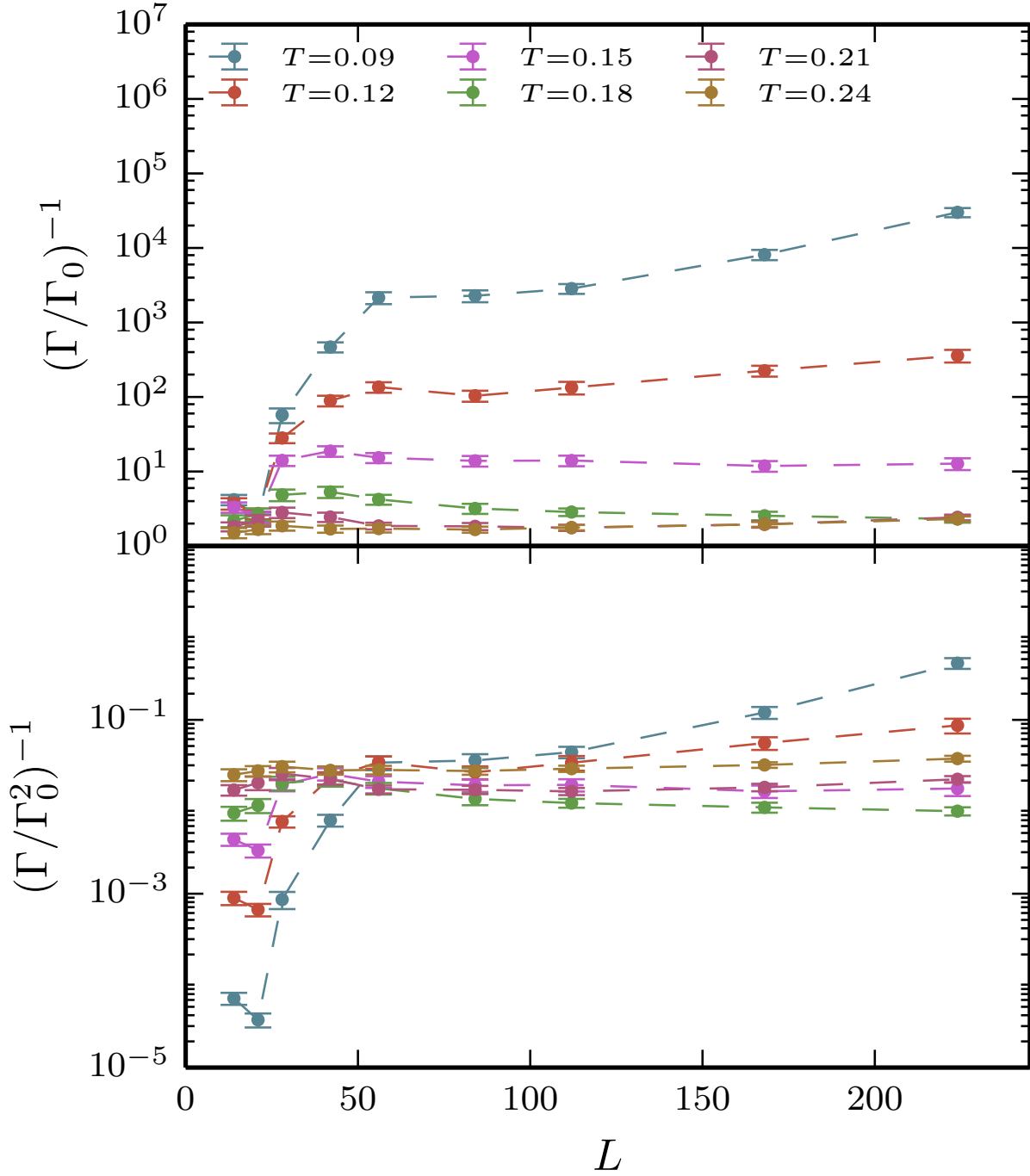


Figure 8.3: (Top) Lifetime enhancement for temperatures both below and above the threshold as a function of system size L , using a measurement fraction $m = 3/7$. Note the monotonic growth in lifetime at low temperatures, as well as the plateau in lifetime for moderately sized systems. A $3/7$ measurement fraction was used for this data. (Bottom) The same data rescaled by an additional factor Γ_0^{-1} , to emphasize the origin and scaling of the plateau in system lifetimes.

lifetime grows monotonically with system size. We find that beyond $L = 100$, finite size effects are significantly reduced, as small-system sizes cannot easily suppress second order errors, such as defects escaping from measurement patches or multiple pairs of defects in the system. Such errors are actually uncorrectable for systems where $L/\lambda \leq 4$ —hence the plateau appearing around $L = 50$ to $L = 100$. Above these system sizes, the exponential scaling returns. This plateau is of height $\mathcal{O}(\Gamma_0^{-2})$ —the characteristic timescale of these “second-order” events. This scaling is made apparent in Fig. 8.3 (bottom), where the lifetime has been scaled by Γ_0^{-2} for each temperature.

Fig. 8.4 depicts the lifetime enhancement as a function of temperature for several different measurement fractions as well as different energy scales, Δ . It is evident that measuring a smaller fraction of the lattice causes the threshold temperature to shift downwards. This dependence of the threshold temperature on the measurement fraction is depicted explicitly in Fig. 8.5 (left panel).

The scaling of the threshold temperature with Δ is presented in Fig. 8.5 (right panel). By contrasting the left and right panels of Figure 8.4, one can deduce the relative benefits of error suppression via more measurement resources versus error suppression via hamiltonian engineering (i.e., a larger gap to excitation).

8.5 Generalization to Higher Dimension

In this section, we sketch how the algorithm presented in Sec. 8.3 generalizes to a higher dimensional stabilizer quantum memory—the 2D toric code. Where the dynamics of the 1D Ising model are typified by one dimensional random walks of defects, the nonequilibrium dynamics of the toric code are driven by two dimensional random walks of quasiparticle excitations. Consider the toric code hamiltonian:

$$H_{\text{TC}} = -\Delta_e \sum_v A_v - \Delta_m \sum_p B_p, \quad (8.15)$$

$$A_v \equiv \prod_{j \in v} \sigma_j^z, \quad B_p \equiv \prod_{j \in p} \sigma_j^x, \quad (8.16)$$

where v denotes the 4-spin vertices of the square lattice, and where p denotes the 4-qubit plaquettes on the edges of a 2D square lattice[76]. While domain wall excitations in the 1D ising model are associated with -1 eigenstates of the $\sigma_z^i \sigma_z^{i+1}$ stabilizers, quasiparticle excitations for the toric code are associated with -1 eigenstates of the A_v and B_p stabilizers as defined in Eq. 8.16.

Broadly speaking, the algorithm is identical, but instead of having “patches” of measurement, there are measurement “rails”, as indicated in Fig. 8.6. One such set of rails must exist for both types of excitations in the toric code—that is, one for the B_p stabilizers, and another set of measurement rails for the A_v stabilizers. The error detection and correction can then be performed completely in parallel for both types of excitations, as they are independent. “Centering” of defects on rails amounts to shift-swapping defects into the center of

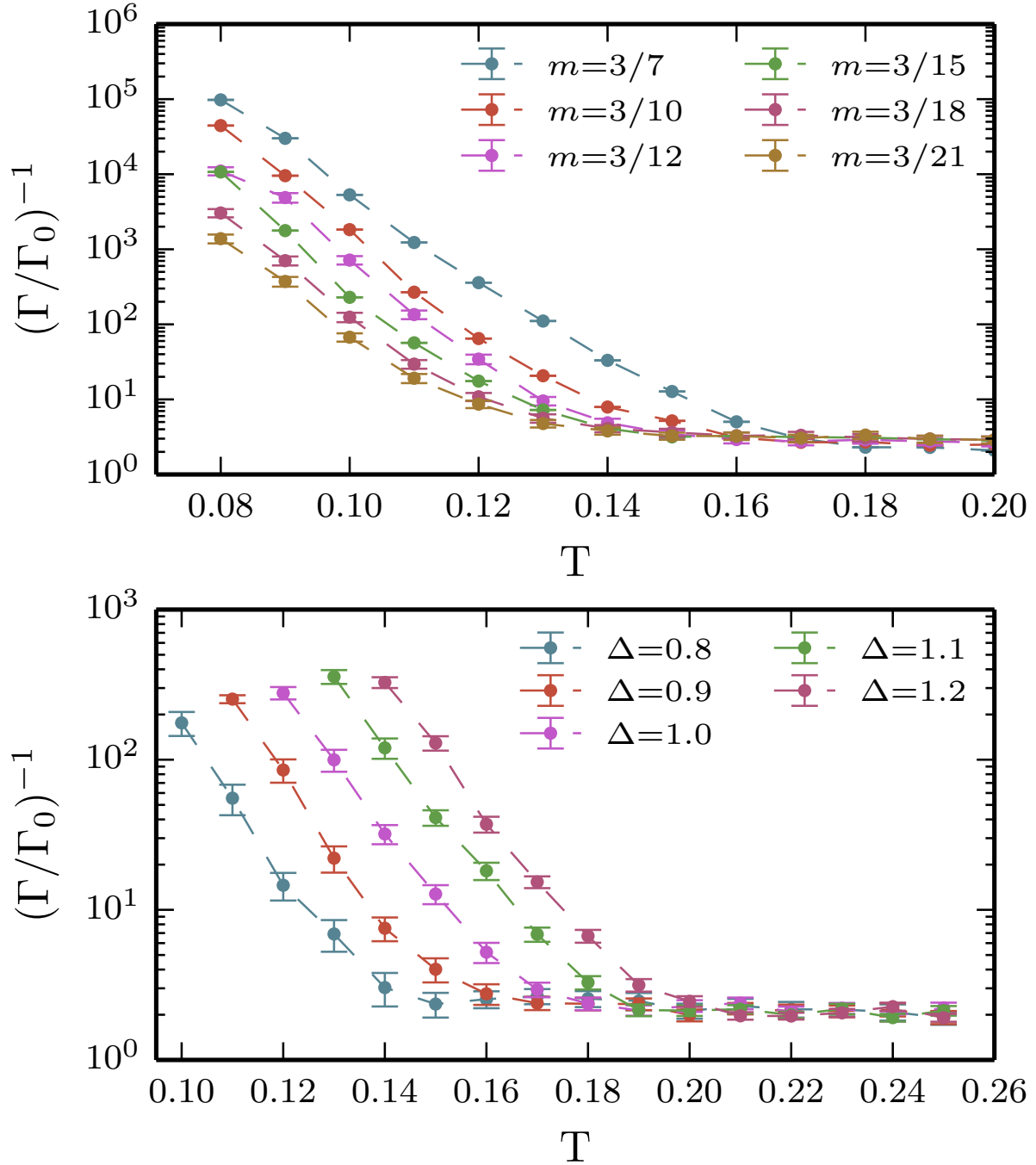


Figure 8.4: (Top) The lifetime enhancement as a function of temperature for several measurement fractions m of 32 (i.e., $L = 32 \cdot 3/m$). Note the threshold temperature decreases with m . (Bottom) Lifetime enhancement as a function of temperature for a variety of energy scales Δ (see Eq. 8.1) for an $L = 224$ system with $m=3/7$.

