Controlling Quantum Information

Thesis by
Andrew J. Landahl

In Partial Fulfillment of the Requirements
for the Degree of
Doctor of Philosophy

California Institute of Technology
Pasadena, California

2002

(Defended May 21, 2002)

Dedicated to my parents

Nancy and John Landahl

and to the memory of my grandparents

Barbara and Aldo Martin

Acknowledgements

I am pleased to acknowledge all of the people who have contributed to my personal and professional development during my graduate career. My most heartfelt gratitude goes to my friends and family, especially to my parents Nancy and John, to my sister Heidi, and to my aunt Carol for their collective unwavering love and support.

I am also especially grateful to my mentor, colleague, and friend John Preskill. John provided me with research suggestions when I first began my studies, the opportunity to travel and establish scientific collaborations, the freedom to pursue my own intellectual interests, and the steadfast encouragement to help me see my research through. John's organization of thought and clarity of expression continue to be deep personal inspirations. I will be fortunate indeed if even a fraction of those qualities have rubbed off on me through my years of study in his company at Caltech.

This thesis would not have been possible were it not from the long list of collaborators from whom I have learned so much: Charlene Ahn, Andrew Childs, Eric Dennis, Enrico Deotto, Andrew Doherty, Eddie Farhi, Sam Gutmann, Jeffrey Goldstone, Alexei Kitaev, and John Preskill. Their insights have been invaluable.

I also thank Jeff Kimble, Hideo Mabuchi, and Kip Thorne who graciously gave of their time to sit on both my candidacy and thesis defense committees. I consider them additional mentors, from whom I have learned greatly, both through formal instruction and informal discussions.

A special thanks goes to Ike Chuang and his quanta research group in the Media Lab at MIT for their hospitality during a fruitful visit which both initiated the research described in Chapter 5 and which has generated an ongoing research collaboration.

One of the wonderful things about my Caltech research experience has been the vast number of quantum information scientists I have met and learned so much from, many of whom who have graciously come and visited Caltech, bringing with

them new ideas and fresh perspectives. I would like to thank the following people for lively and engaging scientific interactions: Scott Aaronsen, Dorit Aharonov, Srinivas Aji, Paul Alsing, Dave Bacon, Howard Barnum, Dave Beckman, Charlie Bennett, Herb Bernstein, Ken Brown, Carl Caves, Nicholas Cerf, Richard Cleve, John Cortese, Sumit Daftuar, Win van Dam, Keshav Dani, Eric Dennis, Ivan Deutsch, David DiVincenzo, Steven van Enk, Chris Fuchs, Daniel Gottesman, Bob Gingrich, Salman Habib, Sean Hallgren, Jim Harrington, Aram Harrow, Patrick Hayden, Peter Høyer, Lawrence Ip, Kurt Jacobs, Daniel Jonathan, Richard Jozsa, Julia Kempe, Rowan Killip, Manny Knill, Inna Kozinsky, Greg Kuperberg, Debbie Leung, Seth Lloyd, Hoi-Kwong Lo, Gerard Milburn, Paul McFadden, Carlos Monchon, Mike Mosca, Ashwin Nayak, Michael Nielsen, Tobias Osbourne, Ben Rahn, Eric Rains, Ben Recht, Mary-Beth Ruskai, Rüdiger Schack, Leonard Schulman, Yaoyun Shi, Ben Schumacher, Peter Shor, John Smolin, Federico Spedialari, Dan Stamper-Kurn, Andrew Steane, Dan Steck, Barbara Terhal, Ashish Thalpiyal, Ben Toner, Umesh Vazarani, Guifre Vidal, Lorenza Viola, Chenyang Wang, Mike Westmoreland, Birgitta Whaley, Clint White, Howard Wiseman, Dave Wineland, Ronald de Wolf, Bill Wootters, and Nathan Wozny.

I would also like to thank my many friends and colleagues in experimental quantum optics at Caltech who have taught me about the importance of keeping close to experiment in theoretical research. Never again will I begin a sentence with "Suppose you have three spins in a box." Thanks to: Mike Armen, John Au, Andy Berglund, Kevin Birnbaum, Andreea Boca, Glov Boi, Dave Boozer, Joe Buck, James Chou, Akira Furusawa, JM Geremia, Win Goh, Christina Hood, Alex Kuzmich, Ron Legere, Ben Lev, Peter Lodahl, Theresa Lynn, Jason McKeever, Christof Naegerl, Dominik Schraeder, John Stockton, David Vernooy, Jon Williams, and Jun Ye.

The administrative assistance I received from Beth Adams, Donna Driscoll, Ann Harvey, Sheri Stoll, and Helen Tuck has been phenomenal; I thank them for going the extra mile so many times. Without them I would have surely been lost in so many ways.

As this thesis signals the end of my formal education, but certainly not the end of a lifetime of learning, I would like to take this opportunity to thank some of the teachers I have had over the years who had major impacts on my intellectual development. In particular, I would like to thank Mrs. Elaine Romanias, my fifth- and sixth-grade teacher who encouraged me to explore problems from creative points-of-view, Mr. Michael Stueben, my geometry teacher who challenged me with the puzzles he developed for his Discover Magazine *Brain Bogglers* column, Mr. Don Hyatt, my computer science teacher who mentored me as I undertook a Westinghouse Science Talent Search project and opened my eyes to the exciting field of research science, Dr. John Dell, my AP physics teacher who gave so generously of his time, lent me many of his physics books, answered my seemingly endless list of physics and math questions, and by whose example I strengthened my resolve to become a physicist, Prof. Yoshi Oono, my Research Experience for Undergraduates mentor who taught me that fundamental physics isn't defined by energy scales, Prof. Lay-Nam Chang, my honors thesis advisor who encouraged me to pursue my research interests in quantum information science, Prof. Kip Thorne, who taught me how to appreciate physics independently from mathematics, and lastly Prof. John Preskill, my Ph.D. advisor who taught me by example that excellence in research and excellence in teaching are not mutually exclusive alternatives.

Since the dawn of science, generous patrons have enabled scientific progress. In my case, I would like to thank Caltech, DARPA, the NSF, and IBM for their financial support during my graduate career. Without their assistance, this thesis would surely not have been possible.

Finally, I would like to thank the following diversions for preserving my sanity during my graduate career: table tennis, fantasy baseball, swing dancing, the a-hats softball team, and Carl's Jr. hamburgers.

Controlling Quantum Information

by

Andrew J. Landahl

In Partial Fulfillment of the
Requirements for the Degree of
Doctor of Philosophy

Abstract

Quantum information science explores ways in which quantum physical laws can be harnessed to control the acquisition, transmission, protection, and processing of information. This field has seen explosive growth in the past several years from progress on both theoretical and experimental fronts. Essential to this endeavor are methods for controlling quantum information.

In this thesis, I present three new approaches for controlling quantum information. First, I present a new protocol for continuously protecting unknown quantum states from noise. This protocol combines and expands ideas from the theories of quantum error correction and quantum feedback control. The result can outperform either approach by itself. I generalize this protocol to all known quantum stabilizer codes, and study its application to the three-qubit repetition code in detail via Monte Carlo simulations.

Next, I present several new protocols for controlling quantum information that are fault-tolerant. These protocols require only local quantum processing due to the topological properties of the quantum error correcting codes upon which they are built. I show that each protocol's fault-dependence behavior exhibits an order-disorder phase transition when mapped onto an associated statistical-mechanical model. I review the critical error rates of these protocols found by numerical study of the associated models, and I present new analytic bounds for them using a self-avoiding random walk argument. Moreover, I discuss fault-tolerant procedures for

encoding, error-correction, computing, and decoding quantum information using these protocols, and calculate the accuracy threshold of fault-tolerant quantum memory for protocols using them.

I end by presenting a new class of quantum algorithms that solve combinatorial optimization problems solely by measurement. I compute the running times of these algorithms by establishing an explicit dynamical model for the measurement process. This model, the digitized version of von Neumann's measurement model, is recognized as Kitaev's phase estimation algorithm. I show that the running times of these algorithms are closely related to the running times of adiabatic quantum algorithms. Finally, I present a two-measurement algorithm that achieves a quadratic speedup for Grover's unstructured search problem.

Preface

In the spring of 1996, I visited Caltech as a prospective graduate student eager to pursue research in the nascent field of quantum computing. I was drawn to Caltech by the recent experimental demonstration of quantum logic by Jeff Kimble's group. Here was a place where ground-breaking research was being done! Although I leaned more towards theory, I was willing to convert to an experimentalist if it meant being involved in this exciting new field. What a surprise it was to meet John Preskill that fateful week—a *theoretical* physicist at Caltech interested in quantum computation. John taught me about quantum error correction through Shor's code and I shared with him what limited knowledge I had about compression and Huffman coding. I realized that I had found the best of both worlds—a place where I could pursue theoretical quantum computing research and be close to quantum computing experiments at the same time.

Although I was familiar with scattered quantum computing results from the background research I undertook for my undergraduate Honors thesis, I learned a thousand-fold more from serving as the teaching assistant for John Preskill's new class on quantum computation and quantum information in 1997–98 and its team-taught version with Alexei Kitaev in 1998–99. It was a bit daunting to serve as TA for a class that had never been taught before, but I'm glad that I accepted the challenge—this Ph.D. thesis is built upon the material I learned there.

In September of 1997, in search of a research project with some real meat to it, John suggested studying ways to develop Kitaev's toric codes into a full-fledged architecture for fault-tolerant quantum computing. That sounded like a straightforward problem, or so I thought. Little did I know that pursuing it would result in a four-year collaborative effort incorporating concepts from so many different fields! The results of this investigation and its associated grand tour through various problems in mathematics, physics, and computer science, are reported in Chapter 4 and in [29].

Having learned so much the last time I was TA for a class I had never taken

before, I accepted in 1999 when Jeff Kimble asked me to TA his class on quantum optics. Once again, I learned much more than I think I would have if I was just taking the class. Perhaps the most important lesson I learned was the importance of doing theoretical science that is *useful* to experimentalists. That principle is reflected in the first chapter of this thesis, where I address the disconnect between the discrete-time language frequently used in quantum information theory and the continuous-time language frequently used in quantum optics.

After finishing the four-year project on fault-tolerance, I was eager to initiate a research project of my own, and hopefully one of shorter duration. I settled on the problem finding common ground between quantum control theory and quantum error correction. From what I had heard from Hideo Mabuchi and Andrew Doherty, both of these fields seemed to have similar goals, but radically different approaches. Reading background articles in both fields only reinforced that notion—the scientists working in the two fields formed essentially mutually exclusive sets. The specific problem I decided to address was one that had been gnawing at me for some time: How well does quantum error correction work when it is restricted to use (experimentally realistic) continuous and weak controls? Charlene Ahn and I taught ourselves about continuous measurement theory and quantum feedback theory from the background literature. We learned even more through many subsequent discussions with Andrew Doherty. I proposed a model for continuous-time quantum error correction that incorporated these new ideas, and Andrew, Charlene, and I explored the model in detail with Monte Carlo simulations as described in Chapter 3 and in [5]. Ultimately this project took a year to complete, but it was very rewarding. More significantly, it started me down the path of exploring problems lying in the intersection of quantum information theory and quantum control theory.

During a visit to MIT in 2002, I had the opportunity to present Eddie Farhi and Andrew Childs with an idea I had relating adiabatic algorithms and measurement. Because I had been thinking about continuous measurements from my previous research, I wondered if continuous measurements could be used elsewhere

in quantum information science, and in particular whether they could be used to mock up adiabatic evolution through the Zeno effect. I was glad to discover that Eddie and Andrew were excited by the idea, and they invited Sam Gutmann and Jeffrey Goldstone to a subsequent meeting where we hashed out in more detail how quantum measurement algorithms should work. The five of us and Enrico Deotto subsequently bounced many e-mails back and forth, which led to the material presented in Chapter 5 and in [22]. I am certain that this project wouldn't have developed as quickly as it did if it weren't for the wealth of expertise in quantum adiabatic algorithms that my MIT collaborators brought to this project.

As you can see, my graduate research experience has approximated a miniature random walk through quantum information science, but if a common thread is to be found, it would be that everything I have worked on is concerned with controlling quantum information, either to make it robust, to make it realistic, or to make it compute. I think that this pragmatic approach is the right one to take to make meaningful progress. For far too long, theoretical quantum mechanics research has been confined to philosophical questions and progress has been difficult to measure. If we challenge ourselves to explore the limits of quantum mechanics and information science through quantum information engineering problems, then we can learn meaningful things about quantum information science itself.

May, 2002 ANDREW J. LANDAHL

Contents

List of Figures

Notation

$\mathcal{H}_n$	Hilbert space of dimension n
$\mathcal{H}^{\otimes m}$	m-fold tensor product of $\mathcal{H}$
$\mathcal{B}$	$\mathcal{H}_2$, the Hilbert space of a qubit
$\mathbf{L}(\mathcal{H})$	linear operators on $\mathcal{H}$
$\mathbf{L}^+(\mathcal{H})$	positive linear operators on $\mathcal{H}$
$\otimes$	tensor product
$\oplus$	direct sum (and also addition modulo 2)
$\mathit{\Pi}_{\mathcal{H}}, \Pi_{\mathcal{H}}$	projector onto the Hilbert space $\mathcal{H}$
$\mathbb{1}_{\mathcal{H}}, I_{\mathcal{H}}$	identity operator on the Hilbert space $\mathcal{H}$
$[A, B]$	the operator commutator $AB - BA$
$A^\dagger$	Hermitian adjoint of A
z^*	complex conjugate of z
$\|\psi\rangle$	Dirac ket vector
$\langle\psi\|$	Dirac bra vector
$\langle\psi \,\|\, \phi\rangle$	inner product
$\|\psi\rangle\langle\psi\|$	projector onto $\|\psi\rangle$
$\|0\rangle$	spin-up state of a qubit $\binom{1}{0} = \|\uparrow\rangle$
$\|1\rangle$	spin-down state of a qubit $\binom{0}{1} = \|\downarrow\rangle$
ρ_c	conditioned density matrix
dW	real Wiener increment
$\mathcal{D}[c]$	decoherence (Lindblad) superoperator $\mathcal{D}[c]\rho = c\rho c^\dagger - \frac{1}{2}c^\dagger c\rho - \frac{1}{2}\rho c^\dagger c$
$\mathcal{H}[c]$	diffusion superoperator $\mathcal{H}[c]\rho = c\rho + \rho c^\dagger - \rho\,\mathrm{tr}(c\rho + \rho c^\dagger)$
$\langle\cdot\rangle_c$	expectation wrt. ρ_c
sgn	sign function, $\mathrm{sgn}\, x = \|x\|/x$, $\mathrm{sgn}\, 0 = 0$
log	logarithm base two
$[[n, k, d]]$	quantum error correcting code encoding k qubits

	in n having distance d
∂C	boundary of C
∂_{rel}	relative boundary of C
$H_2(p)$	Shannon entropy $-p\log p-(1-p)\log(1-p)$
$\Lambda^n(U)$	quantum controlled-U gate with n controls
$\hbar$	Planck's constant (set to one)
k_B	Boltzmann's constant (set to one)
β	inverse temperature $1/k_BT$
$f(n)=\mathcal{O}(g(n))$	there exist numbers C and N such that
	$f(n)\leq Cg(n)$ for all $n\geq N$
$f(n)=\mathrm{poly}(n)$	shorthand for $f(n)=n^{\mathcal{O}(1)}$

Abbreviations:

PVM	projection-valued measure
POVM	positive operator-valued measure
SME	stochastic master equation
CTQEC	continuous-time quantum error correction
CSS	Calderbank-Steane-Shor (*cf.* [19, 95])
SAW	self-avoiding walk
SAP	self-avoiding polygon
CNOT	controlled-NOT gate

Notation for matrices:

$$X \equiv \sigma_x = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \quad Y \equiv \sigma_y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix},$$
$$Z \equiv \sigma_z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} \quad I \equiv \mathbb{1} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$
$$H = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \quad CNOT = \Lambda(\sigma_x)$$

Chapter 1

Introduction

1.1 Quantum information science

Just exactly what is quantum information, and why would someone want to control it? As a first step in answering this question, it is worthwhile to contemplate what is meant by *information*. Loosely speaking, one has more information when one is more certain about which one of a number of mutually exclusive alternatives is true. Since probabilities also measure degrees of certainty, it's natural to expect that probability and information are related. It turns out that they are famously related through the notion of *entropy*, a quantity that Boltzmann was so proud to have invented that he had it engraved on his tombstone.

Okay, so information measures certainty. But what does information have to do with quantum mechanics? Quite a bit, actually. The central idea of quantum mechanics is that maximal information and complete information about a physical system are not the same. This idea shows up in fascinating quantum effects that defy common sense intuition. For example, when one tries to increase one's maximal information about a physical system, a disturbance is created so that some of the previous information one had disappears [43]. Even more strangely, one can perform operations on one system that change the information one has about a causally separated system [36, 13, 7]. These effects, and others like them, lie at the heart of the growing field of quantum information science.

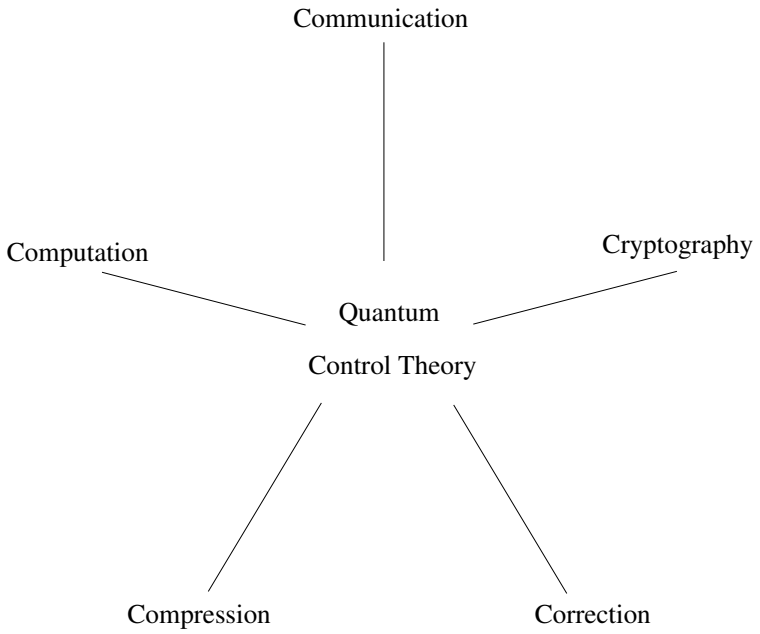

Figure 1.1: The five C's of quantum information science: communication, compression, computation, correction, and cryptography.

Quantum information science currently consists of five major subdisciplines: quantum communication theory, quantum compression theory, quantum computer science, quantum error correction, and quantum cryptography. Each subdiscipline is defined by a particular quantum information task. Often one is concerned with several of these tasks in conjunction. For example, one may wish to process some quantum data, protect it from noise, compress it, and then transmit it securely to trusted parties. In order to understand the extent to which these tasks may be accomplished, it is important to have a theory detailing the limitations and capabilities for establishing control of quantum systems. In other words, "quantum control theory" anchors these subdisciplines, as depicted in Fig. 1.1. I use quotes because I mean the term to represent the full array of quantum control possibilities discovered by quantum information science, not just the narrow set of mathematical techniques generalized from classical control theory.

Coming back to the original question, then, I would respond by defining quan-

tum information as a quantitative measure of the certainty one has about a quantum system, and I would suggest that it might be desirable to control this information for various technological applications including communication, compression, computation, correction, and cryptography. The goal of this thesis is to explore some new ways that quantum mechanics can be harnessed to control quantum information.

1.2 Summary by chapter

The organization of this thesis is as follows. In Chapter 2, I review an eclectic set of background material in quantum information science that will be useful later in the thesis. The material there can be found elsewhere; I include it as a convenience for the reader. In particular, I review the rules of quantum mechanics when both maximal and non-maximal information is available, and list some important properties and equivalent representations of density matrices and quantum operations. I also review perturbation theory, the adiabatic theorem, and the adiabatic approximation—well-worn tools of quantum mechanics that I will make use of in subsequent chapters.

Chapter 3 is the beginning of new research results. I begin by reviewing continuous measurement theory, quantum feedback control, and quantum error correction. I then propose a new method for continuously correcting errors in unknown quantum states that uses ideas from these approaches. I present Monte Carlo simulation results of an analysis of this method applied to a three-qubit system, and show how this method can outperform rate-limited quantum error correction in that system. The results of this research are also reported in [5].

In Chapter 4, I explore a new method for achieving fault-tolerance in quantum systems that is built upon topological quantum error correcting codes. I review these codes and present a list of architectural desiderata conducive to fault-tolerant quantum design. I reflect on how such a design may be adapted to be physically fault-tolerant, and propose a correction algorithm for these codes. I show that the

fault-tolerance behavior of the errors and correction algorithm in this code can be mapped onto an order-disorder transition in a statistical model known as the random-bond Ising model. I discuss order parameters in this model and its generalization to three dimensions, and use the critical behavior of these statistical mechanical models to numerically estimate the accuracy threshold for quantum computation when fault-tolerant design principles haven't been incorporated. I then arrive at an analytical bound on this threshold from a combinatorial counting argument based on self-avoiding polygons. I explore the effects of finite time intervals on the correction algorithm, and adapt it accordingly. I argue that the threshold for fault-tolerant quantum computation should be close to the threshold for fault-tolerant quantum storage using these codes, and proceed to analyze the threshold for fault-tolerant quantum storage by explicit analysis of fault-tolerant circuits used in the recovery algorithm. I present techniques for robustly encoding unknown quantum states via a local tesselation-increasing algorithm, and provide explicit constructions for a universal set of gates for fault-tolerant quantum computation, although I do not analyze the thresholds for these protocols. I conclude with a discussion of a generalization of this method to topological codes in four spatial dimensions, and calculate a threshold for storage using these codes which uses entirely local processing, both quantum and classical. The results of this research are also reported in [29].

Finally, in Chapter 5, I present a new class of quantum algorithms that can solve combinatorial search problems using only a sequence of measurements. I review adiabatic algorithms and the Zeno effect, and present a dynamical model for measurement originally proposed by von Neumann [103] and digitized by Kitaev [59]. I analyze the running time of a measurement algorithm which simulates an adiabatic algorithm by appealing to this dynamical model, and show that it is polynomially related to running time of the adiabatic algorithm it simulates. I then study the specific problem of unstructured quantum search proposed by Grover [53], and show that the quantum measurement algorithm may be adapted to the special properties of this problem so that it saturates the bound for the

fastest possible quantum algorithm solving the problem, namely one which has a quadratic speedup in the number of oracle calls relative to the best possible classical algorithm. The results of this research are also reported in [22].

Chapter 2

Background

The most incomprehensible thing about the universe is that it is comprehensible.

—A. Einstein [35]

Experimental science places sharp constraints on the mathematical objects one can use to consistently represent one's knowledge of Nature. In this chapter, I present these constraints as a set of representational rules that every physical theory consistent with these experiments must obey.

I begin by first stating the rules for when maximal information is available (*quantum mechanics*) and then generalize to the case when non-maximal information is available (*quantum information mechanics*). I elaborate in detail some of the properties of representations for states and dynamics. (Later in the thesis, notably in Chapters 3 and 5, I elaborate the properties of the representation for measurement.) Finally, I review some useful mathematical techniques for describing quantum mechanics in perturbative and adiabatic approximations.

2.1 Quantum mechanics

Maxwell writes [75]

> *Physical science is that department of knowledge which relates to the order of nature, or, in other words, to the regular succession of events.*

Evidently, then, a physical theory needs to represent "events," their "regular succession," and a means for obtaining "knowledge" about them. In more modern language, we would say that a physical theory needs to represent *states*, *dynamics*, and *measurements*. Moreover, the possibility of comparing events suggests a physical theory should also represent *subsystems* of a larger system.

Quantum mechanics is not a physical theory in and of itself. Rather, it is a set of rules for how the concepts of a physical theory should be represented. The rules are designed to ensure that physical theories constructed within them are consistent with the results of prior experiments. Sometimes called *axioms* or *postulates*, I prefer to simply call them *quantum rules* because, unlike axioms or postulates, they can be challenged by experiment.