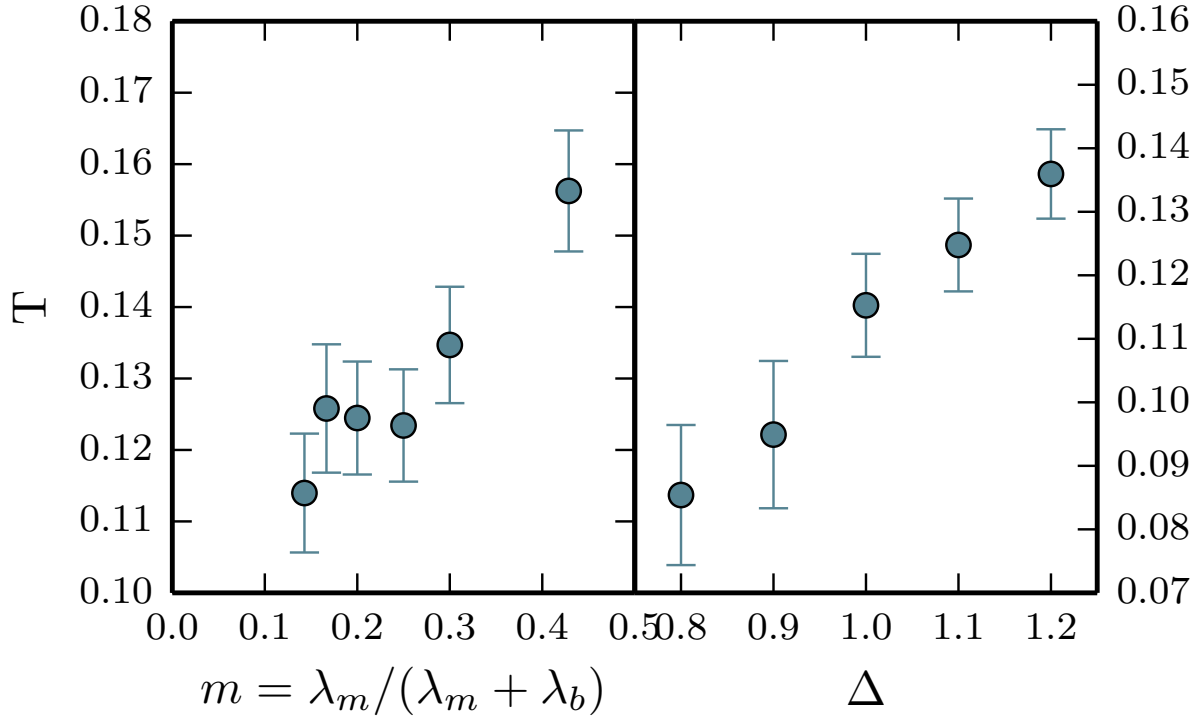


Figure 8.5: The threshold temperatures as a function of measurement fraction, m (left), and the Hamiltonian energy scale Δ (see Eq. 8.1)(right). At zero measurement fraction, the critical temperature is 0, and at unit measurement fraction, the critical temperature is $\mathcal{O}(1)$.

the measurement rail[54]. We conjecture that a sparse measurement strategy with randomly placed measurement patches of fixed diameter might exist for sufficiently large and sufficiently cold systems. However, the rail geometry of Fig. 8.6 is the simplest geometry that allows us to argue for a threshold temperature for the toric code, based on a generalization of the simple error model used for the Ising model in Sec. 8.3. The only difference between the upper bound to the expected scaling of the probability of uncorrectable errors in the toric code versus the expected scaling of the Ising model (i.e., Eq. 8.14) is the prefactor becomes $\propto (L/\lambda)^2$ for the toric code instead of L/λ , where L represents the linear dimension of the toric code.

8.6 Discussion

We have provided numerical and theoretical evidence of a limited measurement error correction protocol for a stabilizer code with string-like excitations. The primary technical

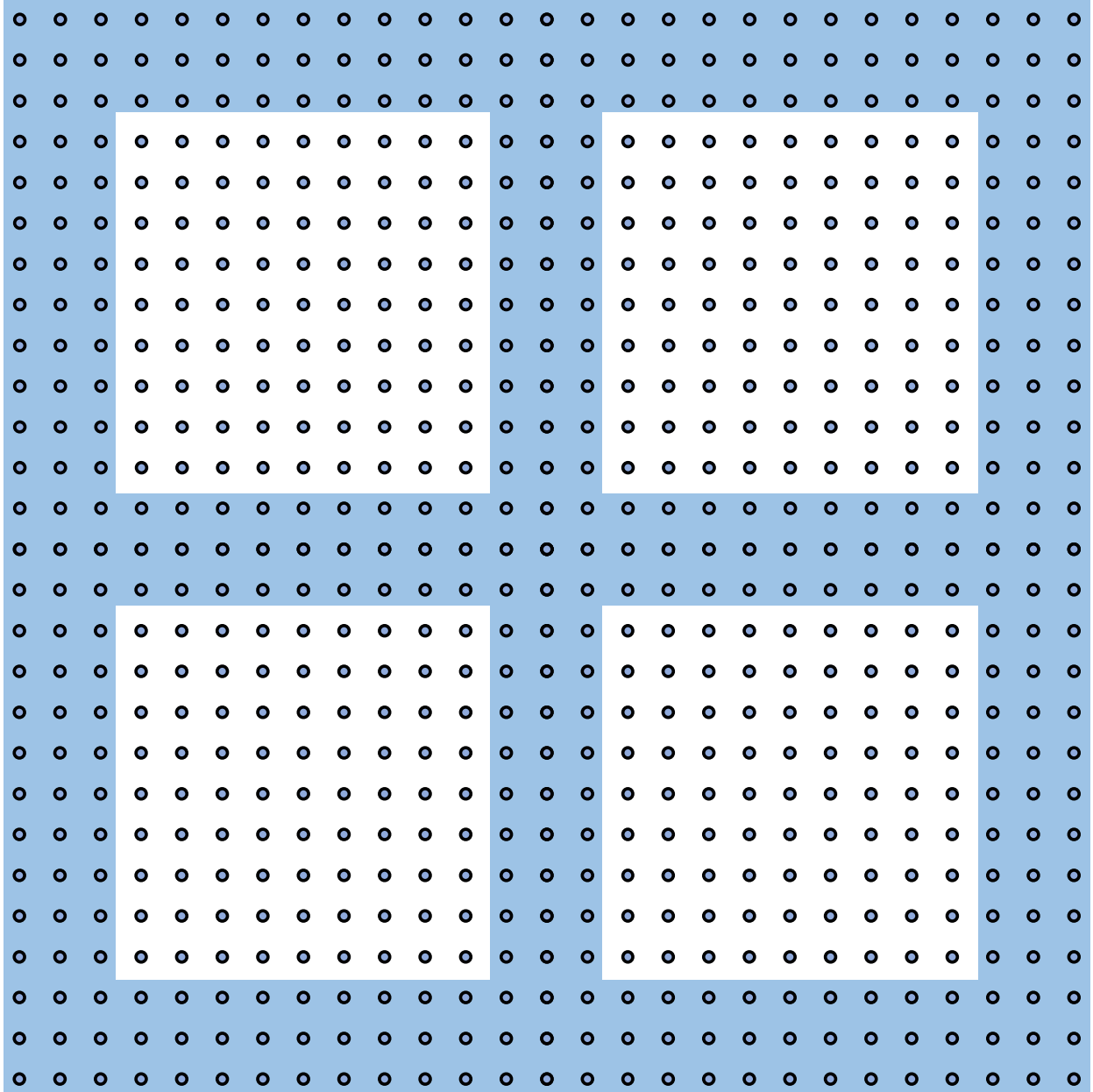


Figure 8.6: Here we sketch one possible $m = 63/144$ (for a 12×12 unit cell) geometry for the measurement rails for a realization of our protocol. More sparse geometries can be realized simply by moving the rails of measurement farther apart. Measured sites are in light blue, and vertex locations for the toric code are circles. Spin variables (not pictured) reside directly between neighboring vertices.

innovation of our algorithm is a Bayesian decoding scheme for pairing defects based on partial information, sketched in Sec. 8.3. When combined with the measurement-free defect localization technique developed in Ref. [54], this decoding scheme performs error correction efficiently and results in a stable quantum memory at temperatures below an empirically determined threshold temperature. So long as an appropriate geometry of measurement devices is in place, and so long as defects undergo diffusive motion via coupling to a thermal bath, this scheme can be extended to higher dimensional stabilizer systems like the toric code, as demonstrated in Sec. 8.5.

Our results for variable measurement fraction complement what is known about decoders in the presence of *noisy* measurements[97]. Figures 8.4 (top) and 8.5 (left) demonstrate how a reduction in the measurement fraction in the lattice corresponds to a concomitant decrease in the threshold temperature, similar to how thresholds are known to be reduced when increasing the noise on measurements.

More fundamentally, our algorithm can be understood as an entropy reduction scheme. Configurations that give rise to errors in the encoded subspace are exponentially suppressed as system size is made larger. This is in contrast with “energetic” suppression—that is, suppression by widening the gap to excitations, Δ (or equivalently lowering the operating temperature). This tradeoff between entropic and energetic contributions is depicted in Figures 8.4 (top versus bottom) and 8.5 (left versus right). Depending on the resource requirements of a particular architecture, the threshold temperature can be tuned either by engineering a larger gap, Δ , or by changing the number of measurements used. In practice, this will depend on the lowest effective temperature available, the maximum measurement rate, as well as the practical difficulty of employing more measuring devices.

8.7 Acknowledgments

This material is based upon work supported by DARPA under Grant No. 3854-UCB-AFOSR-0041 and by the National Science Foundation under Grants No. PIF-0803429 and No. CHE-1213141. CDF was supported by the NSF Graduate Research Fellowship under Grant DGE-1106400 and by the DOE Office of Science Graduate Student Research (SCGSR) program under contract number DESC0014664. MS was supported by the Laboratory Directed Research and Development program at Sandia National Laboratories. Sandia National Laboratories is a multimission laboratory managed and operated by National Technology and Engineering Solutions of Sandia, LLC., a wholly owned subsidiary of Honeywell International, Inc., for the U.S. Department of Energy’s National Nuclear Security Administration under contract DE-NA-0003525.

8.8 Bayesian Decoding for the Ising Model

In this section, we provide further discussion of Eq. 8.11, as well as analytic and numerical arguments for how it can be more simply approximated.

First, we decompose $P(d_1 : d_2)$ into two pieces: a combinatorial piece, and a dynamical piece.

For the combinatorial piece, note that a necessary condition for pairing to be possible is for both defects belonging to a pair to actually be measured. That is, there might be a large number of measured defects d_1, d_2, \dots, d_{n_m} , but the pairing defect for some of these defects might not be measured. Among those defects which are both measured, and which have their pair also measured, then the probability of selecting two defects that are a pair is simply the combinatorial factor $1/\binom{N_{\text{measured pair defects}}}{2}$ where $N_{\text{measured pair defects}}$ counts the average number of measured defects for which their pair is also measured.

The dynamical piece is the probability that d_1 and d_2 are defects whose pairs are also measured. This probability depends on how quickly defects make excursions to measurement sites, as well as how quickly defects are being paired—either erroneously or correctly—by the protocol. We can crudely lower bound this by taking the equilibrium defect distribution, and calculating the probability that a pair of defects lands on a measurement patch. λ_m/λ sites have measurement operators, thus $(\lambda_m/\lambda)L\gamma_+$ is an underestimate of the number of defects on measurement patches. This is an underestimate because the protocol is actually more efficient at concentrating defects on measurement patches than equilibrium dynamics is. Given $L\gamma_+$ pairs, this amounts to a binomial counting argument, and the expected number of measured pairs is simply $L\gamma_+(\lambda_m/\lambda)^2$. Thus, a lower bound to the equilibrium probability of two selected defects being a measured pair is simply $(L\gamma_+(\lambda_m/\lambda)^2)/(L\gamma_+) = (\lambda_m/\lambda)^2$.

As mentioned in Sec. 8.3, $P(d_1^{x_1, t_1} \wedge d_2^{x_2, t_2})$ is the probability that two defects are measured, one at (x_1, t_1) , and the other at (x_2, t_2) . This can be decomposed:

$$\begin{aligned} P(d_1^{x_1, t_1} \wedge d_2^{x_2, t_2}) = & \quad (8.17) \\ P(d_1^{x_1, t_1} \wedge d_2^{x_2, t_2} | d_1 : d_2) P(d_1 : d_2) + & \\ P(d_1^{x_1, t_1} \wedge d_2^{x_2, t_2} | d_1 \not d_2) (1 - P(d_1 : d_2)) & \end{aligned}$$

The first term is precisely Eq. 8.12, multiplied by $P(d_1 : d_2)$, which we estimated above. The second term represents the probability of two measurement events, conditioned on those events *not* being part of a pair. This is essentially the probability that two independent measurement events have occurred, which is approximately the probability that two independent creation events have occurred (assuming defects are measured suitably efficiently). For a suitably large system at moderately low temperature, this probability can be estimated as $\propto (|t_2 - t_1|^2)/(L\gamma_+)^{-2} := \delta(L, T, \Delta t)$.