2.1.1 The rules of quantum mechanics

Representation Rule 2.1.1 (States). *The state of a physical system is represented by a* ray ψ *in a* Hilbert space $\mathcal{H}$*.*

Representation Rule 2.1.2 (Dynamics). *The time-evolution of a physical system is represented by a one-parameter* unitary group $\{U(t)\}_{t\in\mathbb{R}}$ *in the space* $\mathbf{L}(\mathcal{H})$ *of linear operators on* $\mathcal{H}$*. The self-adjoint operator* $H(t)$ *generating* $U(t)$ *is called the* Hamiltonian *of the system.*

Representation Rule 2.1.3 (Measurement). *A measurement of a state* ψ *is represented by the* projection-valued measure (PVM) $\mathcal{P} : \Pi_i \mapsto \psi^\dagger \Pi_i \psi$ *on* $\mathcal{H}$*, where* $\sum_i \Pi_i = \mathbb{1}$ *and* $\Pi_i \Pi_j = \delta_{ij}$*. The measure is perceived as probability.*

Representation Rule 2.1.4 (Subsystems). *The representation of states, dynamics, and measurements on subsystems are combined via the* tensor product *($\otimes$) to create a representation of the corresponding objects on the combined system.*

Rather than working with the ray representation of states, I will often use *Dirac notation* instead:

Definition 2.1.1 (*Dirac notation*). In *Dirac notation*, a ray $\psi \in \mathcal{H}$ is represented by a unit-norm vector in the equivalence class of ψ called the *ket* for ψ in $\mathcal{H}$, denoted by $|\psi\rangle$. The multiplicative phase freedom $e^{i\varphi}$ (where φ is real) in the ket representation of ψ is unphysical; only the ray is physical. (However, the relative phase and amplitude of rays is physical.) The linear functional dual to this ket is called a *bra* and is denoted by $\langle\psi|$. The inner product of kets $|\psi\rangle$ and $|\varphi\rangle$ is denoted by $\langle\psi\,|\,\varphi\rangle$, so that the unit-norm condition for $|\psi\rangle$ reads $\langle\psi\,|\,\psi\rangle = 1$.

When I want to emphasize the irrelevance of the overall phase in Dirac notation, I will represent ψ by the rank-1 projector $|\psi\rangle\langle\psi|$ instead. To simplify notation, I will frequently concatenate kets, operators, and their Hilbert-space labels to denote the tensor product. For example, I might express $U_A \otimes U_B(|0\rangle_A \otimes |1\rangle_B)$ as $U_A \otimes U_B\,|0\rangle_A\,|1\rangle_B$ or $U_A \otimes U_B\,|01\rangle_{AB}$ or even $U_A U_B\,|01\rangle_{AB}$.

2.1.2 General remarks on quantum mechanics

In Chapters 3 and 5, I will re-examine these rules and their ramifications as I develop new techniques for solving quantum information processing problems. Instead of waiting until then to comment on these rules, I would like to make some general remarks regarding them before proceeding further.

Remark 2.1.1 (On interpretations). Addressing just what exactly probability *means* is the subject of *interpretations* of quantum mechanics. As far as I can tell, progress in this metaphysical endeavor is measured only by aesthetics. I'll lay my cards on the table and confess that I subscribe to the *Bayesian interpretation*[1] of

[1] I reserve the right to change my metaphysics in the future!

quantum mechanics; namely, I believe that probabilities represent subjective states of knowledge (or belief) about the outcomes of future events, which are updated according to Bayes' rule. Mine is certainly not the only view! For a lucid description of the Bayesian viewpoint as well as a comparison to other epistemological viewpoints, see the well-written (but unabashedly biased) *samizdat* by Caves on his home page [21].

Remark 2.1.2 (On pictures). Suppose all states and observables in quantum mechanics were rotated by some unitary operator V, *i.e.*, suppose $\psi \to V\psi$ and $\Pi \to V\Pi V^\dagger$. The predictions of this new theory are the same as the old one, because $\psi^\dagger V^\dagger V \Pi V^\dagger V \psi = \psi^\dagger \Pi \psi$. Hence there is a freedom in what objects one uses to represent the states and measurements of a physical theory. Each choice of the unitary V corresponds to what is called a *picture* of quantum mechanics. The reason for moving from one picture to another is to make the dynamics appear simpler.

In this thesis, I will mostly work in the *Schrödinger picture*, where $V = \mathbb{1}$. The infinitesimal form of dynamics in this picture is the well-known *Schrödinger equation*:

$$\frac{d}{dt}|\psi(t)\rangle = \frac{-i}{\hbar} H(t)\,|\psi(t)\rangle.$$

When discussing quantum error correction in Chapters 3 and 4, I will sometimes switch pictures and work in the *Heisenberg picture*, where $V = U$, the evolution operator. The infinitesimal form of dynamics in this picture is the *Heisenberg equation*:

$$\frac{d}{dt}A(t) = \frac{-i}{\hbar}[H(t), A(t)],$$

where $A(t)$ is the Hermitian operator (observable) corresponding to the measurement being acted upon.

Remark 2.1.3 (On Planck's constant). The origin of $\hbar$ in each of the pictures above comes from experiment and appears to be universal, although it is not explicitly included in the quantum rules. I find its appearance the most mysterious

aspect of quantum mechanics (even more than entanglement). Is $\hbar$ a phenomenological parameter? Is it fundamental? Computable? Information-bearing? Nobody really knows. Perhaps scientists of the future will learn how $\hbar$ came to take on the value that it has. In this thesis, I'll take the pragmatic view and simply accept that it has a certain value. To make it even more ignorable, I'll set $\hbar = 1$; in other words, I'll work in units which are set by the value of $\hbar$.

Remark 2.1.4 (On the need for measurement). It is natural to ask whether or not one can derive the quantum measurement rule 2.1.3 from the other three rules. On the one hand, it seems it ought to be possible because the measuring system, the measured system, and their interactive dynamics can be described by the other quantum rules. On the other hand, the quantum measurement rule does not explicitly refer to time, whereas the dynamics rule does. This problem comes up in Chapter 5, where it will be necessary to use a *simulation* of the measurement process by unitary dynamics in order to calculate the computational complexity of measurement.

Remark 2.1.5 (On the need for subsystems). As with the measurement rule, it is unclear whether or not the subsystem rule is fundamental or derivable. In particular, it is known that dynamics can impose subsystem structure through superselection rules. Some have speculated that all subsystem structure is established in this way. Whether or not this is the case remains an open, and perhaps unresolvable, question.

Remark 2.1.6 (On the invertibility of the rules). It is important to recognize that the quantum rules can only be used in the forward direction. Every physical object may be represented by one of these mathematical objects, but not every such mathematical object can be realized as a corresponding physical object. Physical science and computer science place additional restrictions on which objects are physically realizable. For example, causality and computability place restrictions

on physically realizable measurements as shown in [12, 11] and [78]. Similarly the observed statistics of particles with spin (famously) place symmetry restrictions on physically realizable quantum states. In fact, the rules themselves were arrived at by experiments demonstrating that the mathematical objects used to represent physical objects had to be of the form specified by the rules.

2.2 Quantum information mechanics

The rules of quantum mechanics are useful when one has maximal information about a quantum system. When one does not, rays, unitary groups, PVMs, and tensor products are (in general) no longer the correct mathematical objects to use to represent Nature. Instead, one must use density matrices, quantum operations, positive operator-valued measures (POVMs), and direct sums—notions I elaborate in this section via *quantum information rules* analogous the quantum rules of the preceding section.

What is the source of the non-maximality of information? There is no unique answer. In *quantum statistical mechanics*, the source is a coarse-graining of microscopic degrees of freedom. In *open quantum systems mechanics*, the answer is uncontrollable couplings to an external environment. In *quantum communication theory* the answer is a noisy quantum channel. In each of these scenarios, there is an uncontrollable/unobservable part of the quantum system that is deemed responsible for the lack of maximal information. The idea that there exists a maximal information description on a larger system is the basis for the *purification principle*[2] , which I will discuss in Section 2.3.

Without further ado, here are the quantum information rules:

2.2.1 The rules of quantum information mechanics

Representation Rule 2.2.1 (States). *The state of a physical system is represented by a Hermitian, unit-trace, positive operator ρ on a Hilbert space $\mathcal{H}$.*

[2]This principle is sometimes colorfully referred to as the "Church of the larger Hilbert space".

Representation Rule 2.2.2 (Dynamics). *The time-evolution of a physical system is represented by a linear hermiticity-preserving, trace-preserving, completely positive map on* $\mathbf{L}(\mathcal{H})$.

Representation Rule 2.2.3 (Measurement). *A measurement of a state* ρ *is represented by a positive operator-valued measure (POVM) on* $\mathbf{L}(\mathcal{H})$.

Representation Rule 2.2.4 (Subsystems). *Let* $\{A_{ij}\}$ *be a collection of mathematical objects that represent states, dynamics, or measurements on subsystems* $\{S_{ij}\}$. *Then the representation of the object they form on the combined system is a* direct sum ($\oplus$) *over the* tensor product ($\otimes$) *of the corresponding objects on the subsystems:* $\bigoplus_i \bigotimes_j A_{ij}$.

2.3 The density matrix

In this section, I motivate the density matrix as a way to represent an ensemble of quantum states. I then present some well-known theorems regarding density matrices. The proofs of these theorems can be found in many places; for proofs with a quantum information-theoretic flair, see [80, 86, 63].

Consider a probability distribution $\{p_1 \ldots, p_i, \ldots\}$ over a finite set of states given by $\{|\psi\rangle, \ldots, |\psi_i\rangle, \ldots\}$. That is, consider the *ensemble* $\mathcal{E} = \{p_i, |\psi_i\rangle\}$. Suppose we draw a state from this ensemble and measure it. The probability that it will be observed in the subspace $\mathcal{M}$ is

$$p(\mathcal{E}, \mathcal{M}) = \sum_i p_i \, p\left(|\psi_i\rangle, \mathcal{M}\right) \tag{2.1}$$

$$= \sum_i p_i \langle\psi_i| \, \Pi_{\mathcal{M}} \, |\psi_i\rangle \tag{2.2}$$

$$= \sum_i p_i \operatorname{tr}\left(\Pi_{\mathcal{M}} \, |\psi_i\rangle\langle\psi_i|\right) \tag{2.3}$$

$$= \operatorname{tr} \Pi_{\mathcal{M}} \rho, \tag{2.4}$$

where $\rho = \sum_i p_i \, |\psi_i\rangle\langle\psi_i|$ is called the *density matrix* or *density operator* for $\mathcal{E}$.

Because the probabilities and associated measurement outcomes are encoded in ρ, it makes sense to *define* ρ as the quantum state of the ensemble $\mathcal{E}$ where the rule (2.4) defines the probability of measurement outcomes.

A useful way to classify density matrices is in terms of their *purity*:

Definition 2.3.1 (*Purity*). The *purity* of a density matrix ρ is defined as $P(\rho) = \operatorname{tr} \rho^2$. A density matrix is said to be a *pure state* if $P(\rho) = 1$; otherwise it is said to be a *mixed state*.

It is straightforward to verify that pure states are exactly the rank-1 density matrices, *i.e.*, ρ is a pure state iff $\rho = |\psi\rangle\langle\psi|$ for some $|\psi\rangle$.

When does an operator represent the quantum state of an ensemble? In other words, when is an operator a density operator? The answer is given by the following theorem:

Theorem 2.3.1 (Axiomatic characterization). *An operator ρ on the Hilbert space $\mathcal{H}$ is a density matrix iff it is a positive unit-trace Hermitian operator, i.e., iff*

$$i)\ \forall\, |\psi\rangle \in \mathcal{H},\ \langle\psi|\,\rho\,|\psi\rangle \geq 0; \quad ii)\ \operatorname{tr}\rho = 1; \quad iii)\ \rho = \rho^\dagger.$$

This theorem provides an equivalent definition of density matrices, which is why it is given the status of a representational rule in Section 2.2.1. This theorem also has two useful corollaries which characterize the space of density matrices:

Corollary 2.3.1.1 (Convexity). *The density matrices on the Hilbert space $\mathcal{H}$ form a convex subset of the Hermitian operators on $\mathcal{H}$.*

Corollary 2.3.1.2 (Extremal points). *The extremal points of the set of density matrices are the pure states.*

It is tempting to think that further generalization is achieved by considering ensembles of mixed states (ensembles of ensembles). However, any ensemble of mixed states $\mathcal{E} = \{p_i, \rho_i\}$, where $\rho_i = \sum_k p_k^{(i)} |\psi_k^{(i)}\rangle\langle\psi_k^{(i)}|$, is equivalent to the ensemble of pure states $\mathcal{E}' = \{p_i p_k^{(i)}, |\psi_k^{(i)}\rangle\}$, where some states in the ensemble $\mathcal{E}'$ may be repeated. Hence it suffices to restrict attention to ensembles of pure states.

A surprising observation is that the mapping $g : \mathcal{E} \to \rho$ is not injective, *i.e.*, different ensembles can give rise to the same state ρ. Apparently some information about the preparation of a state has no physical consequence. The following theorem cements this idea by characterizing the freedom in choice of ensemble that one may ascribe to a density operator:

Theorem 2.3.2 (Ensemble freedom). *The ensembles* $\mathcal{E} = \{p_i, |\psi_i\rangle\}$ *and* $\mathcal{E} = \{q_i, |\varphi_i\rangle\}$ *give rise to the same density matrix* ρ *iff* $\sqrt{p_i}\,|\psi_i\rangle = \sum_j u_{ij}\sqrt{q_j}\,|\varphi_j\rangle$ *for some unitary matrix with entries* u_{ij}.

A recurring theme in quantum information mechanics is the purification principle alluded to earlier. To define the principle rigorously, it is necessary to first define the *partial trace*:

Definition 2.3.2 (*Partial trace*). Let ρ_{AB} be a density matrix on $\mathbf{L}(\mathcal{H}_A \otimes \mathcal{H}_B)$. The *partial trace* $\mathrm{tr}_B : \mathbf{L}(\mathcal{H}_A \otimes \mathcal{H}_B) \to \mathbf{L}(\mathcal{H}_A)$ of ρ_{AB} over the space $\mathcal{H}_B$ is defined by the linear extension of the map

$$\mathrm{tr}_B(|a_1\rangle\langle a_2| \otimes |b_1\rangle\langle b_2|) \equiv |a_1\rangle\langle a_2|\,\mathrm{tr}\,|b_1\rangle\langle b_2|\,, \tag{2.5}$$

where $|a_1\rangle, |a_2\rangle \in \mathcal{H}_A$ and $|b_1\rangle, |b_2\rangle \in \mathcal{H}_B$. The output of the partial trace, ρ_A, is called the *reduced density matrix* on $\mathcal{H}_A$.

Using the partial trace, the purification principle may be formulated as follows:

Theorem 2.3.3 (Purification principle). *Let* ρ_A *be a density matrix on* $\mathcal{H}_A$. *Then* $\rho_A = \mathrm{tr}_B\,|\psi\rangle_{AB\,AB}\langle\psi|$ *for some pure state* $|\psi\rangle_{AB} \in \mathcal{H}_A \otimes \mathcal{H}_B$, *where* $\mathcal{H}_B$ *is a Hilbert space having* $\dim \mathcal{H}_B \leq \dim \mathcal{H}_A$. *The ket* $|\psi\rangle_{AB}$ *is called a* purification *of* ρ_A.

As might be expected for an extension to a larger Hilbert space, the purification of a density matrix ρ is not unique. The following theorem characterizes the freedom in purifications:

Theorem 2.3.4 (Purification freedom). *Let* $|\psi\rangle_{AB}, |\varphi\rangle_{AB} \in \mathcal{H}_A \otimes \mathcal{H}_B$ *be purifications of* $\rho_A \in \mathbf{L}(\mathcal{H}_A)$*. Then there exists a unitary operator* $U \in \mathbf{L}(\mathcal{H})$ *such that* $|\psi\rangle = (\mathbb{1} \otimes U)\,|\varphi\rangle$*.*

This purification freedom also allows pure bipartite quantum states to be expressed in a standard form called the *Schmidt decomposition*, as the following corollary shows:

Corollary 2.3.4.1 (Schmidt decomposition). *Let* $|\psi\rangle_{AB} \in \mathcal{H}_A \otimes \mathcal{H}_B$*. Then there exist orthonormal sets of vectors* $\{|i\rangle_A\} \subset \mathcal{H}_A$ *and* $\{|i\rangle_B\} \subset \mathcal{H}_B$ *and nonnegative numbers* p_i *summing to 1 such that* $|\psi\rangle_{AB} = \sum_i \sqrt{p_i}\,|i\rangle_A\,|i\rangle_B$*. The numbers* $\sqrt{p_i}$ *are called the* Schmidt coefficients *of* $|\psi\rangle_{AB}$*.*

The real power behind the Schmidt decomposition is that the *Schmidt coefficients*, $\sqrt{p_i}$ are, by construction, invariant under local unitary operations on each subsystem—such operations merely rotate the orthonormal bases of each subsystem to other orthonormal bases of those subsystems. A quantity that is invariant under all such transformations is its *Schmidt number*:

Definition 2.3.3 (*Schmidt number*). The *Schmidt number* of a bipartite pure state $|\psi\rangle_{AB}$ is the number of nonzero Schmidt coefficients in its Schmidt decomposition.

The Schmidt number has many applications in the study of *entanglement*, a complete discussion of which is beyond the scope of this thesis.

In the special case of a density matrix on a two-dimensional Hilbert space, the density matrix can be represented by a three-dimensional *Bloch vector*. This representation is called the *Bloch sphere representation* (although it probably should be called the *Bloch ball representation*) for the density matrix:

Theorem 2.3.5 (Bloch sphere representation). *A density matrix* $\rho \in \mathcal{B}$*, where* $\mathcal{B}$ *is the Hilbert space of a qubit, can be uniquely expressed as*

$$\rho = \frac{1}{2}\left(\mathbb{1} + \mathbf{p} \cdot \boldsymbol{\sigma}\right), \tag{2.6}$$

where $\mathbf{p} \in \mathbb{R}^3$ *satisfies* $\|\mathbf{p}\| \leq 1$ *and* $\boldsymbol{\sigma} = (\sigma_1, \sigma_2, \sigma_3)$*, where the* σ_i *are the (non-identity) Pauli matrices. The vector* $\mathbf{p} \in \mathbb{R}^3$ *is called the* Bloch vector *for* ρ*, the space spanned by Bloch vectors is called the* Bloch ball*, the boundary of which is called the* Bloch sphere*.*

One of the most useful properties of the Bloch ball representation is that the boundary of the Bloch ball, the Bloch sphere, corresponds precisely to the extremal points of density matrices, the pure states. The boundary points of a generic space are not necessarily extremal, as can be seen by the example of a triangle, but for qubit density matrices (and, as it turns out, only for qubit density matrices), boundary points and extremal points are one and the same.

As a parting remark on density matrices, I would like to point out that density matrices represent the most general classical states as well, namely probability distributions over orthogonal sets of pure states. In fact, one can *define* what one means by "classical" by this subset of density matrices:

Definition 2.3.4 (*Classical*). A quantum state is said to be *classical relative to the basis* β when the density matrix describing that state is diagonal in the β basis.

Notice that the definition of classicality is always with reference to some basis. A *classical system* is one in which the dynamics keep classical states classical.

2.4 Quantum operations

Because density matrices are the most general representation for states of a physical system, it is natural to ask what the corresponding most general representations are for dynamics and measurements. A reasonable criterion for these representations is that they preserve the ensemble interpretation of density matrices. In other words, dynamics and measurements should act linearly on density matrices. A linear map between Hilbert spaces is a *superoperator*:

Definition 2.4.1 (*Superoperator*). A *superoperator* Q is a linear map between operator spaces of Hilbert spaces: $Q : \mathbf{L}(\mathcal{H}) \to \mathbf{L}(\mathcal{K})$.

A subset that is usually regarded as the "right" one to represent physically realizable superoperators is the set of *quantum operations*. Quantum operations have several representations, all of which are equivalent. It is therefore a matter of taste as to which one is taken as the definition. I shall take the axiomatic representation as the definition below, and consider the equivalence of the other representations to be theorems. As in the previous discussion regarding density matrices, proofs of these theorems can be found in [80, 86, 63].

Definition 2.4.2 (*Quantum operation; Axiomatic representation*). A map Q is called a *quantum operation* if it is a completely positive superoperator that maps density matrices to density matrices. In other words, the superoperator Q is a quantum operation on $\mathbf{L}(\mathcal{H})$ iff it satisfies the following axioms (note that ρ is not necessarily a density matrix in the criteria below):

$$\begin{array}{lll} i) & \forall \rho \in \mathbf{L}(\mathcal{H}) & \operatorname{tr} Q(\rho) = \operatorname{tr} \rho \\ ii) & \forall \rho \in \mathbf{L}(\mathcal{H}) & Q(\rho)^\dagger = Q(\rho^\dagger) \\ iii) & \forall \rho \in \mathbf{L}^+(\mathcal{H} \otimes \mathcal{K}) & (Q \otimes \mathbb{1}_{\mathbf{L}(\mathcal{K})})(\rho) \geq 0, \end{array}$$

where $\mathcal{K}$ is any Hilbert space and $\mathbf{L}^+(\mathcal{G})$ represents the positive operators on $\mathcal{G}$.

This definition is equivalent to the representational rule for dynamics stated in Sec. 2.2.1.

Theorem 2.4.1 (Unitary representation). *A superoperator Q is a quantum operation on $\mathbf{L}(\mathcal{H}_A)$ iff it can be expressed as*

$$Q(\rho_A) = \operatorname{tr}_B \left[U_{AB} (\rho_A \otimes |\psi\rangle_B \, {}_B\langle\psi|) U_{AB}^\dagger \right] \tag{2.7}$$

for some $|\psi\rangle_B \in \mathcal{H}_B$ and some unitary operator $U_{AB} \in \mathbf{L}(\mathcal{H}_A \otimes \mathcal{H}_B)$.

Theorem 2.4.2 (Operator-sum representation). *A superoperator Q is a quantum operation on $\mathbf{L}(\mathcal{H})$ iff it can be expressed as*

$$Q(\rho_A) = \sum_i A_i \rho A_i^\dagger, \qquad \text{where } \sum_i A_i^\dagger A_i = \mathbb{1}. \tag{2.8}$$

Theorem 2.4.3 (Matrix representation). *A superoperator Q on $\mathbf{L}(\mathcal{H})$, expressed in coordinate form as*

$$Q(|i\rangle\langle j|) = \sum_{i'j'} Q_{(i'i)(j'j)} \left|i'\right\rangle\left\langle j'\right|, \tag{2.9}$$

is a quantum operation iff the following are satsified

$$\begin{aligned} &i) \quad \textstyle\sum_k Q_{(ki)(kj)} = \delta_{ij} \\ &ii) \quad Q^*_{(i'i)(j'j)} = Q_{(j'j)(i'i)} \\ &iii) \quad (Q_{(i'i)(j'j)}) \text{ is a positive matrix.} \end{aligned}$$

2.5 Perturbation theory

Few problems can be solved exactly using the mathematical framework of quantum mechanics. What might be called the "art" of physics is first determining which effects are the most important in a problem and then applying an appropriate approximation method. One well-developed approximation method is the perturbation method, useful when one has a problem that is only slightly deformed, or perturbed, from a previously solved problem. The basic approach in the perturbation method is to expand the problem and its solution in a power series in the perturbation, keeping only the lowest order terms.

In this section, I will review the application of the perturbation method to the problem of finding the eigenstates of a time-independent Hamiltonian that is only slightly perturbed from a time-independent Hamiltonian having a discrete spectrum of nondegenerate eigenvalues. For obvious reasons, this procedure is

called *nondegenerate time-independent perturbation theory*. My discussion here closely follows Messiah [77].

Let $H^{(0)}$ be a time-independent Hamiltonian with a discrete spectrum of nondegenerate energy eigenvalues $\{E_i^{(0)}\}$ and corresponding eigenvectors $\{|E_i^{(0)}\rangle\}$. Namely, let

$$H^{(0)}|E_i^{(0)}\rangle = E_i^{(0)}|E_i^{(0)}\rangle, \tag{2.10}$$

where

$$\langle E_i^{(0)}|E_j^{(0)}\rangle = \delta_{ij} \tag{2.11}$$

and

$$\sum_i |E_i^{(0)}\rangle\langle E_i^{(0)}| = \mathbb{1}. \tag{2.12}$$

Consider the perturbed Hamiltonian $H = H^{(0)} + \delta H'$ arising from the time-independent perturbation $\delta H'$. Let E_i be the energy eigenvalue of H that tends to $E_i^{(0)}$ when $\delta \to 0$. It will also be a nondegenerate eigenvalue when δ is small:

$$H|E_i\rangle = E_i|E_i\rangle. \tag{2.13}$$

The corresponding eigenvector is defined up to a constant, which we shall fix via

$$\langle 0\,|\,E_i\rangle = 1, \tag{2.14}$$

where $|E_i^{(0)}\rangle \equiv |0\rangle$.