In practice, at low temperature for moderately sized systems, $P(d_1 : d_2)$ is very nearly 1. This arises from the low density of defects meaning that only rarely are there even a pair of defects in the system. Of course, if system size is made sufficiently large, this bare

probability will become diminished, but it is still the case that defects within a separation distance $2\sqrt{D|t_2 - t_1|}$ are, more often than not, a pair at low temperature. For the same reason, $P(d_1^{x_1, t_1} \wedge d_2^{x_2, t_2} | d_1 : d_2)$ is very nearly 0 because this probability is roughly equivalent to the probability that two independent pair creation events have occurred, which is unlikely at low temperature and moderate system size. Again, for sufficiently large systems this probability grows, but it is likewise the case that this probability is small for defects within a distance $2\sqrt{D|t_2 - t_1|}$. Then, if we write $P(d_1 : d_2) = 1 - \epsilon(L, T)$, and perform some rearranging:

$$\begin{aligned}
 P(d_1 : d_2 | d_1^{x_1, t_1} \wedge d_2^{x_2, t_2}) &= \\
 &= \frac{1}{1 + \frac{P(d_1^{x_1, t_1} \wedge d_2^{x_2, t_2} | d_1 : d_2) \epsilon(L, T)}{P(d_1^{x_1, t_1} \wedge d_2^{x_2, t_2} | d_1 : d_2)(1 - \epsilon(T))}} \\
 &\geq \frac{1}{1 + \frac{\frac{\delta(L, T, \Delta t)}{1 - \epsilon(L, T)}}{P(d_1^{x_1, t_1} \wedge d_2^{x_2, t_2} | d_1 : d_2)}} \\
 &\approx \frac{1}{1 + \frac{\delta(L, T, \Delta t)}{P(d_1^{x_1, t_1} \wedge d_2^{x_2, t_2} | d_1 : d_2)}}
 \end{aligned} \tag{8.18}$$

Thus, only when $P(d_1^{x_1, t_1} \wedge d_2^{x_2, t_2} | d_1 : d_2) \ll \delta(L, T, \Delta t)$ is this factor not equal to 1. This naturally occurs when comparing defects that are much farther apart than diffusive motion would usually allow. For example: for a very large system, if one defect of a pair is measured at site 0 and another defect belonging to another independent pair is measured at site $L/2$ shortly thereafter (compared to the timescale for defect motion), it is exceedingly unlikely for these two measured defects to be a pair because it's exponentially unlikely for such a long random excursion to occur. In this way, the factor $P(d_1^{x_1, t_1} \wedge d_2^{x_2, t_2} | d_1 : d_2)$ serves as an indicator function which answers the question, "Could these two defects have arisen from a random walk starting in the same place?". The factor $\delta(L, T, \Delta t)$ sets the cutoff for a plausible excursion—i.e., when the error function is much less than this term, the denominator of 8.18 blows up, and the probability of performing that fusion is essentially zero.

In practice, the precise details of these additional factors arising from Bayes theorem aren't too important for the protocol to function, and we find that using the conditional probability $P(d_1^{x_1, t_1} \wedge d_2^{x_2, t_2} | d_1 : d_2)$ itself as a proxy for the full expression from Bayes theorem is sufficient to reliably correct errors. We provide some heuristic comparisons of different decoding schemes in Appendix 8.10.

8.9 Alternative Algorithm for Estimating Defect Lifetimes

The algorithm, as presented in Sec. 8.3, is susceptible to errors due to systematically underestimating defect lifetimes. In practice, this error rate is small—small enough that it was not detectable in our numerical studies—but it is nonetheless present and has the potential to spoil the increase in lifetime with system size for the protocol. In this section, we outline how this problem introduces a system size independent uncorrectable lengthscale into the algorithm, and we provide an alteration to our presented algorithm that can account for these errors, restoring the expected system size scaling.

The maximum correctable lengthscale

If we denote the timescale over which defects escapes measurement patches on average as τ_{esc} , then a defect pair that is separated by much more than $\sqrt{D\tau_{\text{esc}}}$ will be overwhelmingly likely to escape from its measurement patch before the denominator of Eq. (8.12) can grow large enough to match the defect to its pair—potentially spoiling the system size scaling of the algorithm.

In practice, by occasionally scaling the size of the measurement patches, λ_m , keeping the ratio of measured sites to unmeasured sites fixed, this maximum distance can be tuned larger. For our simulations, we did not need to perform this measurement patch scaling, because the rate of defect escape was so small, even for $\lambda_m = 3$. Care must be taken, however, because measurement patches cannot be made arbitrarily large without violating the condition mentioned in point (2).

Now, Suppose two defects come into a configuration where they are separated by a distance $C\sqrt{D\tau_{\text{esc}}}$ for an integer C and system diffusion constant D . We will estimate the C after which it is just as likely for a defect pair to be corrected by the algorithm as it is to cause an error. Without loss of generality, define the left defect to be at position 0.

Suppose the left defect has recently escaped a measurement patch and been recaptured, thus its estimated age is 0. For the error correcting protocol to be able to pair these defects, it must remain on its measurement patch for a time $C \cdot \tau_{\text{esc}}$. But over a timescale τ_{esc} , the defect is equally likely to escape its measurement patch as it is to remain on it, resetting its effective age. Treating this as a binomial process, we need to estimate the expected amount of time it takes the defect to remain on its measurement patch for C consecutive timescales τ_{esc} . Call this timescale $T(C, \tau)$ for C consecutive events with timescale τ . More colloquially—this is equivalent to the expected number of coin flips before a coin has a run of C “heads” in a row. For an event with probability p of occurring, this takes the form

$$T(C, \tau) = \tau \frac{p^{-C} - 1}{1 - p} \quad (8.19)$$

Thus, after a time $T(C, \tau)$, a defect could, in principle, be paired with another defect a distance $C\sqrt{(D\tau)}$ away by the algorithm.

Recall that a diffusion process with constant D will, in δt time, become displaced by a distance $\delta x = \sqrt{D\delta t}$. Because the defects are trapped by the measurement patches, the diffusion rate must be renormalized: $\delta x = \sqrt{D\delta t \cdot (\lambda)/(\gamma_0\tau_{\text{esc}})}$. This is because it takes a time τ_{esc} for a defect to actually perform escape from a measurement patch to perform a random step, and this random step has a characteristic length equal to the length of the unit cell, λ . Thus, on average, it takes a time $\delta t = C^2\gamma_0\tau_{\text{esc}}^2/\lambda$ for a diffusive process to perform an excursion of distance $C\sqrt{D\tau_{\text{esc}}}$.

Note that when the timescale over which it takes a defect to remain on a patch C consecutive times equals the timescale it takes a defect to travel a distance $C\sqrt{(D\tau_{\text{esc}})}$, it is no longer likely for error correction to work. This only becomes worse as defects become more separated—the pairing defect is more likely to diffuse than it is likely that its pair will remain trapped on a measurement site. Setting these timescales equal results in the transcendental equation,

$$\tau_{\text{esc}}2^C = \gamma_0 C^2 \tau_{\text{esc}}^2 / \lambda \quad (8.20)$$

Asymptotically for large $\tau_{\text{esc}}/\lambda$, $C = \log(\gamma_0\tau_{\text{esc}}/\lambda)$. Thus, the lengthscale $\sqrt{D\tau_{\text{esc}}} \log(\gamma_0\tau_{\text{esc}}/\lambda)$ is approximately the maximum correctable lengthscale for our protocol, in the absence of any other corrective measures. Note that, for our system parameters, this is many thousands of unit cells, and therefore was not detectable by our finite system size analysis.

Accurate lifetime estimation

To ameliorate this issue, we can modify our protocol with additional steps to keep track of defect lifetimes. That is, within step 1 of the algorithm, perform the following subprotocol:

Let each measurement patch, m_i , have two internal clocks, T_1^i and T_2^i .

1. If a measurement patch becomes unoccupied without a corrective operation being applied, record the time at which the patch was measured as empty, $T_1 = T_{\text{empty}}$, and keep the most recent lifetime, t_{age} in memory with a decay constant set by τ_{decay} . Thus, $T_2 \equiv t_{\text{age}}(t) = t_{\text{age}_0} \exp(-(t - T_{\text{empty}})/\tau_{\text{decay}})$.

2. If a defect is subsequently remeasured on this patch at time t_j , treat it within the original protocol as if it had been measured at time $t_{\text{age}}(t_j)$. So long as a defect is on the measurement patch, leave T_2^j constant, and update $T_1 = T_{\text{current}}$, where T_{current} is the current system clock time.

3. If a defect is subsequently measured on patch m_i with no “active” memory of a lifetime—i.e., $T_2^i < .01$ —calculate the probabilities given by Eq. 8.18 between m_i and all other unoccupied measurement patches, m_j , using $T_{\text{current}} - T_1^j$ as the diffusion timescale in Eq. 8.18 for the current system time, T_{current} . Then, probabilistically set T_2^i equal to T_2^j , and then reset T_2^j to 0.

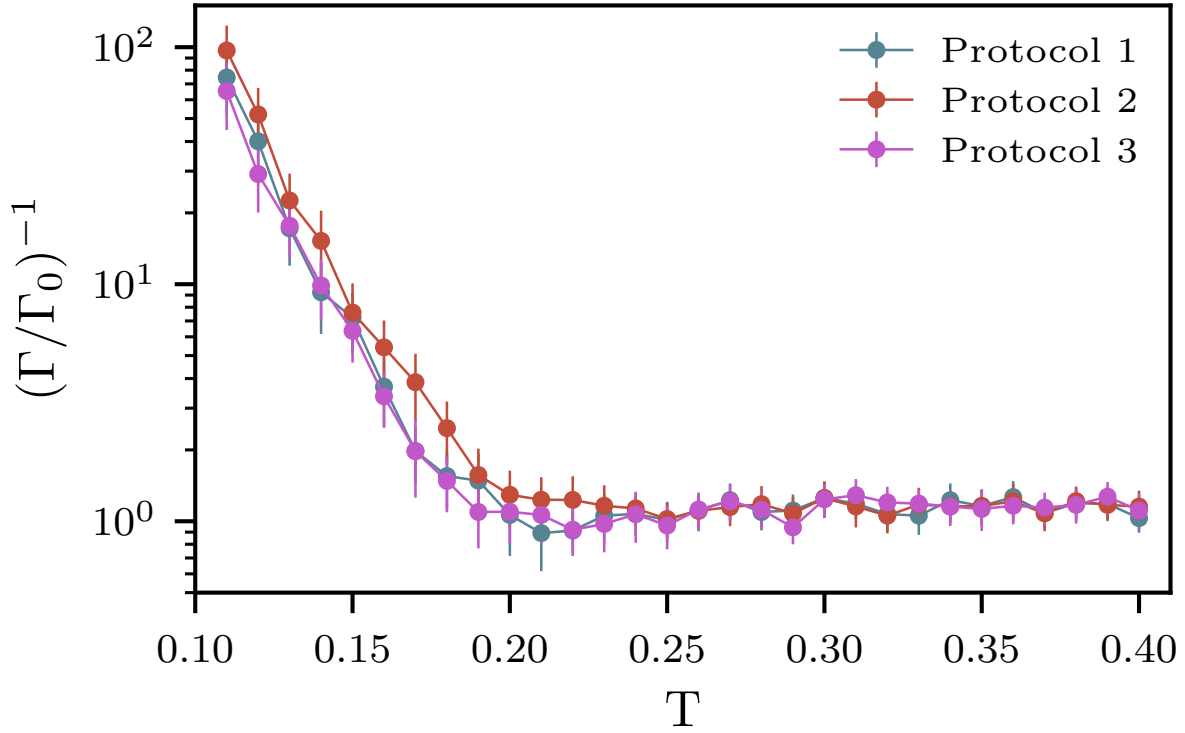


Figure 8.7: Comparison of three different functional forms used as proxies for $P(d_1 : d_2)$. Protocol 1 is simply the error function expression given by Eq. 8.12. Protocol 2 is the probability density $\frac{1}{2\pi D|t_2-t_1|} \exp(-\frac{|x_2-x_1|^2}{2D|t_2-t_1|})$. Protocol 3 is the more complicated expression in the last line of Eq. 8.18. In practice, each approximation for $P(d_1 : d_2)$ is seen to perform approximately equally well.