When δ is sufficiently small, the changes in the energy levels caused by H' are much smaller than the differences between them, so the new energy eigenvalues and (unnormalized) eigenvectors can be expanded in terms of the old ones in a power series:

$$E_i = E_i^{(0)} + \delta\varepsilon_i^{(1)} + \delta^2\varepsilon_i^{(2)} + \cdots \tag{2.15}$$

$$|E_i\rangle = |0_i\rangle + \delta\,|1_i\rangle + \delta^2\,|2_i\rangle + \cdots. \tag{2.16}$$

By eq. (2.14), the perturbative eigenvectors are orthogonal to the $\delta \to 0$ eigenvector:

$$\langle 0_i \,|\, 1_i \rangle = \langle 0_i \,|\, 2_i \rangle = \cdots = \langle 0_i \,|\, n_i \rangle = 0. \tag{2.17}$$

Substituting the expansions (2.15) and (2.16) into eq. (2.13) and matching terms of the same order, one obtains the following set of equations:

$$\begin{aligned}
&(H^{(0)} - E^{(0)})\,|0_i\rangle &&&&&&&&= 0\\
&(H^{(0)} - E^{(0)})\,|1_i\rangle &+\ &(\delta H' - \varepsilon_i^{(1)})\,|0_i\rangle &&&&&= 0\\
&(H^{(0)} - E^{(0)})\,|2_i\rangle &+\ &(\delta H' - \varepsilon_i^{(1)})\,|1_i\rangle &-\ &\varepsilon_i^{(2)}\,|0_i\rangle &&&= 0\\
&\quad\cdots\\
&(H^{(0)} - E^{(0)})\,|n_i\rangle &+\ &(\delta H' - \varepsilon_i^{(1)})\,|(n-1)_i\rangle &+\ &\cdots &-\ &\varepsilon_i^{(n)}\,|0_i\rangle &= 0.
\end{aligned} \tag{2.18}$$

The first-order correction is usually all that is considered—when higher order corrections are important, the perturbation method begins to break down.

The first-order correction to the eigenstate $|E_i^{(0)}\rangle$ is

$$|1_i\rangle = \sum_j |E_j^{(0)}\rangle\langle E_j^{(0)}|1_i\rangle, \tag{2.19}$$

where $\langle E_j^{(0)}|1_i\rangle$ is found by projecting the first of eqs. (2.18) onto the bra $\langle E_j^{(0)}|$ (and recalling the orthogonality condition (2.17)). When $i \neq j$, the resulting overlap is

$$\langle E_j^{(0)}|1_i\rangle = \frac{1}{E_i^{(0)} - E_j^{(0)}}\langle E_j^{(0)}|\delta H'|E_i^{(0)}\rangle, \tag{2.20}$$

and when $i = j$, the resulting overlap vanishes by the orthogonality condition (2.17). Hence, to first order in δ, the new (unnormalized) eigenstates are

$$|E_i\rangle \cong |E_i^{(0)}\rangle + \delta \sum_{j\neq i} \left(\frac{\langle E_j^{(0)}|H'|E_i^{(0)}\rangle}{E_i^{(0)} - E_j^{(0)}} \right) |E_j^{(0)}\rangle. \tag{2.21}$$

A quantity that will be useful in Chapter 3 is the *error distance* d_E between

associated energy eigenstates of the perturbed and unperturbed Hamiltonians:

$$d_E(|E_i\rangle,\, |E_i^{(0)}\rangle) \equiv 1 - |\langle E_i|E_i^{(0)}\rangle|^2. \tag{2.22}$$

In the present context, the error distance can be interpreted as the probability that the perturbing Hamiltonian $\delta H'$ will cause an "erroneous" transition to a different state. Although the error distance can sometimes be calculated exactly, it is helpful to bound it generically in terms of two variables: the smallest separation in energy eigenvalues of $H^{(0)}$, or *minimum gap*,

$$g = \min_{i \neq j} \left(E_i^{(0)} - E_j^{(0)} \right), \tag{2.23}$$

and the minimum variance of H' in the ith state of $H^{(0)}$, or *state perturbation variance*,

$$(\Gamma_i)^2 = (\Delta H')_i = \langle E_i^{(0)}|(H')^2|E_i^{(0)}\rangle - \langle E_i^{(0)}|H'|E_i^{(0)}\rangle^2. \tag{2.24}$$

Using these two quantities, and using the normalized version of eq. (2.21), one can bound the error probability of the perturbed eigenstates:

$$\begin{aligned}
d_E(|E_i\rangle, |E_i^{(0)}\rangle) &= 1 - |\langle E_i|E_i^{(0)}\rangle|^2 && (2.25)\\
&\cong 1 - \left(1 + \delta^2 \sum_{j \neq i} \frac{|\langle E_j^{(0)}|H'|E_i^{(0)}\rangle|^2}{|E_i^{(0)} - E_j^{(0)}|^2} \right)^{-1} && (2.26)\\
&\cong \delta^2 \sum_{j \neq i} \frac{|\langle E_j^{(0)}|H'|E_i^{(0)}\rangle|^2}{|E_i^{(0)} - E_j^{(0)}|^2} && (2.27)\\
&\geqslant \frac{\delta^2}{g^2} \left(\sum_j \langle E_i^{(0)}|H'|E_j^{(0)}\rangle\langle E_j^{(0)}|H'|E_i^{(0)}\rangle - |\langle E_i^{(0)}|H'|E_i^{(0)}\rangle|^2 \right) && (2.28)\\
&= \frac{\delta^2 \Gamma_i^2}{g^2}. && (2.29)
\end{aligned}$$

2.6 The adiabatic theorem

The *adiabatic theorem* is a powerful theorem in quantum mechanics that relates the asymptotic behavior of a unitary operator generated by a time-varying Hamiltonian and the projectors onto the eigenspaces of that Hamiltonian. Much of my treatment of this theorem[3] and its associated approximation is adapted from Messiah [77].

Theorem 2.6.1 (Adiabatic theorem). *Consider a Hamiltonian that continuously varies from H_0 at time t_0 to H_1 at time t_1 such that its spectrum of eigenvalues $E_1, E_2, \ldots$ remains discrete throughout the variation. Let $H(s)$ denote the Hamiltonian at time $t = t_0 + sT$, where*

$$T = t_1 - t_0 \qquad s = \frac{t - t_0}{T}, \tag{2.30}$$

and let $E_j(s)$ and $\Pi_j(s)$ denote the eigenvalues and associated subspace projectors of $H(s)$. Suppose that the following conditions are satisfied:

(*i*) (Continuity) *$E_j(s)$ and $\Pi_j(s)$ are continuous functions of s.*

(*ii*) (Non-crossing) *$E_j(s) \neq E_k(s)$ when $j \neq k$ for all $s \in [0, 1]$.*

(*iii*) (Differentiability) *$\frac{d\Pi_j}{ds}$ and $\frac{d^2\Pi_j}{ds^2}$ are piecewise continuous functions on $[0, 1]$.*

Then the unitary evolution operator $U_T(s) \equiv U(t, t_0)$ generated by $H(s)$ via

$$U_T(s) = \int_0^s d\sigma \exp(-iH(\sigma)) \tag{2.31}$$

$$= \mathbb{1} + i \int_0^s TH(\sigma)U(\sigma)d\sigma \tag{2.32}$$

[3]The theorem is even more general than the statement I give for it here. The restrictions I place are sufficient for how I shall consider it in Chapter 5, and make the proof simpler.

has the following asymptotic property:

$$\lim_{T\to\infty} U_T(s)\Pi_j(0) = \Pi_j(s) \lim_{T\to\infty} U_T(s) \tag{2.33}$$

$$(j = 1, 2, \ldots) \tag{2.34}$$

Proof. Let $H(s)$ be the Hamiltonian

$$H(s) = \sum_j E_j(s)\Pi_j(s) \tag{2.35}$$

that generates the unitary operator

$$U_T(s) = \mathbb{1} - i\int_0^s TH(\sigma)U(\sigma)d\sigma. \tag{2.36}$$

Consider the operator $K(s)$ defined by

$$K(s) = i\sum_j \frac{d\Pi_j}{ds}\Pi_j(s). \tag{2.37}$$

This operator is Hermitian, as can be verified using the product rule for and linearity of differentiation, and the Hermiticity, idempotency, and unit-summability

of projectors:

$$K(s)^\dagger = -i\sum_j \Pi_j(s)^\dagger \frac{d\Pi_j^\dagger}{ds} \tag{2.38}$$

$$= -i\sum_j \Pi_j(s)\frac{d\Pi_j}{ds} \tag{2.39}$$

$$= -i\sum_j \left(\frac{d(\Pi_j\Pi_j)}{ds} - \frac{d\Pi_j}{ds}\Pi_j\right) \tag{2.40}$$

$$= -i\sum_j \frac{d\Pi_j}{ds} + K(s) \tag{2.41}$$

$$= -i\frac{d}{ds}\sum_j \Pi_j + K(s) \tag{2.42}$$

$$= -i\frac{d}{ds}\mathbb{1} + K(s) \tag{2.43}$$

$$= K(s). \tag{2.44}$$

As $K(s)$ is Hermitian, it may be thought of as a Hamiltonian, which generates the unitary operator

$$A(s) = \mathbb{1} - i\int_0^s K(\sigma)A(\sigma)d\sigma. \tag{2.45}$$

By definitions (2.37) and (2.45), we see that

$$\frac{d}{ds}\left(A^\dagger\Pi_j A\right) = iA^\dagger K\Pi_j A + A^\dagger\frac{d\Pi_j}{ds}A - iA^\dagger\Pi_j K A \tag{2.46}$$

$$= A^\dagger\left(-\frac{d\Pi_j}{ds}\Pi_j + \frac{d\Pi_j}{ds} - \Pi_j\frac{d\Pi_j}{ds}\right)A \tag{2.47}$$

$$= A^\dagger\left(-\frac{d(\Pi_j\Pi_j)}{ds} + \Pi_j\frac{d\Pi_j}{ds} + \frac{d\Pi_j}{ds} - \Pi_j\frac{d\Pi_j}{ds}\right)A \tag{2.48}$$

$$= 0, \tag{2.49}$$

so that, in particular,

$$A^\dagger(s)\Pi_j(s)A(s) = \Pi_j(0). \tag{2.50}$$

Let $\Phi_T(s)$ be the unitary operator generated by $TA^\dagger(s)H(s)A(s)$:

$$\Phi_T(s) = \mathbb{1} - i\int_0^s TA^\dagger(\sigma)H(\sigma)A(\sigma)\Phi_T(\sigma)d\sigma. \tag{2.51}$$

This integral equation for $\Phi_T(s)$ may be solved with the help of eqs. (2.35) and (2.50):

$$\Phi_T(s) = \sum_j \exp\left[-iT\int_0^s E_j(\sigma)d\sigma\right] \Pi_j(0) \tag{2.52}$$

$$= \sum_j e^{-iT\varphi_j(s)}\Pi_j(0), \tag{2.53}$$

where I have introduced the following to simplify notation:

$$\varphi_j(s) = \int_0^s E_j(\sigma)d\sigma. \tag{2.54}$$

Consider the unitary operator

$$W(s) = \Phi_T^\dagger(s)A^\dagger(s)U_T(s). \tag{2.55}$$

By eqs. (2.36), (2.45) and (2.51), $W(s)$ obeys the integral equation

$$W(s) = \mathbb{1} + i\int_0^s \bar{K}(\sigma)W(\sigma)d\sigma, \tag{2.56}$$

where

$$\bar{K} \equiv \Phi_T^\dagger A^\dagger K A \Phi_T. \tag{2.57}$$

Integrated by parts, eq. (2.56) may be rewritten as

$$W(s) = \mathbb{1} + iF(s)W(s) + \int_0^s F(\sigma)\bar{K}(\sigma)W(\sigma)d\sigma, \tag{2.58}$$

where

$$F(s) \equiv \int_0^s \bar{K}(\sigma) d\sigma. \tag{2.59}$$

Using the solution (2.53) for $\Phi_T(s)$ and the conjugation property (2.50), $F(s)$ may be rewritten as

$$F(s) = \sum_{jk} \int_0^s e^{iT(\varphi_j - \varphi_k)} K_{jk}^{(A)}(\sigma) d\sigma, \tag{2.60}$$

where

$$K_{jk}^{(A)}(s) \equiv \Pi_j(0) A^\dagger(\sigma) K(\sigma) A(\sigma) \Pi_k(0) \tag{2.61}$$

$$= A^\dagger(\sigma) \Pi_j(\sigma) K(\sigma) \Pi_k(\sigma) A(\sigma). \tag{2.62}$$

The terms in the sum (2.60) when $j = k$ vanish because $\Pi_j K \Pi_j = 0$ by eq. (2.37). The remaining terms may be integrated by parts:

$$F(s) = \sum_{j \neq k} \frac{1}{iT} \left[e^{iT(\varphi_j - \varphi_k)} \frac{K_{jk}^{(A)}}{E_j - E_k} \bigg|_0^s - \int_0^s e^{iT(\varphi_j - \varphi_k)} \left[\frac{d}{d\sigma} \left(\frac{K_{jk}^{(A)}}{E_j - E_k} \right) \right] \right] d\sigma. \tag{2.63}$$

The terms in the outer brackets are independent of T and remain finite when conditions (i)–(iii) are satisfied, so

$$F(s) = \mathcal{O}\left(\frac{1}{T}\right). \tag{2.64}$$

Hence, eq. (2.58) for $W(s)$, when rewritten using eq. (2.55), reads

$$U_T(s) = A(s) \Phi_T(s) \left[1 + \mathcal{O}\left(\frac{1}{T}\right) \right]. \tag{2.65}$$

Applying the $\Phi_T(s)$ solution (2.53) and the conjugation property (2.50) to this

expression and taking the asymptotic limit, we obtain the desired result:

$$\lim_{T\to\infty} U_T(s)\Pi_j(0) = \Pi_j(s) \lim_{T\to\infty} U_T(s) \tag{2.66}$$

$$(j = 1, 2, \ldots) \tag{2.67}$$

■

2.7 The adiabatic approximation

The proof of the adiabatic theorem suggests that when T is large, it is reasonable to make the following *adiabatic approximation* to $U_T(s)$:

$$U_T(s) \simeq A(s)\Phi_T(s), \tag{2.68}$$

where $A(s)$ and $\Phi_T(s)$ are defined by (2.45) and (2.51). One way to view this approximation is as a simulation of the unitary $U_T(s)$, much in the same way that a quantum circuit is a simulation of a unitary operation. To understand the complexity of this simulation, it is important to quantify how good this approximation is. The *error distance* is one reasonable measure of its performance. (Other measures are certainly possible.) Operationally, this distance measures the probability that a measurement won't be able to distinguish between $U_T(1)\,|i\rangle$ and $A(1)\Phi_T(1)\,|i\rangle$, where $|i\rangle$ is an initial eigenstate of $H(0)$. This probability should be small when the approximation is good.

The error distance of the adiabatic approximation is

$$\Delta \equiv d_E(A(1)\Phi(1)\,|i\rangle,\, U_T(1)\,|i\rangle) \tag{2.69}$$

$$= 1 - |\,\langle i|\,\Phi_T^\dagger(1)A^\dagger(1)U_T(1)\,|i\rangle|^2 \tag{2.70}$$

$$= 1 - |\,\langle i|\,W\,|i\rangle|^2 \tag{2.71}$$

$$= \sum_j \langle i|\,W\,|j\rangle\langle j|\,W^\dagger\,|i\rangle - |\,\langle i|\,W\,|i\rangle|^2 \tag{2.72}$$

$$= \sum_{j\neq i} |\,\langle j|\,W\,|i\rangle|^2. \tag{2.73}$$

where $W \equiv W(1)$ is defined by eq. (2.55).

Because W can be expressed in terms of $F \equiv F(1)$ via eq. (2.58), and because $F = \mathcal{O}(1/T)$ by eq. (2.64), we can substitute the solution of eq. (2.58) into itself and obtain a power series for W in $1/T$. To first order in $1/T$, this expansion yields $W = \mathbb{1} + iF$. Introducing the notational shorthands $\omega_{ij} \equiv \varphi_i - \varphi_j$ and $\alpha_{ij} \equiv \langle i|\,K_{ij}^{(A)}\,|j\rangle$, and using eqs. (2.60) and (2.62) for F and $K^{(A)}$, the error distance Δ may be rewritten as follows:

$$\Delta = \sum_{j\neq i} |\,\langle j|\,W\,|i\rangle|^2 \tag{2.74}$$

$$= \sum_{j\neq i} |\,\langle j|\,F\,|i\rangle|^2 \tag{2.75}$$

$$= \sum_{j\neq i} \left| \int_0^1 ds\, \alpha_{ij}(s)\, e^{iT\omega_{ij}(s)} \right|^2. \tag{2.76}$$

The function $\alpha_{ij}(s)$ may be expressed in terms of $H'(s) \equiv dH/ds$ via the following identity:

$$\Pi_i \frac{dH}{ds} \Pi_j = \Pi_i \left(\sum_k E_k \frac{d\Pi_k}{ds} + \frac{dE_k}{ds}\Pi_k \right) \Pi_j \tag{2.77}$$

$$= \sum_k E_k \Pi_i \left[\frac{d}{ds}(\Pi_k\Pi_j) - \Pi_k \frac{d\Pi_j}{ds} \right] \Pi_j + \frac{dE_j}{ds}\delta_{ij}\Pi_i \tag{2.78}$$

$$= (E_j - E_i)\Pi_i \frac{d\Pi_j}{ds}\Pi_j + \frac{dE_j}{ds}\delta_{ij}\Pi_i. \tag{2.79}$$

Applying this identity, $\alpha_{ij}(s)$ becomes

$$\begin{aligned}
\alpha_{ij}(s) &= \langle i| K_{ij}^{(A)}(s) |j\rangle && (2.80)\\
&= \langle i| A^\dagger(s)\Pi_i(s)K(s)\Pi_j(s)A(s) |j\rangle && (2.81)\\
&= i\langle i| A^\dagger(s)\Pi_i(s)\sum_k \frac{d\Pi_k}{ds}\Pi_k(s)\Pi_j(s)A(s) |j\rangle && (2.82)\\
&= i\langle i| A^\dagger(s)\Pi_i(s)\frac{d\Pi_j}{ds}\Pi_j(s)A(s) |j\rangle && (2.83)\\
&= \frac{i}{E_j(s)-E_i(s)}\langle i| A^\dagger(s)\Pi_i(s)\frac{dH}{ds}\Pi_j(s)A(s) |j\rangle && (2.84)\\
&= \frac{i}{E_j(s)-E_i(s)}\langle i| \Pi_i(0)A^\dagger(s)H'(s)A(s)\Pi_j(0) |j\rangle && (2.85)\\
&= \frac{i}{E_j(s)-E_i(s)}\langle i| A^\dagger(s)H'(s)A(s) |j\rangle && (2.86)\\
&= i\frac{\langle i(s)| H'(s) |j(s)\rangle}{E_j(s)-E_i(s)}, && (2.87)
\end{aligned}$$

where $|i(s)\rangle$ denotes the eigenstate of $H(s)$ arrived at from the eigenstate $|i\rangle$ of $H(0)$ by continuity (*viz.*, by application of $A(s)$).

Although it is not entirely rigorous, Messiah [77] argues that the integral on the right-hand side of eq. (2.76) should have a value no greater in order-of-magnitude than the value it has when $\alpha_{ij}(s)$ and $\omega_{ij}(s)$ are independent of s. In other words, he argues that

$$\begin{aligned}
\Delta &\leq \sum_{j\neq i}\max_s|\alpha_{ij}(s)|^2\max_\sigma\frac{4\sin^2\omega_{ij}(\sigma)T/2}{T^2|\omega_{ij}(\sigma)|^2} && (2.88)\\
&\leq \frac{4}{T^2}\sum_{j\neq i}\frac{\max_s|\alpha_{ij}(s)|^2}{\min_\sigma|\omega_{ij}(\sigma)|^2}. && (2.89)
\end{aligned}$$

Taking this argument at face value, and introducing notational shorthands for the *minimum gap* and *maximal perturbation variance* in a manner similar to the

way it was introduced in Section 2.5,

$$g \equiv \min_{\sigma,j} |\omega(\sigma)_{ij}| \tag{2.90}$$

$$\Gamma^2 \equiv \max_s \langle i(s)| \left(\frac{dH}{ds}\right)^2 |i(s)\rangle - \langle i(s)| \frac{dH}{ds} |i(s)\rangle^2 \tag{2.91}$$

$$\equiv \Delta H' \text{ in state } |i(s)\rangle, \tag{2.92}$$

the bound on the error distance becomes

$$\Delta \leq \frac{4}{T^2 g^4} \max_s \sum_{j \neq i} \left| \langle i(s)| H'(s) |j(s)\rangle \right|^2 \tag{2.93}$$

$$= \frac{4}{T^2 g^4} \max_s \langle i(s)| H'(s)^2 |i(s)\rangle - \langle i(s)| H'(s) |i(s)\rangle^2 \tag{2.94}$$

$$\equiv \frac{4\Gamma^2}{T^2 g^4}, \tag{2.95}$$

which applies whenever the maximum over s can be taken outside the sum. (In general, this bound may not apply; only the weaker version with the maximization inside the sum applies.)

Thus, to ensure that $\Delta \ll 1$, we require that

$$T \gg \frac{\Gamma}{g^2}. \tag{2.96}$$

This is the central result I will use later in Chapter 5.

Chapter 3

Continuous-time quantum error correction

Abstract

In this chapter, I describe a new protocol for continuously protecting *unknown* quantum states from decoherence that incorporates design principles from both quantum error correction and quantum feedback control. This protocol uses continuous measurements and Hamiltonian operations, which are weaker control tools than are typically assumed for quantum error correction. A cost function appropriate for unknown quantum states is developed and used to optimize the state-estimate feedback. This protocol is studied in detail for the three-qubit bit-flip code by the use of Monte Carlo simulations. For this code, it is shown that the protocol improves the fidelity of quantum states beyond what is achievable using ordinary quantum error correction when the time between quantum error correction cycles is limited.

The work presented in this chapter is the result of a collaboration with Ahn and Doherty [5]. The single qubit example in 3.4.2 and the proof in 3.7 are due to Ahn.

3.1 Introduction

Long-lived coherent quantum states are essential for many quantum information science applications including quantum cryptography [15], quantum computation [80, 86], and quantum teleportation [16]. Unfortunately, coherent quantum states have extremely short lifetimes in realistic open quantum systems due to strong decohering interactions with the environment. Overcoming this decoherence is the chief hurdle faced by experimenters studying quantum-limited systems.

Quantum error correction is a "software solution" to this problem [92, 94]. It works by redundantly encoding quantum information across many quantum systems. The key to this approach is the use of measurements which reveal information about which errors have occurred and not about the encoded data. This feature is particularly useful for protecting the unknown quantum states that appear frequently in the course of quantum computations. The physical tools used in this approach are projective von Neumann measurements that discretize errors onto a finite set and fast unitary gates that restore corrupted data. When combined with fault-tolerant techniques, and when all noise sources are below a critical value known as the accuracy threshold, quantum error correction enables quantum computations of arbitrary length with arbitrarily small output error, or so-called fault-tolerant quantum computation [93, 60].

Quantum feedback control is also sometimes used to combat decoherence [111, 46, 97]. This approach has the advantage of working well even when control tools are limited. The information about the quantum state fed into the controller typically comes from continuous measurements and the operations the controller applies in response are typically bounded-strength Hamiltonians. The performance of the feedback may also be optimized relative to the resources that are available. For example, one can design a quantum feedback control scheme which minimizes the distance between a quantum state and its target subject to the constraint that all available controlling manipulations have bounded strengths [31].

The availability of quantum error correction, which can protect unknown quan-

tum states, and quantum feedback control, which uses weak measurements and slow controls, suggests that there might be a way to merge these approaches into a single technique with all of these features. Previous work to account for continuous time using quantum error correction has focused on "automatic" recovery and has neglected the role of continuous measurement [9, 25, 84, 10]. On the other hand, previous work on quantum state protection using quantum feedback control has focused on protocols for known states and has not addressed the issue of protecting unknown quantum states [106, 68]; however see [73] for related work.

This chapter is organized as follows. In Sec. 3.2, I review quantum feedback control and introduce the formalism of stochastic master equations. In Sec. 3.3, I present the three-qubit bit-flip code as a simple example of a quantum error-correcting code and sketch the general theory using the stabilizer formalism. In Sec. 3.4, I present a protocol for continuous-time quantum error correction as derived from an optimal non-Markovian feedback strategy. In Sec. 3.5, I demonstrate the efficacy of this feedback strategy for the bit-flip code via Monte Carlo simulations, and compare the behavior to discrete quantum error correction when the time between quantum error correction cycles is finite. Section 3.6 concludes.