Finally, we impose that τ_{decay} is several times larger than the diffusive timescale for defects to migrate between measurement patches, but still much smaller than the characteristic timescale over which unpaired defect creation occurs within a unit cell. This ensures that lifetimes T_2^i decay reasonably quickly if a pair of defects happens to self-annihilate far away from a measurement patch, and will be near 0 should a new creation event occur, but also ensures that lifetimes are kept in memory long enough to be useful for subsequent re-detection events of escaped defects.

8.10 Alternative decoding schemes

While the approximate Bayesian fusion probability expressed in Eq. 8.18 works well in practice, we find empirically that the precise prefactors of the probability calculation are not terribly important for the decoder functioning correctly. That is, we find that the final expression in Eq. 8.18 works about as well as $P(d_1^{x_1, t_1} \wedge d_2^{x_2, t_2} | d_1 : d_2)$, and that even using the raw probability density,

$$\frac{1}{2\pi D|t_2 - t_1|} \exp\left(-\frac{|x_2 - x_1|^2}{2D|t_2 - t_1|}\right) \quad (8.21)$$

from Eq. 8.12 serves as a decent proxy for the probability, even if this is mathematically dubious in principle.

What is most important for the function of the protocol is that the fusion probability correctly incorporates the expectation that defects are *diffusive*. An amount of sloppiness in this calculation is tolerable, because the defects are efficiently trapped by the protocol, and remain trapped for a long time relative to the diffusive timescales for the system. But so long as defects that are plausibly “close” to one another are the defects that are fused, then we find the protocol to extend system lifetimes effectively.

We plot a comparison of the three aforementioned decoding schemes in Fig. 8.7

Chapter 9

Additional topics

1:44 For, lo, as soon as we introduce change into our computational models, many notions that were previously straightforward become problematical.

King James Programming Tumblr

Throughout my graduate work at Berkeley, I also worked on a variety of topics unrelated to the broader research program of developing a finite temperature fault tolerant quantum memory. This chapter presents brief motivations and short summaries of those other works.

9.1 Tensor Networks, Transport, and Efficient State Ansatzes

Background

A generic quantum state of n qubits requires 2^n complex amplitudes to fully specify. While this cost is prohibitive for describing all but the smallest quantum mechanical systems on classical computers, it is possible to very accurately *approximate* quantum states, or even exactly describe broad classes of physically relevant states if one chooses the right sort of state ansatz using purely classical resources.

Quantum state approximation has a long and productive history. The modern thrusts in the field mostly share a common ideological if not technical underpinning with Steve White's Density Matrix Renormalization Group (DMRG) technique from the early 90s[132]. The common thread of these techniques is to carefully choose a basis of states that well describes a class of quantum states (e.g. ground states of physically interesting Hamiltonians) that is also simultaneously numerically efficient to calculate. DMRG, in essentially the same form as it was originally proposed, is still used to calculate observables for systems that have approximately one-dimensional symmetries, like spin chains[131, 84] or even the electronic properties of polymers[7].

The modern interpretation of techniques like DMRG casts it as a variational optimization of a set of *local* degrees of freedom[27]. For example, suppose we aim to describe the ground state of a one-dimensional spin chain on N sites with qubit degrees of freedom on each site. Then, the most general ground state wavefunction could be written:

$$\Psi(x_1, x_2, \dots, x_N) = \sum_i^{2^N} c_i |\phi_{x_i}\rangle \quad (9.1)$$

where c_i are complex amplitudes, and ϕ_{x_i} are some complete basis for the Hilbert space on N sites (e.g., $\phi_{x_k} \in \{|\uparrow\rangle, |\downarrow\rangle\}^{\otimes N}$). If, however, we know that the relevant physics for the problem can be captured entirely with *local* information (i.e., perhaps we know, a priori, that a mean field approximation will give a decent approximation to the exact observables of the problem), we might intuit that this function can be reparameterized by local variables as a so-called *Matrix Product State*[27] (MPS):

$$\Psi(x_1, x_2, \dots, x_N) = \sum_{i,j,\dots,z} A_{i,j}^{x_1} B_{j,k}^{x_2} C_{k,l}^{x_3} \dots Y_{y,z}^{x_{N-1}} Z_{z,i}^{x_N} \quad (9.2)$$

where the A, B, \dots, Z objects' lower indices run from 1 to an upper index d called the *bond dimension*. These 2 by d by d objects $A_{i,j}^{x_k}$ are called *tensors*. Often, especially in translationally invariant systems, the different tensors A, B etc. are numerically identical.

Note that by choosing d large enough, this class of states can describe any state that Eq. 9.1 can. The modern understanding of DMRG casts it as one particular procedure for variationally updating the parameters of the different $A_{i,j}^{x_k}$ at fixed d so that the resulting state well describes the physics of some problem of interest.

Of course, this is only one choice for how to locally parameterize an ansatz for a quantum state. Drawn as a graph, the MPS parameterization can be visualized as a linear chain, where vertices in the chain index the different $A_{i,j}^{x_k}$, and the edges indicate different tensor indices, as in Fig. 9.1. By considering more complicated graphs, more rich classes of quantum states can be considered.

One particular state ansatz, pioneered by Guifré Vidal, called the Multiscale Entanglement Renormalization Ansatz (MERA) was shown to be particularly useful for describing systems with specific entanglement properties[123]. Specifically, MERA states can well describe systems with entanglement area laws, and power-law decaying correlations. In a long series of follow-up works, Glen Evenbly and Vidal used MERA as well as related schemes to attack numerous one and two dimensional lattice system of interest in the physics literature[40, 47, 43, 44, 42, 41, 39, 46, 45].

Transport in spin chains

With the power of MERA-style methods for lightly entangled systems well established, it was surprising that these tools had not yet been applied to so-called *non-equilibrium steady*

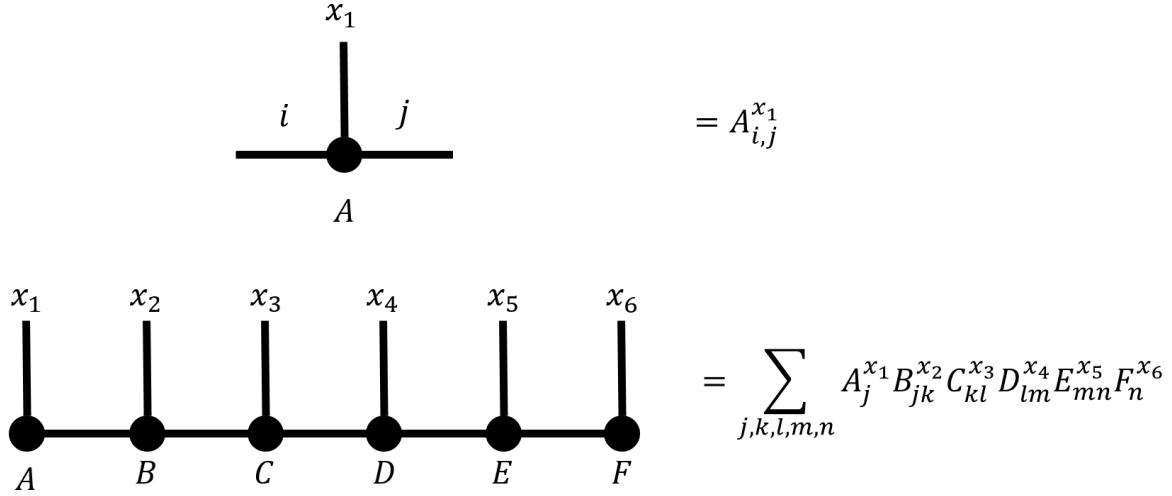


Figure 9.1: Graphical shorthand for tensors. Unique tensors are associated with vertices of graphs. Edges connecting vertices correspond to summed indices. Free edges are free indices.

states (NESS). Such states are typified by two (or more) thermodynamic reservoirs held at different temperatures (or chemical potentials), with some out-of-equilibrium flow occurring between the reservoirs through a channel connecting them.

In joint work with Raghu Mahajan, Sam Mumford, Norm Tubman, and Brian Swingle, we showed that these NESS states are lightly entangled under modest conditions, and numerically studied the transport properties of several toy systems Mahajan et al. [88]. The primary technical result was the proof of the existence of an efficiently calculable MERA-style tensor network representation for NESS states under the condition of local thermodynamic equilibrium. This condition is equivalent to the assumption that the density matrix of the system can be written as:

$$\rho_{NESS} \propto e^{-\sum_x \beta(x) \epsilon(x)} \quad (9.3)$$

where $\beta(x)$ can be interpreted as a *local temperature*, and $\epsilon(x)$ is the energy density.

My primary contribution to the manuscript was extensive numerical calculations of transport properties of a toy out-of-equilibrium fermionic system defined as follows. Consider a one-dimensional fermionic chain partitioned into three contiguous segments: L, C, and R. The base Hamiltonian for the full wire is the simple hopping model,

$$H = \sum_{x=1}^{N-1} -w(c_x^\dagger c_{x+1} + c_{x+1}^\dagger c_x) \quad (9.4)$$

where $N = N_L + N_C + N_R$, and where c, c^\dagger are the usual fermionic creation and annihilation operators. Between the segments connecting L and C and the segments C and R, the hopping matrix element was set to a separate, tunable variable w' . Additionally, in the L and R regions, Lindblad operators of the form $L_j = u^\dagger c$ drove the left and right regions to different tunable temperatures. We solved these systems primarily through exact diagonalization.

We found that many transport properties for this many-body system, like the linear response conductance, thermal conductance, and mutual information, were well described by a simple two-site model in the thermodynamic limit, due to the system's simple entanglement structure. This work was also extended to open fermionic systems by Zanoci and Swingle [139].

Monte Carlo tensor network renormalization

When building tensor network states, the most common procedure for determining the “relevant” basis states for a calculation is use of a dimensionality reduction technique like the Singular Value Decomposition (SVD). That is to say, the tensors, $A_{i,j}$ appearing in the state ansatz, are decomposed into a product of tensors, $A_{i,j} = U_{i,m} S_{m,n} V_{n,j}$ where S is diagonal. By retaining the largest entries of $S_{m,n}$ and truncating the others, one can often retain accuracy while drastically reducing the size of the tensors under study. Unfortunately, this process generically introduces an unavoidable truncation error, which, while small, is necessary to avoid if one wishes to capture long range physics, especially in critical systems.

One promising technique for avoiding this truncation error while still working with tensors of a reduced dimensionality is Tensor Network Monte Carlo [48]. With fixed bond dimension, this technique allows one to faithfully sample the distribution of singular values for the tensors used in an ansatz, thus exactly *exactly* capturing physics at all important length scales. This can be understood as a time-space tradeoff—by working with a fixed, small bond dimension, more time must be spent averaging over monte carlo samples (i.e., relevant singular values) to accurately calculate observables.

In joint work with William Huggins, Miles Stoudenmire, Norm Tubman, and Birgitta Whaley, we extended this technique to more complicated renormalization-based tensor network geometries, and provided benchmark calculations on the 2D classical Ising model, explicitly demonstrating this time-space tradeoff by comparing convergence times with choice of bond dimension size[67]. My role in this work was largely advisory, and William Huggins performed the entirety of the coding and analysis of the data.

ASCI in Practice

Although it is not a tensor network based algorithm, Tubman's recently developed Adaptive Sampling Configuration Interaction (ASCI) belongs, in spirit, to the same family of

algorithms that intelligently search for the most important basis states (in this case, determinants) for use in studying an electronic structure problem[121]. In short, ASCI iteratively improves a wavefunction ansatz by deterministically searching through the space of determinants for determinants that score highly according to a specific objective function. For generic ASCI, this objective function is the CI coefficient:

$$A_i = \frac{\sum_{j \neq i} H_{ij} C_j^k}{E_k - H_{ii}} \quad (9.5)$$

where C_j^k is the configuration interaction coefficient of the j^{th} determinant, E_k is the energy of the wavefunction on the k^{th} iteration of the algorithm, and H_{ij} are Hamiltonian matrix elements between determinants i and j .

This method has proven powerful for tackling many-electron systems with better performance than competing techniques. In joint work with Norm Tubman, Daniel Levine, Diptarka Hait, Martin Head-Gordon, and Birgitta Whaley, we demonstrated the power of ASCI on a variety of chemically interesting/difficult problems, and benchmarked different versions of ASCI which utilized different sorting methods. My primary contribution to this work was the calculation of energies of various transition metal dimers using a mixture of in-house code developed by Norm Tubman as well as the *quantum_chemistry_library*.

9.2 Topology and Geometry in Neural Networks

Late in my graduate career, I became interested in the fundamental theory of neural networks. These tools began to explode in popularity around 2012 when a confluence of better algorithmic techniques, better hardware, and renewed interest allowed researchers to construct larger and more powerful neural networks than had been considered possible previously[81].

At their simplest, neural networks are high-dimensional function approximators comprised of matrix multiplications convolved with the application of some nonlinear function, repeated some number of times. That is, for data-pairs we wish to approximate, $\{(x_i, y_i)\}$, a neural network is a function, f , defined by some set of variables θ which maps the x_i approximately to the y_i . This function can be written:

$$f(x_i) \rightarrow y_i : \phi(W^L \dots \phi(W^2 \phi(W^1 x_j)) \dots) \quad (9.6)$$

where the $\theta = \{W^k\}$ are matrices and are exactly the free parameters to be optimized, ϕ is some nonlinear function (commonly tanh, relu, or sigmoids), and L is some number of *layers* of computation. These functions are commonly visualized as stacks of computations in a graph, see, e.g., Fig. 9.2.

In general, this optimization problem is usually solved by constructing a *loss function* (sometimes, cost function), for example:

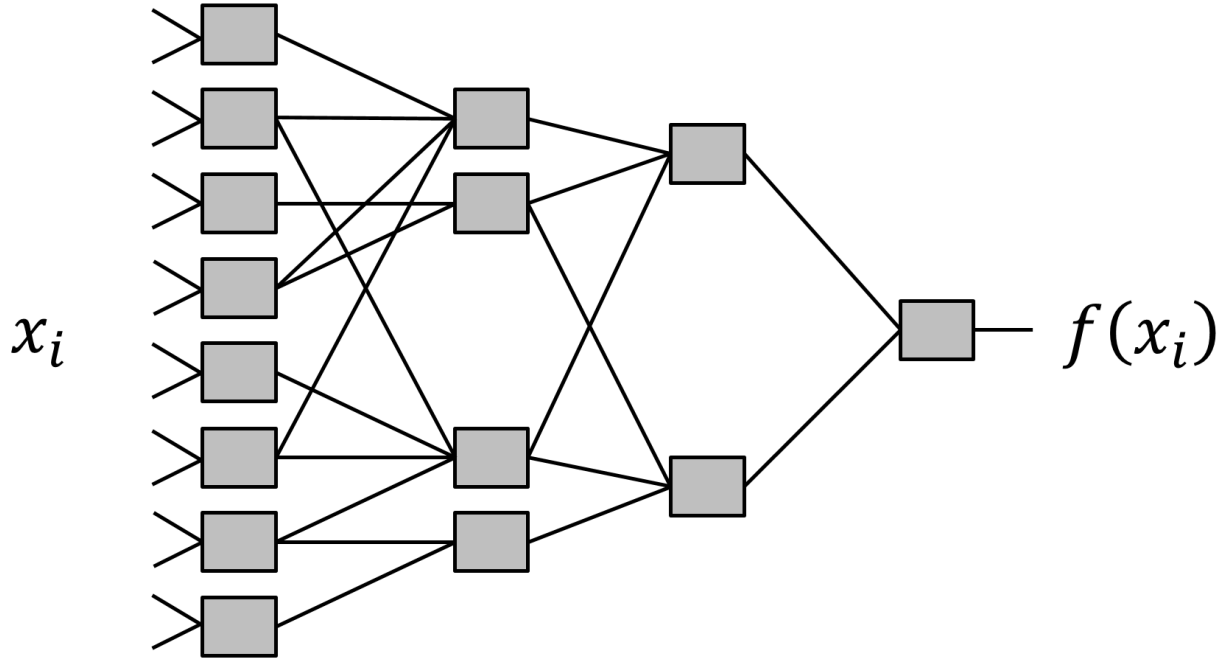


Figure 9.2: A cartoon of a neural network mapping an input vector x_i to a single output. Grey boxes indicate the combined process of multiplying input data by a matrix followed by application of a nonlinearity. Edges indicate flow of data through the graph.

$$L(\{(x, y)\}, \theta) = \langle |f(x_i, \theta) - y_i|^2 \rangle_i \quad (9.7)$$

and then using some variety of stochastic gradient descent to determine an optimal (or sufficiently good) set of parameters θ which minimize the loss. This function is so named because it represents the penalty, cost, or what is *lost* by having a particular set of parameters for a given optimization problem.

In practice, this optimization problem is ludicrously high dimensional, with the number of parameters θ often well exceeding the number of input-output pairs $\{(x_i, y_i)\}$ being used. It is, then, an outstanding question why this procedure results in “good” optimization solutions, even though it is wildly overparameterized.

I became interested in the properties of these loss functions, because I was surprised how little was known about them. Considerable effort had been spent (and is still spent) characterizing the local and global minima of these functions[33], but significantly less was known about, for example, the level sets of these functions. By level sets, I mean families of parameters $\{\phi\}$ such that $L(\{(x, y)\}, \phi) = L_0$ for some fixed L_0 .

In joint work with Joan Bruna, I aimed to characterize these level sets both numerically and theoretically[53]. We discovered that the characteristic curvature of these level sets grows as a power law with inverse loss—in other words, the better the neural network one considers, the family of neural networks that perform equally as well as that network lives on a manifold that grows more jagged as neural network performance increases. This discovery was not too surprising—it is intuitively obvious that the optimization problem becomes harder as loss values decrease.

More surprisingly, we discovered that these level sets were *connected* at essentially all loss values studied for neural networks used in practice. In other words, it is almost always possible to find a continuous path from one model θ_A that achieves loss L_0 to another model θ_B that achieves loss L_0 such that the loss along the connecting path remains fixed at L_0 .

For this work, I devised (several) algorithms for determining connecting paths between models, as well as the code used for solving the neural network optimization problem. Additionally, Joan and I leveraged this observation into a new proof that deep linear networks (i.e., networks like Eq. 9.6 for which ϕ is the identity function) have no local minima, although Joan did the majority of the heavy-lifting for this proof. Finally, we also proved that 1-layer networks with *relu* style nonlinearity, asymptotically do not have any local minima, where *relu* is defined as x for $x > 0$, and 0 for $x < 0$. In particular, if δL_0 is the largest amount one needs to increase in the loss function above L_0 to continuously navigate from θ_A to θ_B , we demonstrated a bound on the size of δL_0 which asymptotically approaches 0 as the number of parameters increases.

Chapter 10

Conclusion

25:34 But the field of computer science and AI research. All attracted bright people who contributed great things to the church

7:2 And it came to pass, that he who fleeth from the noise and confusion of ordinary software engineering or academic research.

26:58 These are the two binary numbers to be added

King James Programming Tumblr¹²³

10.1 Overview and Outlook

Over the course of this thesis, we have developed a set of theoretical tools necessary to understand, first, why the toric code fails as a quantum memory at finite temperature, and second, some algorithmic tricks one can perform to try to restore some order to the stabilizer codes like the toric code.

First, we characterized the finite temperature dynamics of quasiparticle defects in the toric code, and then we leveraged successively more expensive resources (first unitaries, then measurements), to attempt to dissipate these error modes, in the hopes of restoring the scaling of the lifetime of the toric code with system size.

Ultimately, we were able to restore the favorable scaling with system size, but at the necessary cost of introducing some amount of measurement. It's an open question, but I conjecture it is possible to retain this favorable scaling while also relaxing the measurement scheme to use purely local information exchange, like in Chris Harrington's cellular automata type schemes[62]. Specifically: this thesis resolves that any constant density of measurements may be used, but it's possible that once locality constraints are introduced, there could be a finite threshold in measurement density that depends on the locality constraint.

It also remains entirely open whether a true finite temperature fault tolerant quantum memory actually exists in three dimensions. Whether 4 dimensions are necessary, or whether a fractal construction of intermediate dimension can be made is an open question.

While these questions remain, there is still intense effort in developing new and better stabilizer codes for use in near term machines. While these codes are usually not used as Hamiltonian stabilizer codes, the field continues to evolve rapidly.

10.2 Contemporaneous and ongoing works on quantum memories

These ideas are but a small corner of a greater effort to build objects like quantum memories. I would be remiss to not outline, or at least signal-boost several other efforts to construct similar objects that have occurred contemporaneously with my work.

Several contemporaneous efforts have been made to construct models that have strong topological protection. Haah’s code is a beautiful example of a three dimensional stabilizer code with interesting self-correction properties, and its existence contributed to my intuition that a self correcting quantum memory probably exists, but may be terribly difficult to actually construct[60]. Bombin’s work in color codes has similarly been foundational to my understanding of topological stabilizer codes[8, 86]. Pastawski, Yoshida, Harlow, and Preskill’s holographic code construction was the most exciting new idea I encountered in my graduate career, and I’ve followed its development with interest[103]. Probably the most exciting recent work marrying the broad literature of symmetry protected phases and quantum memories is from Roberts and Bartlett [108], where they demonstrate a three dimensional system that retains topological order at finite temperature—though it is yet unknown whether a qubit can be stored in the phase. There have been several attempts to build fractal quantum memories, first by Brell[19], where the quantum memory resides on a lattice with intermediate fractal dimension between 2 and 3. Personally, I find the question of the minimal dimension necessary for a true quantum memory to exist to be fascinating, and fully expect that it need not be, (we hope), exactly 4, or any other integer necessarily.

There has also been a tremendous amount of work into different ways to “fix” the toric code. A research effort concerning so-called welded toric codes introduced by Michnicki has resulted in three-dimensional codes that have power-law energy barriers with size[91], though followup work demonstrated that neither welded codes, nor Haah’s code was topologically ordered above zero temperature, these models are concrete progress towards the vision of the topological qubit[112]. Landon-Cardinal’s quest for topological stability has been inspiring throughout my thesis, as have been his papers on topological order in stabilizer systems, particularly [82, 83]. Similarly, Hutter’s ingenuity in trying to add gadgets to stabilizer systems to stabilize them has been motivating and thoroughly impressive, particularly in [70, 68].

Finally, the best summary of the state of quantum memory research circa my tenure at Berkeley is the thoroughly excellent review by Brown et al. [22], which I could not recommend more highly to any newcomers to this field.

10.3 Epilogue

It has been fascinating to work in this field as real quantum devices have catapulted from theoretical dreams to delicate, experimental realities. The race is far from won, however, and there still remains tremendous work to do before quantum computers can be made practically useful. In fact, the quest for a finite temperature fault tolerant quantum memory is still largely unfulfilled. If this thesis as well as the broader literature on quantum memory is any indication, construction of a truly fault tolerant finite temperature quantum memory may even require completely new ideas.

Ironically, it's entirely possible that practically useful quantum devices will exist before we prove that finite temperature quantum memories can (or can't) be constructed. The prevalence of high fidelity *active* quantum error corrected memories may even obviate the need for a passive version. This is the arc of speculative technologies in a sentence—that sometimes, something else entirely just works better.

There is something uniquely frustrating about the stubbornness with which nature refuses to provide a say one way or the other regarding finite temperature quantum memories, however. The tightness of the 4-dimensional bound sometimes feels like a complexity-theoretic barrier—almost as if nature allowing a $4 - \epsilon$ dimensional finite temperature quantum memory would cause some drastic and improbable collapse of a problem hierarchy. Nature taunts us with robust classical memories every time we stumble upon a magnet—where then, are its quantum cousins?

Whether we find one or not, the next decade will be increasingly more Quantum than those before it. The construction of a viable machine is now largely an engineering problem, and the inevitable hype of new technologies notwithstanding, quantum computers will be transformative. I can only look forward to seeing what will happen.