3.2 Quantum feedback control

3.2.1 Open quantum systems

To describe quantum feedback control, we first need to describe uncontrolled open quantum system dynamics. Let $\mathcal{S}$ be an open quantum system weakly coupled to a reservoir $\mathcal{R}$, whose self-correlation time is much shorter than both the time scale of the system's dynamics and the time scale of the system-reservoir interaction. The Born-Markov approximation applies in this case and enables us to write down a *master equation* [20] describing the induced dynamics in $\mathcal{S}$:

$$\dot{\rho} = -i\,[H,\,\rho] + \sum_{\mu=1}^{m} \mathcal{D}[c_\mu]\,\rho. \tag{3.1}$$

Here ρ denotes the reduced density matrix for $\mathcal{S}$, H its Hamiltonian, and $\mathcal{D}$ a decohering Lindblad superoperator that takes a system-reservoir coupling operator (or *jump operator*) c as an argument and acts on density matrices as

$$\mathcal{D}[c]\rho = c\rho c^\dagger - \frac{1}{2}c^\dagger c\rho - \frac{1}{2}\rho c^\dagger c. \tag{3.2}$$

One way to derive this master equation is to imagine that the reservoir continuously measures the system but quickly forgets the outcomes because of rapid thermalization. The induced dynamics on $\mathcal{S}$ therefore appear as an average over all possible *quantum trajectories* that could have been recorded by the reservoir.

What kinds of measurements can the reservoir continuously perform? The most general measurement quantum mechanics allows is a positive operator-valued measure (POVM) $\{E_j\}$ acting on $\mathcal{S}$. According to a theorem by Kraus [69], the POVM $\{E_j\}$ can always be decomposed as

$$\sum_i \Omega_{ij}^\dagger \Omega_{ij} = E_j, \tag{3.3}$$

such that its stochastic action is $\rho \to \rho_j$ with probability $p_j = \mathrm{tr}(\rho E_j)$, where

$$\rho_j = \frac{1}{\mathrm{tr}(\rho E_j)} \sum_i \Omega_{ij}\rho\Omega_{ij}^\dagger. \tag{3.4}$$

This POVM is called a *strong measurement* when it can generate finite state changes and a *weak measurement* when it cannot [72]. POVMs that generate the master equation (3.1) involve infinitesimal changes of state, and therefore are weak measurements.

One reservoir POVM that results in the master equation (3.1) is the continuous weak measurement with Kraus operators

$$\Omega_0(dt) = 1 - \left(iH + \frac{1}{2}\sum_{\mu=1}^{m} c_\mu^\dagger c_\mu \right) dt \tag{3.5}$$

$$\Omega_\mu(dt) = \sqrt{dt}\, c_\mu, \qquad \mu = 1, \ldots, m. \tag{3.6}$$

Moreover, any POVM related to the one above via the unitary rotation

$$\Omega'_\alpha = \sum_\beta U_{\alpha\beta}\Omega_\beta \tag{3.7}$$

will generate the same master equation. We call each distinct POVM that generates the master equation when averaged over quantum trajectories an *unravelling* [112] of the master equation.

3.2.2 Quantum feedback control

The previous discussion of the master equation suggests a route for feedback control. If we replace the reservoir with a device that records the measurement current, then we could feed the measurement record back into the system's dynamics by way of a controller. For example, the master equation (3.1) with $m = 1$ can be unravelled into the *stochastic master equation* (SME) [20, 113]

$$\begin{aligned} d\rho_c(t) &= -i\,[H,\,\rho_c(t)]\,dt \\ &\quad +\mathcal{D}[c]\rho_c(t)dt + \mathcal{H}[c]\rho_c(t)dW(t) \qquad (3.8) \\ dQ(t) &= \langle c + c^\dagger \rangle_c\, dt + dW(t), \qquad (3.9) \end{aligned}$$

where ρ_c is the conditioned density matrix, conditioned on the outcomes of the measurement record $Q(t)$, the expectation $\langle a \rangle_c$ means $\mathrm{tr}(\rho_c a)$, dW is a normally distributed infinitesimal random variable with mean zero and variance dt (a *Wiener increment* [45]), and $\mathcal{H}$ is a superoperator that takes a jump operator as an argument and acts on density matrices as

$$\mathcal{H}[c]\rho = c\rho + \rho c^\dagger - \rho\ \mathrm{tr}\,[c\rho + \rho c^\dagger]. \tag{3.10}$$

This sort of unravelling occurs, for example, when one performs a continuous weak homodyne measurement of a field c by first mixing it with a classical local oscillator in a beamsplitter and then measuring the output beams with photodetec-

tors [113]. The stochastic model (3.8–3.9) is flexible enough to incorporate other noise sources such as detector inefficiency, dark counts, time delays, and finite measurement bandwidth [107].

We can add feedback control by introducing a $Q(t)$-dependent Hamiltonian to the dynamics of ρ_c. There are two well-studied ways of doing this. The first, and simplest, is to use Wiseman-Milburn feedback [111, 113], or *current feedback*, in which the feedback depends only on the instantaneous measurement current $I_Q(t) = dQ(t)/dt$. For example, adding the Hamiltonian $I_Q(t)F$ to the SME (3.8) using current feedback leads to the dynamics [111]

$$\begin{aligned} d\rho_c(t) &= -i\,[H,\,\rho_c(t)]\,dt \\ &\quad +\mathcal{D}[c]\rho_c(t)dt + \mathcal{H}[c]\rho_c(t)dW(t) \\ &\quad -i\left[F, c\rho_c(t) + \rho_c(t)c^\dagger\right]dt \\ &\quad +\mathcal{D}[F]\rho_c(t)dt - i\,[F,\rho_c(t)]\,dW \qquad (3.11) \\ dQ(t) &= \langle c + c^\dagger\rangle_c\,dt + dW(t). \qquad (3.12) \end{aligned}$$

The second, and more general, way to add feedback is to modulate the Hamiltonian by a functional of the entire measurement record. An important class of this kind of feedback is *estimate feedback* [31], in which feedback is a function of the current conditioned state estimate ρ_c. This kind of feedback is of especial interest because of the quantum Bellman theorem [30], which proves that the optimal feedback strategy will be a function only of conditioned state expectation values for a large class of physically reasonable cost functions. An example of such an estimate feedback control law analogous to the current feedback Hamiltonian used in (3.11) is to add the Hamiltonian $\langle I_Q(t)\rangle_c F = \langle c + c^\dagger\rangle_c F$, which depends on what we *expect* the current $I_Q(t)$ should be given the previous measurement history rather than its actual instantaneous value. Adding this feedback to the

SME (3.8) leads to the dynamics

$$\begin{aligned} d\rho_c(t) &= -i\,[H,\,\rho_c(t)]\,dt \\ &\quad +\mathcal{D}[c]\rho_c(t)dt + \mathcal{H}[c]\rho_c(t)dW(t) \\ &\quad -i\langle I_Q\rangle_c\,[F,\rho_c(t)]\,dt \qquad (3.13) \\ dQ(t) &= \langle c + c^\dagger\rangle_c\,dt + dW(t). \qquad (3.14) \end{aligned}$$

3.3 Quantum error correction

Although quantum feedback control has many merits, it has not been used to protect unknown quantum states from noise. Quantum error correction, however, is specifically designed to protect unknown quantum states; for this reason it has been an essential ingredient in the design of quantum computers [48, 66, 85]. The salient aspects of quantum error correction can already be seen in the three-qubit bit-flip code, even though it is not a fully quantum error correcting code. For that reason, I shall introduce quantum error correction with this example and discuss its generalization using the stabilizer formalism.

3.3.1 The bit-flip code

The bit-flip code protects a single two-state quantum system, or qubit, from bit-flipping errors by mapping it onto the state of three qubits:

$$\begin{aligned} |0\rangle &\rightarrow |000\rangle \equiv |\bar{0}\rangle \qquad (3.15) \\ |1\rangle &\rightarrow |111\rangle \equiv |\bar{1}\rangle. \qquad (3.16) \end{aligned}$$

The states $|\bar{0}\rangle$ and $|\bar{1}\rangle$ are called the *basis states* for the code and the space spanned by them is called the *codespace*, whose elements are called *codewords*.

After the qubits are subjected to noise, quantum error correction proceeds in two steps. First, the parities of neighboring qubits are projectively measured.

These are the observables[1]

$$M_0 = ZZI \tag{3.17}$$

$$M_1 = IZZ. \tag{3.18}$$

The *error syndrome* is the pair of eigenvalues (m_0, m_1) returned by this measurement.

Once the error syndrome is known, the second step is to apply one of the following unitary operations conditioned on the error syndrome:

$$(-1,+1) \rightarrow XII \tag{3.19}$$

$$(-1,-1) \rightarrow IXI \tag{3.20}$$

$$(+1,-1) \rightarrow IIX \tag{3.21}$$

$$(+1,+1) \rightarrow III. \tag{3.22}$$

This procedure has two particularly appealing characteristics: the error syndrome measurement does not distinguish between the codewords, and the projective nature of the measurement discretizes all possible quantum errors onto a finite set. These properties hold for general stabilizer codes as well.

If the bit-flipping errors arise from reservoir-induced decoherence, then prior to quantum error correction the qubits evolve via the master equation

$$d\rho_{\text{noise}} = \gamma(\mathcal{D}[XII] + \mathcal{D}[IXI] + \mathcal{D}[IIX])\rho\, dt, \tag{3.23}$$

where γdt is the probability of a bit-flip error on each qubit per time interval

[1] We use the notation of [48] in which X, Y, and Z denote the Pauli matrices σ_x, σ_y and σ_z respectively, and concatenation denotes a tensor product (e.g., $ZZI = \sigma_z \otimes \sigma_z \otimes I$).

$[t, t+dt]$. This master equation has the solution

$$\begin{aligned}\rho(t) = & \\ & a\,(t)\,\rho_0 \\ & +b\,(t)\,(XII\rho_0 XII + IXI\rho_0 IXI + IIX\rho_0 IIX) \\ & +c\,(t)\,(XXI\rho_0 XXI + XIX\rho_0 XIX + IXX\rho_0 IIX) \\ & +d\,(t)\,XXX\rho_0 XXX, \end{aligned} \tag{3.24}$$

where

$$a(t) = \left(1 + 3e^{-2\gamma t} + 3e^{-4\gamma t} + e^{-6\gamma t}\right)/8 \tag{3.25}$$

$$b(t) = \left(1 + e^{-2\gamma t} - e^{-4\gamma t} - e^{-6\gamma t}\right)/8 \tag{3.26}$$

$$c(t) = \left(1 - e^{-2\gamma t} - e^{-4\gamma t} + e^{-6\gamma t}\right)/8 \tag{3.27}$$

$$d(t) = \left(1 - 3e^{-2\gamma t} + 3e^{-4\gamma t} - e^{-6\gamma t}\right)/8. \tag{3.28}$$

The functions $a(t)$–$d(t)$ express the probability that the system is left in a state that can be reached by zero, one, two, or three bit-flips from the initial state, respectively. After quantum error correction is performed, single errors are identified correctly but double and triple errors are not. As a result, the recovered state, averaged over all possible measurement syndromes, is

$$\rho = (a\,(t) + b\,(t))\,\rho_0 + (c\,(t) + d\,(t))\,XXX\rho_0 XXX. \tag{3.29}$$

The overlap of this state with the initial state depends on the initial state, but is at least as large as when the initial state is $|\bar{0}\rangle$; namely, it is at least as large as

$$\begin{aligned} F_{\bar{3}} &= \left(2 + 3e^{-2\gamma t} - e^{-6\gamma t}\right)/4 \\ &\simeq 1 - 3(\gamma t)^2. \end{aligned} \tag{3.30}$$

Recalling that a single qubit subject to this decoherence has error probability

$p = \gamma t$, we see that, when applied sufficiently often, the bit-flip code reduces the error probability on each qubit from $\mathcal{O}(p)$ to $\mathcal{O}(p^2)$.