Bibliography

- [1] Tameem Albash et al. “Quantum adiabatic Markovian master equations”. In: *New Journal of Physics* 14.12 (2012), p. 123016.
- [2] R Alicki, M Fannes, and M Horodecki. “A statistical mechanics view on Kitaev’s proposal for quantum memories”. In: *Journal of Physics A: Mathematical and Theoretical* 40.24 (June 2007), pp. 6451–6467. ISSN: 1751-8113. DOI: 10.1088/1751-8113/40/24/012. URL: <http://stacks.iop.org/1751-8121/40/i=24/a=012?key=crossref.aa7b775d8a44d0614cf41de16a71cfe4>.
- [3] R Alicki, M Fannes, and M Horodecki. “On thermalization in Kitaev’s 2D model”. In: *Journal of Physics A: Mathematical and Theoretical* 42.6 (Feb. 2009), p. 065303. ISSN: 1751-8113. DOI: 10.1088/1751-8113/42/6/065303. URL: <http://stacks.iop.org/1751-8121/42/i=6/a=065303?key=crossref.a09e71df8e159a1b2d2a1c04a2ba29f5>.
- [4] R. Alicki et al. “On thermal stability of topological qubit in Kitaev’s 4D model”. In: (Nov. 2008). arXiv: 0811.0033. URL: <http://arxiv.org/abs/0811.0033>.
- [5] Dave Bacon. “Operator quantum error-correcting subsystems for self-correcting quantum memories”. In: *Physical Review A* 73.1 (2006), p. 012340. ISSN: 1050-2947. DOI: 10.1103/PhysRevA.73.012340. URL: <http://link.aps.org/doi/10.1103/PhysRevA.73.012340>.
- [6] Charles-Edouard Bardyn and Torsten Karzig. “Exponential Lifetime Improvement in Topological Quantum Memories”. In: (2015). arXiv: 1512.04528. URL: <http://arxiv.org/abs/1512.0452>.
- [7] William Barford. *Electronic and optical properties of conjugated polymers*. Vol. 159. Oxford University Press, 2013.
- [8] Hector Bombin and Miguel Angel Martin-Delgado. “Topological quantum distillation”. In: *Physical review letters* 97.18 (2006), p. 180501.
- [9] H Bombin et al. “Self-correcting quantum computers”. In: *New Journal of Physics* 15.5 (May 2013), p. 055023. ISSN: 1367-2630. DOI: 10.1088/1367-2630/15/5/055023. URL: <http://stacks.iop.org/1367-2630/15/i=5/a=055023?key=crossref.0aa2e96fdc4b8ba0d8e5a81f31362948>.
- [10] H Bombin et al. “Strong Resilience of Topological Codes to Depolarization”. In: *Physical Review X* 021004 (2012), pp. 1–10. DOI: 10.1103/PhysRevX.2.021004Subject.

- [11] Alfred B Bortz, Malvin H Kalos, and Joel L Lebowitz. “A new algorithm for Monte Carlo simulation of Ising spin systems”. In: *Journal of Computational Physics* 17.1 (1975), pp. 10–18.
- [12] Carlos Alexandre Brasil, Felipe Fernandes Fanchini, and Reginaldo de Jesus Napolitano. “A simple derivation of the Lindblad equation”. In: *Revista Brasileira de Ensino de Física* 35.1 (2013), pp. 01–09.
- [13] S. B. Bravyi and A. Yu. Kitaev. “Quantum codes on a lattice with boundary”. In: (Nov. 1998). arXiv: 9811052 [quant-ph]. URL: <http://arxiv.org/abs/quant-ph/9811052>.
- [14] Sergey Bravyi and Jeongwan Haah. “Quantum Self-Correction in the 3D Cubic Code Model”. In: *Physical Review Letters* 111.20 (Nov. 2013), p. 200501. ISSN: 0031-9007. DOI: 10.1103/PhysRevLett.111.200501. URL: <http://link.aps.org/doi/10.1103/PhysRevLett.111.200501>.
- [15] Sergey Bravyi and Matthew B. Hastings. “A Short Proof of Stability of Topological Order under Local Perturbations”. In: *Communications in Mathematical Physics* 307.3 (Sept. 2011), pp. 609–627. ISSN: 0010-3616. DOI: 10.1007/s00220-011-1346-2. URL: <http://link.springer.com/10.1007/s00220-011-1346-2>.
- [16] Sergey Bravyi, Matthew B. Hastings, and Spyridon Michalakis. “Topological quantum order: Stability under local perturbations”. In: *Journal of Mathematical Physics* 51.9 (2010), p. 093512. ISSN: 00222488. DOI: 10.1063/1.3490195. URL: <http://scitation.aip.org/content/aip/journal/jmp/51/9/10.1063/1.3490195>.
- [17] Sergey Bravyi, Bernhard Leemhuis, and Barbara M. Terhal. “Topological order in an exactly solvable 3D spin model”. In: *Annals of Physics* 326.4 (Apr. 2011), pp. 839–866. ISSN: 00034916. DOI: 10.1016/j.aop.2010.11.002. URL: <http://linkinghub.elsevier.com/retrieve/pii/S0003491610001910>.
- [18] Sergey Bravyi and Barbara Terhal. “A no-go theorem for a two-dimensional self-correcting quantum memory based on stabilizer codes”. In: *New Journal of Physics* 11.4 (Apr. 2009), p. 043029. ISSN: 1367-2630. DOI: 10.1088/1367-2630/11/4/043029. URL: <http://stacks.iop.org/1367-2630/11/i=4/a=043029?key=crossref.b6a8ac229f295aa66876f2899aa89f38>.
- [19] Courtney G Brell. “A proposal for self-correcting stabilizer quantum memories in 3 dimensions (or slightly less)”. In: *New Journal of Physics* 18.1 (2016), p. 013050.
- [20] Nikolas P Breuckmann et al. “Local decoders for the 2D and 4D toric code”. In: *arXiv preprint arXiv:1609.00510* (2016).
- [21] Heinz-Peter Breuer, Francesco Petruccione, et al. *The theory of open quantum systems*. Oxford University Press on Demand, 2002.
- [22] Benjamin J Brown et al. “Quantum Memories at Finite Temperature”. In: (2014), pp. 1–48. arXiv: arXiv:1411.6643v1.

- [23] Claudio Castelnovo and Claudio Chamon. “Entanglement and topological entropy of the toric code at finite temperature”. In: *Physical Review B* 76.18 (2007), p. 184442. ISSN: 1098-0121. DOI: 10.1103/PhysRevB.76.184442. URL: <http://link.aps.org/doi/10.1103/PhysRevB.76.184442>.
- [24] Claudio Castelnovo and Claudio Chamon. “Topological order in a three-dimensional toric code at finite temperature”. In: *Physical Review B* 78.15 (Oct. 2008), p. 155120. ISSN: 1098-0121. DOI: 10.1103/PhysRevB.78.155120. URL: <http://link.aps.org/doi/10.1103/PhysRevB.78.155120>.
- [25] Claudio Castelnovo and Claudio Chamon. “Topological quantum glassiness”. In: *Philos. Mag.* 92.1-3 (Jan. 2012), pp. 304–323. ISSN: 1478-6435. DOI: 10.1080/14786435.2011.609152. URL: <http://www.tandfonline.com/doi/abs/10.1080/14786435.2011.609152>.
- [26] Claudio Chamon. “Quantum Glassiness in Strongly Correlated Clean Systems: An Example of Topological Overprotection”. In: *Physical Review Letters* 94.4 (Jan. 2005), p. 040402. ISSN: 0031-9007. DOI: 10.1103/PhysRevLett.94.040402. URL: <http://link.aps.org/doi/10.1103/PhysRevLett.94.040402>.
- [27] Garnet Kin-Lic Chan et al. “Matrix product operators, matrix product states, and ab initio density matrix renormalization group algorithms”. In: *The Journal of chemical physics* 145.1 (2016), p. 014102.
- [28] Herman Chernoff. “A career in statistics”. In: *Past, Present, and Future of Statistical Science* (2014), p. 29.
- [29] Stefano Chesi, Beat Röthlisberger, and Daniel Loss. “Self-correcting quantum memory in a thermal environment”. In: *Physical Review A* 82.2 (Aug. 2010), p. 022305. ISSN: 1050-2947. DOI: 10.1103/PhysRevA.82.022305. URL: <http://link.aps.org/doi/10.1103/PhysRevA.82.022305>.
- [30] Stefano Chesi et al. “Thermodynamic stability criteria for a quantum memory based on stabilizer and subsystem codes”. In: *New Journal of Physics* 12.2 (Feb. 2010), p. 025013. ISSN: 1367-2630. DOI: 10.1088/1367-2630/12/2/025013. URL: <http://stacks.iop.org/1367-2630/12/i=2/a=025013?key=crossref.a5c828ea669d8c1c442934ce8013>
- [31] J. Chiaverini et al. “Realization of quantum error correction”. In: *Nature* 432.7017 (2004), pp. 602–605. ISSN: 0028-0836. DOI: 10.1038/nature03074. URL: <http://www.nature.com/doifinder/10.1038/nature03074>.
- [32] A.D. Córcoles et al. “Demonstration of a quantum error detection code using a square lattice of four superconducting qubits”. In: *Nature Communications* 6 (2015), p. 6979. ISSN: 2041-1723. DOI: 10.1038/ncomms7979. arXiv: 1410.6419. URL: <http://www.nature.com/doifinder/10.1038/ncomms7979>.
- [33] Yann N Dauphin et al. “Identifying and attacking the saddle point problem in high-dimensional non-convex optimization”. In: *Advances in neural information processing systems*. 2014, pp. 2933–2941.

- [34] John Dengis, Robert König, and Fernando Pastawski. “An optimal dissipative encoder for the toric code”. In: *New Journal of Physics* 16.1 (Jan. 2014), p. 013023. ISSN: 1367-2630. DOI: 10.1088/1367-2630/16/1/013023. URL: <http://stacks.iop.org/1367-2630/16/i=1/a=013023?key=crossref.a495ab2d26cd17f3c383c186d8b3bb05>.
- [35] Eric Dennis et al. “Topological quantum memory”. In: *Journal of Mathematical Physics* 43.9 (2002), p. 4452. ISSN: 00222488. DOI: 10.1063/1.1499754. URL: http://jmp.aip.org/resource/1/jmapaq/v43/i9/p4452%5C_s1.
- [36] Guillaume Duclos-Cianci and David Poulin. “Fast Decoders for Topological Quantum Codes”. In: *Physical Review Letters* 104.5 (Feb. 2010), p. 050504. ISSN: 0031-9007. DOI: 10.1103/PhysRevLett.104.050504. URL: <http://link.aps.org/doi/10.1103/PhysRevLett.104.050504>.
- [37] Sébastien Dusuel et al. “Robustness of a Perturbed Topological Phase”. In: *Physical Review Letters* 106.10 (Mar. 2011), p. 107203. ISSN: 0031-9007. DOI: 10.1103/PhysRevLett.106.107203. URL: <http://link.aps.org/doi/10.1103/PhysRevLett.106.107203>.
- [38] Jack Edmonds. “Paths, trees, and flowers”. In: *Canadian Journal of mathematics* 17.3 (1965), pp. 449–467.
- [39] G Evenbly and G Vidal. “Entanglement renormalization in noninteracting fermionic systems”. In: *Physical Review B* 81.23 (2010), p. 235102.
- [40] Glen Evenbly and Guifr Vidal. “Frustrated antiferromagnets with entanglement renormalization: Ground state of the spin-1 2 heisenberg model on a kagome lattice”. In: *Physical review letters* 104.18 (2010), p. 187203.
- [41] Glen Evenbly and Guifre Vidal. “Class of highly entangled many-body states that can be efficiently simulated”. In: *Physical review letters* 112.24 (2014), p. 240502.
- [42] Glen Evenbly and Guifre Vidal. “Quantum criticality with the multi-scale entanglement renormalization ansatz”. In: *Strongly correlated systems*. Springer, 2013, pp. 99–130.
- [43] Glen Evenbly and Guifre Vidal. “Tensor network renormalization”. In: *Physical review letters* 115.18 (2015), p. 180405.
- [44] Glen Evenbly and Guifre Vidal. “Tensor network renormalization yields the multi-scale entanglement renormalization ansatz”. In: *Physical review letters* 115.20 (2015), p. 200401.
- [45] Glen Evenbly and Guifré Vidal. “Algorithms for entanglement renormalization: boundaries, impurities and interfaces”. In: *Journal of Statistical Physics* 157.4-5 (2014), pp. 931–978.
- [46] Glen Evenbly and Guifré Vidal. “Entanglement renormalization in free bosonic systems: real-space versus momentum-space renormalization group transforms”. In: *New Journal of Physics* 12.2 (2010), p. 025007.

- [47] Glen Evenbly and Guifré Vidal. “Tensor network states and geometry”. In: *Journal of Statistical Physics* 145.4 (2011), pp. 891–918.
- [48] Andrew J Ferris. “Unbiased Monte Carlo for the age of tensor networks”. In: *arXiv preprint arXiv:1507.00767* (2015).
- [49] Austin G. Fowler. “Proof of Finite Surface Code Threshold for Matching”. In: *Physical Review Letters* 109.18 (Nov. 2012), p. 180502. ISSN: 0031-9007. DOI: 10.1103/PhysRevLett.109.180502. URL: <http://link.aps.org/doi/10.1103/PhysRevLett.109.180502>.
- [50] Austin G. Fowler, Ashley M. Stephens, and Peter Groszkowski. “High-threshold universal quantum computation on the surface code”. In: *Physical Review A* 80.5 (Nov. 2009), p. 052312. ISSN: 1050-2947. DOI: 10.1103/PhysRevA.80.052312. URL: <http://link.aps.org/doi/10.1103/PhysRevA.80.052312>.
- [51] Michael H. Freedman. “Quantum Computation and the Localization of Modular Functors”. In: *Foundations of Computational Mathematics* 1.2 (June 2001), pp. 183–204. ISSN: 1615-3375. DOI: 10.1007/s102080010006. URL: <http://www.springerlink.com/openurl.asp?genre=article%5C&id=doi:10.1007/s102080010006>.
- [52] Michael H. Freedman, Michael Larsen, and Zhenghan Wang. “A Modular Functor Which is Universal for Quantum Computation”. In: *Communications in Mathematical Physics* 227.3 (June 2002), pp. 605–622. ISSN: 0010-3616. DOI: 10.1007/s002200200645. URL: <http://www.springerlink.com/openurl.asp?genre=article%5C&id=doi:10.1007/s002200200645>.
- [53] C Daniel Freeman and Joan Bruna. “Topology and geometry of half-rectified network optimization”. In: *arXiv preprint arXiv:1611.01540* (2016).
- [54] C. Daniel Freeman, C. M. Herdman, and K. B. Whaley. “Engineering autonomous error correction in stabilizer codes at finite temperature”. In: *Phys. Rev. A* 96 (1 July 2017), p. 012311. DOI: 10.1103/PhysRevA.96.012311. URL: <https://link.aps.org/doi/10.1103/PhysRevA.96.012311>.
- [55] C. Daniel Freeman et al. “Relaxation dynamics of the toric code in contact with a thermal reservoir: Finite-size scaling in a low-temperature regime”. In: *Physical Review B* 90.13 (Oct. 2014), p. 134302. ISSN: 1098-0121. DOI: 10.1103/PhysRevB.90.134302. URL: <http://link.aps.org/doi/10.1103/PhysRevB.90.134302>.
- [56] C Daniel Freeman et al. “Stable quantum memories with limited measurement”. In: *arXiv preprint arXiv:1708.02260* (2017).
- [57] Keisuke Fujii et al. “Measurement-Free Topological Protection Using Dissipative Feedback”. In: *Physical Review X* 4.4 (Dec. 2014), p. 041039. ISSN: 2160-3308. DOI: 10.1103/PhysRevX.4.041039. URL: <http://link.aps.org/doi/10.1103/PhysRevX.4.041039>.

- [58] Roy J. Glauber. “Time-Dependent Statistics of the Ising Model”. In: *Journal of Mathematical Physics* 4.2 (1963), p. 294. ISSN: 00222488. DOI: 10.1063/1.1703954. URL: <http://link.aip.org/link/JMAPAQ/v4/i2/p294/s1%5C&Agg=doi>.
- [59] Daniel Gottesman. “? 1 0 0”. In: *Physical Review A* 57.1 (1998), pp. 127–137.
- [60] Jeongwan Haah. “Local stabilizer codes in three dimensions without string logical operators”. In: *Physical Review A* 83.4 (Apr. 2011), p. 042330. ISSN: 1050-2947. DOI: 10.1103/PhysRevA.83.042330. URL: <http://link.aps.org/doi/10.1103/PhysRevA.83.042330>.
- [61] Alioscia Hamma, Claudio Castelnovo, and Claudio Chamon. “Toric-boson model : Toward a topological quantum memory at finite temperature”. In: *Physical Review B* (2009), pp. 1–6. DOI: 10.1103/PhysRevB.79.245122.
- [62] James William Harrington. “Analysis of quantum error-correcting codes : symplectic lattice codes and toric codes Thesis by”. In: *Thesis* (2004).
- [63] Matthew Hastings. “Topological Order at Nonzero Temperature”. In: *Physical Review Letters* 107.21 (Nov. 2011), pp. 1–5. ISSN: 0031-9007. DOI: 10.1103/PhysRevLett.107.210501. URL: <http://link.aps.org/doi/10.1103/PhysRevLett.107.210501>.
- [64] Matthew B. Hastings, Grant H. Watson, and Roger G. Melko. “Self-Correcting Quantum Memories Beyond the Percolation Threshold”. In: *Physical Review Letters* 112.7 (Feb. 2014), p. 070501. ISSN: 0031-9007. DOI: 10.1103/PhysRevLett.112.070501. URL: <http://link.aps.org/doi/10.1103/PhysRevLett.112.070501>.
- [65] C. M. Herdman et al. “Stroboscopic Generation of Topological Protection”. In: *Physical Review Letters* 104.23 (June 2010), p. 230501. ISSN: 0031-9007. DOI: 10.1103/PhysRevLett.104.230501. URL: <http://link.aps.org/doi/10.1103/PhysRevLett.104.230501>.
- [66] Michael Herold et al. “Cellular-automaton decoders for topological quantum memories”. In: *Nature Publishing Group* April 2014 (2015). DOI: 10.1038/npjqi.2015.10. URL: <http://dx.doi.org/10.1038/npjqi.2015.10>.
- [67] William Huggins et al. “Monte Carlo Tensor Network Renormalization”. In: *arXiv preprint arXiv:1710.03757* (2017).
- [68] Adrian Hutter et al. “Effective quantum-memory Hamiltonian from local two-body interactions”. In: *Physical Review A* 012321 (2014), pp. 1–12. DOI: 10.1103/PhysRevA.90.012321.
- [69] Adrian Hutter et al. “Self-correcting quantum memory with a boundary”. In: *Physical Review A* 86.5 (Nov. 2012), p. 052340. ISSN: 1050-2947. DOI: 10.1103/PhysRevA.86.052340. URL: <http://link.aps.org/doi/10.1103/PhysRevA.86.052340>.
- [70] Adrian Hutter et al. “Self-correcting quantum memory with a boundary”. In: *Phys. Rev. A* 86 (5 Nov. 2012), p. 052340. DOI: 10.1103/PhysRevA.86.052340. URL: <http://link.aps.org/doi/10.1103/PhysRevA.86.052340>.

- [71] S. Iblisdir et al. “Thermal states of anyonic systems”. In: *Nuclear Physics B* 829.3 (2010), pp. 401–424. ISSN: 05503213. DOI: 10.1016/j.nuclphysb.2009.11.009. URL: <http://linkinghub.elsevier.com/retrieve/pii/S055032130900604X>.
- [72] P. Jouzdani, E. Novais, and E. R. Mucciolo. “Fidelity of the surface code in the presence of a bosonic bath”. In: *Physical Review A* 88.1 (July 2013), p. 012336. ISSN: 1050-2947. DOI: 10.1103/PhysRevA.88.012336. URL: <http://link.aps.org/doi/10.1103/PhysRevA.88.012336>.
- [73] Eliot Kapit, John T Chalker, and Steven H Simon. “Passive correction of quantum logical errors in a driven, dissipative system: A blueprint for an analog quantum code fabric”. In: *Phys. Rev. A* 91.6 (June 2015), p. 62324. DOI: 10.1103/PhysRevA.91.062324. URL: <http://link.aps.org/doi/10.1103/PhysRevA.91.062324>.
- [74] Alastair Kay. “Capabilities of a Perturbed Toric Code as a Quantum Memory”. In: *Physical Review Letters* 107.27 (Dec. 2011), 270502 (2011). ISSN: 0031-9007. DOI: 10.1103/PhysRevLett.107.270502. URL: <http://link.aps.org/doi/10.1103/PhysRevLett.107.270502>.
- [75] Gerhard Keller. *Equilibrium states in ergodic theory*. Vol. 42. Cambridge university press, 1998.
- [76] Alexei Kitaev. “Fault-tolerant quantum computation by anyons”. In: *Annals of Physics* 303.1 (Jan. 2003), pp. 2–30. ISSN: 00034916. DOI: 10.1016/S0003-4916(02)00018-0. URL: <http://linkinghub.elsevier.com/retrieve/pii/S0003491602000180>.
- [77] Israel Klich. “On the stability of topological phases on a lattice”. In: *Annals of Physics* 325.10 (Oct. 2010), pp. 2120–2131. ISSN: 00034916. DOI: 10.1016/j.aop.2010.05.002. URL: <http://linkinghub.elsevier.com/retrieve/pii/S000349161000093X>.
- [78] E. Knill et al. “Benchmarking Quantum Computers: The Five-Qubit Error Correcting Code”. In: *Phys. Rev. Lett.* 86 (25 June 2001), pp. 5811–5814. DOI: 10.1103/PhysRevLett.86.5811. URL: <https://link.aps.org/doi/10.1103/PhysRevLett.86.5811>.
- [79] Andrzej Kossakowski et al. “Quantum detailed balance and KMS condition”. In: *Communications in Mathematical Physics* 57.2 (1977), pp. 97–110.
- [80] Alexey A Kovalev and Leonid P Pryadko. “Fault tolerance of quantum low-density parity check codes with sublinear distance scaling”. In: *Physical Review A* 020304 (2013), pp. 1–5. DOI: 10.1103/PhysRevA.87.020304.
- [81] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. “Imagenet classification with deep convolutional neural networks”. In: *Advances in neural information processing systems*. 2012, pp. 1097–1105.
- [82] Olivier Landon-Cardinal and David Poulin. “Local Topological Order Inhibits Thermal Stability in 2D”. In: *Physical Review Letters* 110.9 (Feb. 2013), p. 090502. ISSN: 0031-9007. DOI: 10.1103/PhysRevLett.110.090502. URL: <http://link.aps.org/doi/10.1103/PhysRevLett.110.090502>.

- [83] Olivier Landon-Cardinal et al. “Perturbative instability of quantum memory based on effective long-range interactions”. In: *Physical Review A* 91.3 (2015), p. 032303.
- [84] José Ignacio Latorre, Enrique Rico, and Guifré Vidal. “Ground state entanglement in quantum spin chains”. In: *arXiv preprint quant-ph/0304098* (2003).
- [85] Jimmie Lawson. *Matrix Lie Groups and Control Theory*. [Online; accessed 12-June-2018]. 2007. URL: %5Curl%7Bhttps://www.math.lsu.edu/~lawson/liecontrol.pdf%7D.
- [86] Daniel A Lidar and Todd A Brun. *Quantum error correction*. Cambridge University Press, 2013.
- [87] Daniel Lidar and Todd Brun. *Quantum Error Correction*. 1st ed. Cambridge University Press, 2013.
- [88] Raghu Mahajan et al. “Entanglement structure of non-equilibrium steady states”. In: *arXiv preprint arXiv:1608.05074* (2016).
- [89] Dalimil Mazáč and Alioscia Hamma. “Topological order, entanglement, and quantum memory at finite temperature”. In: *Annals of Physics* 327.9 (Sept. 2012), pp. 2096–2111. ISSN: 00034916. DOI: 10.1016/j.aop.2012.05.004. URL: <http://linkinghub.elsevier.com/retrieve/pii/S0003491612000723>.
- [90] Spyridon Michalakis and Justyna P. Zwolak. “Stability of Frustration-Free Hamiltonians”. In: *Communications in Mathematical Physics* 322.2 (July 2013), pp. 277–302. ISSN: 0010-3616. DOI: 10.1007/s00220-013-1762-6. URL: <http://link.springer.com/10.1007/s00220-013-1762-6>.
- [91] Kamil Michnicki. “3-d quantum stabilizer codes with a power law energy barrier”. In: *arXiv preprint arXiv:1208.3496* (2012).
- [92] Carlos Mochon. “Anyon computers with smaller groups”. In: *Physical Review A* 69.3 (2004), p. 032306. ISSN: 1050-2947. DOI: 10.1103/PhysRevA.69.032306. URL: <http://link.aps.org/doi/10.1103/PhysRevA.69.032306>.
- [93] Carlos Mochon. “Anyons from nonsolvable finite groups are sufficient for universal quantum computation”. In: *Physical Review A* 67.2 (2003), p. 022315. ISSN: 1050-2947. DOI: 10.1103/PhysRevA.67.022315. URL: <http://link.aps.org/doi/10.1103/PhysRevA.67.022315>.
- [94] F Montalenti and R Ferrando. “Probability of dimer reassociation in two dimensions”. In: *Physical Review E* 61.4 (Apr. 2000), pp. 3411–3416. ISSN: 1063-651X. DOI: 10.1103/PhysRevE.61.3411. URL: <http://link.aps.org/doi/10.1103/PhysRevE.61.3411>.
- [95] Chetan Nayak et al. “Non-Abelian anyons and topological quantum computation”. In: *Reviews of Modern Physics* 80.3 (Sept. 2008), pp. 1083–1159. ISSN: 0034-6861. DOI: 10.1103/RevModPhys.80.1083. URL: <http://link.aps.org/doi/10.1103/RevModPhys.80.1083>.

- [96] Naomi Nickerson. “Practical fault-tolerant quantum computing”. In: (2015).
- [97] Naomi H. Nickerson. “Error correcting power of small topological codes”. In: (2016), pp. 12–10. arXiv: [arXiv:1609.01753](https://arxiv.org/abs/1609.01753).
- [98] Michael A Nielsen and Isaac Chuang. *Quantum computation and quantum information*. 2002.
- [99] D Nigg et al. “Quantum computations on a topologically encoded qubit”. In: *Science* 345.6194 (2014), pp. 285–287.
- [100] E Novais and Eduardo R Mucciolo. “Surface Code Threshold in the Presence of Correlated Errors”. In: 010502. January (2013), pp. 1–5. DOI: [10.1103/PhysRevLett.110.010502](https://doi.org/10.1103/PhysRevLett.110.010502).
- [101] Zohar Nussinov and Gerardo Ortiz. “A symmetry principle for topological quantum order”. In: *Annals of Physics* 324.5 (2009), pp. 977–1057. ISSN: 00034916. DOI: [10.1016/j.aop.2008.11.002](https://doi.org/10.1016/j.aop.2008.11.002). URL: <http://linkinghub.elsevier.com/retrieve/pii/S0003491608001711>.
- [102] Zohar Nussinov and Gerardo Ortiz. “Autocorrelations and thermal fragility of anyonic loops in topologically quantum ordered systems”. In: *Physical Review B* 77.6 (2008), p. 064302. ISSN: 1098-0121. DOI: [10.1103/PhysRevB.77.064302](https://doi.org/10.1103/PhysRevB.77.064302). URL: <http://link.aps.org/doi/10.1103/PhysRevB.77.064302>.
- [103] Fernando Pastawski et al. “Holographic quantum error-correcting codes: Toy models for the bulk/boundary correspondence”. In: *Journal of High Energy Physics* 2015.6 (2015), p. 149.
- [104] Philip Pearle. “Simple derivation of the Lindblad equation”. In: *European Journal of Physics* 33.4 (2012), p. 805.
- [105] Fabio L Pedrocchi, Stefano Chesi, and Daniel Loss. “Quantum memory coupled to cavity modes”. In: *Physical Review B* 115415. January (2011), pp. 1–15. DOI: [10.1103/PhysRevB.83.115415](https://doi.org/10.1103/PhysRevB.83.115415).
- [106] Fabio L. Pedrocchi et al. “Enhanced thermal stability of the toric code through coupling to a bosonic bath”. In: *Physical Review A* 88.6 (Dec. 2013), p. 062313. ISSN: 1050-2947. DOI: [10.1103/PhysRevA.88.062313](https://doi.org/10.1103/PhysRevA.88.062313). URL: <http://link.aps.org/doi/10.1103/PhysRevA.88.062313>.
- [107] John Preskill. “Quantum Error Correction Course Notes”. In: (Apr. 2018). URL: <http://www.theory.caltech.edu/~preskill/ph219/index.html>.
- [108] Sam Roberts and Stephen D Bartlett. “Symmetry-protected self-correcting quantum memories”. In: *arXiv preprint arXiv:1805.01474* (2018).
- [109] Beat Röthlisberger et al. “Incoherent dynamics in the toric code subject to disorder”. In: *Physical Review A* 85.2 (Feb. 2012), p. 022313. ISSN: 1050-2947. DOI: [10.1103/PhysRevA.85.022313](https://doi.org/10.1103/PhysRevA.85.022313). URL: <http://link.aps.org/doi/10.1103/PhysRevA.85.022313>.

- [110] Richard Serfozo. *Basics of applied stochastic processes*. Springer Science & Business Media, 2009.
- [111] Abbas Al-Shimary, James R Wootton, and Jiannis K Pachos. “Lifetime of topological quantum memories in thermal environment”. In: *New Journal of Physics* 15.2 (Feb. 2013), p. 025027. ISSN: 1367-2630. DOI: 10.1088/1367-2630/15/2/025027. URL: <http://stacks.iop.org/1367-2630/15/i=2/a=025027?key=crossref.e5acfd3cbb9c40c2282c51a81dedd33c>.
- [112] Karthik Siva and Beni Yoshida. “Topological order and memory time in marginally-self-correcting quantum memory”. In: *Physical Review A* 95.3 (2017), p. 032324.
- [113] Ashley M Stephens. “Fault-tolerant thresholds for quantum error correction with the surface code”. In: *Physical Review A* 89.2 (2014), p. 022321.
- [114] Koujin Takeda and Hidetoshi Nishimori. “Nishimori Line Construction”. In: (2003), pp. 1–23. arXiv: 0310279v2 [arXiv:hep-th].
- [115] Maika Takita et al. “Experimental demonstration of fault-tolerant state preparation with superconducting qubits”. In: (), pp. 1–11. arXiv: 1705.09259. URL: <https://arxiv.org/pdf/1705.09259.pdf>.
- [116] Kristan Temme. “Thermalization time bounds for”. In: (2014). arXiv: arXiv:1412.2858v1.
- [117] Kristan Temme and Michael J Kastoryano. “How fast do stabilizer Hamiltonians thermalize?” In: *arXiv preprint arXiv:1505.07811* (2015).
- [118] Barbara M. Terhal. “Quantum error correction for quantum memories”. In: *Rev. Mod. Phys.* 87 (2 Apr. 2015), pp. 307–346. DOI: 10.1103/RevModPhys.87.307. URL: <https://link.aps.org/doi/10.1103/RevModPhys.87.307>.
- [119] Giacomo Torlai and Roger G Melko. “A Neural Decoder for Topological Codes”. In: (2016), pp. 1–6. arXiv: arXiv:1610.0423.
- [120] Simon Trebst et al. “Breakdown of a Topological Phase: Quantum Phase Transition in a Loop Gas Model with Tension”. In: *Physical Review Letters* 98.7 (Feb. 2007), p. 070602. ISSN: 0031-9007. DOI: 10.1103/PhysRevLett.98.070602. URL: <http://link.aps.org/doi/10.1103/PhysRevLett.98.070602>.
- [121] Norm M Tubman et al. “A deterministic alternative to the full configuration interaction quantum Monte Carlo method”. In: *The Journal of chemical physics* 145.4 (2016), p. 044112.
- [122] I. S. Tupitsyn et al. “Topological multicritical point in the phase diagram of the toric code model and three-dimensional lattice gauge Higgs model”. In: *Physical Review B* 82.8 (Aug. 2010), p. 085114. ISSN: 1098-0121. DOI: 10.1103/PhysRevB.82.085114. URL: <http://link.aps.org/doi/10.1103/PhysRevB.82.085114>.
- [123] Guifré Vidal. “Class of quantum many-body states that can be efficiently simulated”. In: *Physical review letters* 101.11 (2008), p. 110501.

- [124] O Viyuela, a Rivas, and M a Martin-Delgado. “Generalized toric codes coupled to thermal baths”. In: *New Journal of Physics* 14.3 (Mar. 2012), p. 033044. ISSN: 1367-2630. DOI: 10.1088/1367-2630/14/3/033044. URL: <http://stacks.iop.org/1367-2630/14/i=3/a=033044?key=crossref.1d624fa36a697052ea672c14000b6891>.
- [125] David S Wang et al. “Threshold error rates for the toric and surface codes”. In: *arXiv preprint arXiv:0905.0531* (2009).
- [126] Fern H E Watson and Sean D Barrett. “Logical error rate scaling of the toric code”. In: *New Journal of Physics* 16.9 (Sept. 2014), p. 093045. ISSN: 1367-2630. DOI: 10.1088/1367-2630/16/9/093045. URL: <http://stacks.iop.org/1367-2630/16/i=9/a=093045?key=crossref.3ccb7f83da86446ad598a2898aae4ff1>.
- [127] Dave Wecker et al. “Gate-count estimates for performing quantum chemistry on small quantum computers”. In: *Physical Review A* 90.2 (Aug. 2014), p. 022305. ISSN: 1050-2947. DOI: 10.1103/PhysRevA.90.022305. URL: <http://link.aps.org/doi/10.1103/PhysRevA.90.022305>.
- [128] Hendrik Weimer et al. “A Rydberg quantum simulator”. In: *Nature Physics* 6.5 (Mar. 2010), pp. 382–388. ISSN: 1745-2473. DOI: 10.1038/nphys1614. URL: <http://www.nature.com/doifinder/10.1038/nphys1614>.
- [129] X.G. Wen. “Topological Orders in Rigid States”. In: *Int. J. Mod. Phys. B* 4.2 (1990), p. 239. DOI: 10.1142/S0217979290000139. URL: <http://www.worldscinet.com/abstract?id=pii:S0217979290000139>.
- [130] Xiao-Gang Wen. “Quantum orders in an exact soluble model”. In: *Physical review letters* 90.1 (2003), p. 016803.
- [131] SR White, RM Noack, and DJ Scalapino. “Resonating valence bond theory of coupled Heisenberg chains”. In: *Physical review letters* 73.6 (1994), p. 886.
- [132] Steven R White. “Density matrix formulation for quantum renormalization groups”. In: *Physical review letters* 69.19 (1992), p. 2863.
- [133] Wikipedia. *Markov Chain*. [Online; accessed 1-June-2018]. 20018. URL: https://en.wikipedia.org/wiki/Markov_chain%7D.
- [134] James Wootton. “A simple decoder for topological codes”. In: *Entropy* 17.4 (2015), pp. 1946–1957.
- [135] James R. Wootton and Daniel Loss. “High Threshold Error Correction for the Surface Code”. In: *Physical Review Letters* 109.16 (Oct. 2012), p. 160503. ISSN: 0031-9007. DOI: 10.1103/PhysRevLett.109.160503. URL: <http://link.aps.org/doi/10.1103/PhysRevLett.109.160503>.
- [136] Beni Yoshida. “Feasibility of self-correcting quantum memory and thermal stability of topological order”. In: *Annals of Physics* 326.10 (Oct. 2011), pp. 2566–2633. ISSN: 00034916. DOI: 10.1016/j.aop.2011.06.001. URL: <http://linkinghub.elsevier.com/retrieve/pii/S0003491611001023>.

- [137] Kevin C Young et al. “Finite temperature quantum simulation of stabilizer Hamiltonians”. In: *Journal of Physics B: Atomic, Molecular and Optical Physics* 45.15 (Aug. 2012), p. 154012. ISSN: 0953-4075. DOI: 10.1088/0953-4075/45/15/154012. URL: <http://stacks.iop.org/0953-4075/45/i=15/a=154012?key=crossref.edee0d03725a793d3017e2ac56d2b04d>.
- [138] WM Young and EW Elcock. “Monte Carlo studies of vacancy migration in binary ordered alloys: I”. In: *Proceedings of the Physical Society* 89.3 (1966), p. 735.
- [139] Cristian Zanoci and Brian G Swingle. “Entanglement and thermalization in open fermion systems”. In: *arXiv preprint arXiv:1612.04840* (2016).
- [140] Long Zhang, Su-Peng Kou, and Youjin Deng. “Quench dynamics of the topological quantum phase transition in the Wen-plaquette model”. In: *Physical Review A* 83.6 (2011), p. 062113.