3.3.2 Stabilizer formalism

The bit-flip code is one of many quantum error correcting codes that can be described by the *stabilizer formalism* [48]. Let $\mathcal{C}$ be a 2^k-dimensional subspace of a 2^n-dimensional n-qubit Hilbert space. Then the system can be thought of as encoding k qubits in n, where the codewords are elements of $\mathcal{C}$. Let us further define the *Pauli group* to be $P_n = \{\pm 1, \pm i\} \otimes \{I, X, Y, Z\}^{\otimes n}$, and let the *weight* of an operator in P_n be the number of non-identity components it has when written as a tensor product of operators in P_1. The *stabilizer* of $\mathcal{C}$, $S(\mathcal{C})$, is the group of operators which fix all codewords in $\mathcal{C}$. We call $\mathcal{C}$ an $[[n, k, d]]$ *stabilizer code* when (a) the $n - k$ generators of $S(\mathcal{C})$ form a subgroup of P_n and (b) d is the smallest weight of an element in $P_n \setminus S(\mathcal{C})$ that commutes with every element in $S(\mathcal{C})$.

In this general setting, quantum error correction proceeds in two steps. First, one projectively measures the stabilizer generators to infer the error syndrome. Second, one applies a unitary recovery operator conditioned on the error syndrome. The strong measurement used in this procedure guarantees that all errors, even unitary errors, are discretized onto a finite set. For this reason I will sometimes refer to this procedure as discrete quantum error correction. When the noise rate is low and when correction is applied sufficiently often, this procedure reduces the error probability from $\mathcal{O}(p)$ to $\mathcal{O}(p^2)$.

3.4 Continuous quantum error correction via quantum feedback control

In this section, I present a method for continuously protecting an unknown quantum state using weak measurement, state estimation, and Hamiltonian correction. As in the previous section, this method is introduced via the bit-flip code and then generalized.

3.4.1 Bit-flip code: Theoretical model

Suppose ρ is subjected to bit-flipping decoherence as in (3.23); to protect against such decoherence, we have seen that we can encode ρ using the bit-flip code (3.15–3.16). Here we shall define a similar protocol that operates continuously and uses only weak measurements and slow corrections.

The first part of the protocol is to weakly measure the stabilizer generators ZZI and IZZ for the bit-flip code, even though these measurements will not completely collapse the errors. To localize the errors even further, we also measure the remaining nontrivial stabilizer operator ZIZ.[2] The second part of the protocol is to apply the slow Hamiltonian corrections XII, IXI, and IIX corresponding to the unitary corrections XII, IXI, and IIX, with control parameters λ_k that are to be determined. If we parameterize the measurement strength by κ and perform the measurements using the unravelling (3.8–3.9), the SME describing the protocol is

$$\begin{aligned} d\rho_c &= \gamma(\mathcal{D}[XII]+\mathcal{D}[IXI]+\mathcal{D}[IIX])\rho_c dt \\ &\quad +\kappa(\mathcal{D}[ZZI]+\mathcal{D}[IZZ]+\mathcal{D}[ZIZ])\rho_c dt \\ &\quad +\sqrt{\kappa}(\mathcal{H}[ZZI]dW_1+\mathcal{H}[IZZ]dW_2 \\ &\qquad +\mathcal{H}[ZIZ]dW_3)\rho_c \\ &\quad -i[F,\rho_c]dt \qquad (3.31) \\ dQ_1 &= 2\kappa\langle ZZI\rangle_c dt+\sqrt{\kappa}dW_1 \qquad (3.32) \\ dQ_2 &= 2\kappa\langle IZZ\rangle_c dt+\sqrt{\kappa}dW_2 \qquad (3.33) \\ dQ_3 &= 2\kappa\langle ZIZ\rangle_c dt+\sqrt{\kappa}dW_3, \qquad (3.34) \end{aligned}$$

where

$$F=\lambda_1 XII+\lambda_2 IXI+\lambda_3 IIX \qquad (3.35)$$

[2]The modest improvement gained by this extra measurement is offset by an unfavorable scaling in the number of extra measurements required when applied to general $[[n,k,d]]$ codes having 2^{n-k} stabilizer elements and only $n-k$ generators.

is the feedback Hamiltonian having control parameters λ_k.

Following the logic of quantum error correction, it is natural to choose the λ_k to be functions of the error syndrome. For example, the choice

$$\begin{aligned}
\lambda_1 &= \frac{\lambda}{8}(1-\langle ZZI\rangle_c)(1+\langle IZZ\rangle_c)(1-\langle ZIZ\rangle_c) \\
\lambda_2 &= \frac{\lambda}{8}(1-\langle ZZI\rangle_c)(1-\langle IZZ\rangle_c)(1+\langle ZIZ\rangle_c) \\
\lambda_3 &= \frac{\lambda}{8}(1+\langle ZZI\rangle_c)(1-\langle IZZ\rangle_c)(1-\langle ZIZ\rangle_c),
\end{aligned} \tag{3.36}$$

where λ is the maximum feedback strength that can be applied, is reasonable[3]: it acts trivially when the state is in the codespace and applies a maximal correction when the state is orthogonal to the codespace. Unfortunately this feedback is sometimes harmful when it need not be. For example, when the controller receives no measurement inputs (i.e., $\kappa = 0$), it still adds an extra coherent evolution which, on average, will drive the state of the system away from the state we wish to protect.

This weakness of the feedback strategy suggests that we should choose the feedback more carefully. To do this, we introduce a cost function describing how far away the state is from its target and choose a control which minimizes this cost. The difficulty is that the target is an *unknown* quantum state. However, we can choose the target to be the codespace, which we do know. We choose the cost function, therefore, to be the norm of the component of the state outside the codespace. Since the codespace projector is $\Pi_{\mathcal{C}} = \frac{1}{4}(III + ZZI + ZIZ + IZZ)$, the cost function is $1 - f$, where $f(\rho) = \mathrm{tr}(\rho\Pi_{\mathcal{C}})$. Under the SME (3.31), the time evolution of f due to the feedback Hamiltonian F is

$$\begin{aligned}
\dot{f}_{fb} &= 2\lambda_1\langle YZI + YIZ\rangle_c \\
&\quad +2\lambda_2\langle ZYI + IYZ\rangle_c \\
&\quad +2\lambda_3\langle ZIY + IZY\rangle_c .
\end{aligned} \tag{3.37}$$

[3]The factor of $\frac{1}{8}$ is included to limit the maximal strength of any parameter λ_k to λ.

Maximizing $\dot{f}_{fb}$ minimizes the cost, yielding the optimal feedback coefficients

$$\begin{aligned} \lambda_1 &= \lambda \operatorname{sgn}\langle YZI + YIZ\rangle_c \\ \lambda_2 &= \lambda \operatorname{sgn}\langle ZYI + IYZ\rangle_c \\ \lambda_3 &= \lambda \operatorname{sgn}\langle ZIY + IZY\rangle_c , \end{aligned} \tag{3.38}$$

where, again, λ is the maximum feedback strength that can be applied.

This feedback scheme is a *bang-bang* control scheme, meaning that the control parameters λ_k are always at the maximum or minimum value possible (λ or $-\lambda$, respectively), which is a typical control solution both classically [56] and quantum mechanically [102]. In practice, the bang-bang optimal controls (3.38) can be approximated by a bandwidth-limited sigmoid, such as a hyperbolic tangent function.

The control solution (3.38) requires the controller to integrate the SME (3.31) using the measurement currents $Q_i(t)$ and the initial condition ρ_c. However, typically the initial state $\rho_c(0)$ will be unknown. Fortunately the calculation of the feedback (3.38) does not depend on where the initial condition is within the codespace, so the controller may assume the maximally mixed initial condition $\rho_e = \frac{1}{2}(|\bar{0}\rangle\langle\bar{0}| + |\bar{1}\rangle\langle\bar{1}|)$ for its calculations. This property generalizes for a wide class of stabilizer codes, as is proved in Sec. 3.7; this property is conjectured to hold for all stabilizer codes.

3.4.2 Intuitive one-qubit picture

Before generalizing the procedure, it is helpful to gain some intuition about how it works by considering an even simpler "code": the spin-up state (i.e., $|0\rangle$) of a single qubit. The stabilizer is $M_0 = Z$, the noise it protects against is bit flips X, and the correction Hamiltonian is proportional to X. The optimal feedback, by a similar analysis to that for the bit-flip code, is $F = \lambda \operatorname{sgn}\langle Y\rangle_c X$, and the resulting stochastic master equation can be rewritten as a set of Bloch sphere equations as follows:

$$d\langle X\rangle_c = -2\gamma\langle X\rangle_c\,dt - 2\sqrt{\kappa}\langle X\rangle_c\langle Z\rangle_c\,dW \tag{3.39}$$

$$\begin{aligned} d\langle Y\rangle_c &= -2\gamma\langle Y\rangle_c\,dt - 2\kappa\langle Y\rangle_c - 2\sqrt{\kappa}\langle Y\rangle_c\langle Z\rangle_c\,dW \\ &\quad -2\lambda(\mathrm{sgn}\langle Y\rangle_c)\langle Z\rangle_c\,dt \end{aligned} \tag{3.40}$$

$$\begin{aligned} d\langle Z\rangle_c &= -2\gamma\langle Z\rangle_c\,dt + 2\sqrt{\kappa}(1-\langle Z\rangle_c^2)dW \\ &\quad +2\lambda(\mathrm{sgn}\langle Y\rangle_c)\langle Y\rangle_c\,dt. \end{aligned} \tag{3.41}$$

The Bloch vector representation $(\langle X\rangle, \langle Y\rangle, \langle Z\rangle)$ [85] of the qubit provides a simple geometric picture of evolution. Decoherence (the γ term) shrinks the Bloch vector, measurement (the κ terms) lengthens the Bloch vector and moves it closer to the z-axis, and correction (the λ term) rotates the Bloch vector in the y–z plane. Fig. 3.1 depicts this evolution: depending on whether the Bloch vector is in the hemisphere with $\langle Y\rangle > 0$ or $\langle Y\rangle < 0$, the feedback will rotate the vector as quickly as possible in such a way that it is always moving towards the codespace (spin-up state). Note that if the Bloch vector lies exactly on the z-axis with $\langle Z\rangle < 0$, rotating it either way will move it towards the spin-up state—the two directions are equivalent, and it suffices to choose one of them arbitrarily.

3.4.3 Feedback for a general code

This continuous feedback approach generalizes for a full $[[n, k, d]]$ quantum error-correcting code, which can protect against depolarizing noise [85] acting on each qubit independently. The depolarizing channel, unlike the bit-flip channel, generates a full range of quantum errors—it applies either X, Y, or Z to each qubit equiprobably at a rate γ. We weakly measure the $n-k$ stabilizer generators $\{M_l\}$ with strength κ. For each syndrome m, we apply a slow Hamiltonian correction F_m with control strength λ_m, the weight of each correction being d or less. The

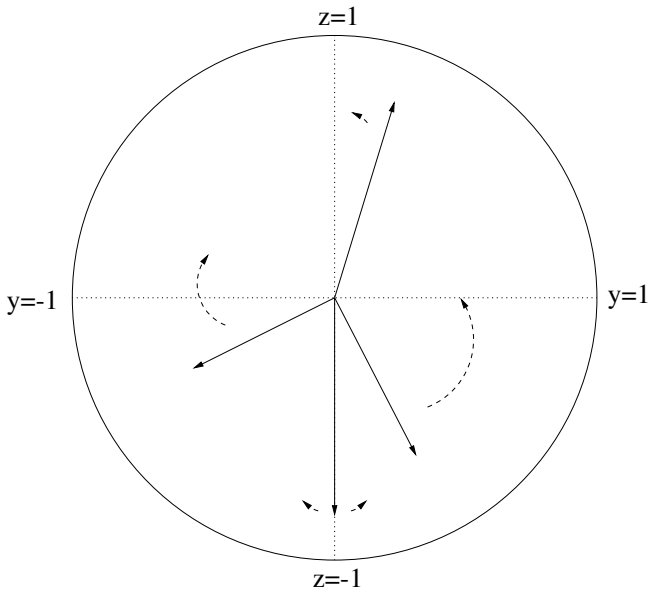

Figure 3.1: Bloch sphere showing the action of the feedback scheme on one qubit. Wherever the Bloch vector is in the y–z plane, the feedback forces it back to the spin-up state, which is the codespace of this system. All the vectors shown lie, without loss of generality, in the $x = 0$ plane.

SME describing this process is

$$\begin{aligned} d\rho_c &= \gamma \sum_{j=x,y,z} \sum_{i=1}^{n} (\mathcal{D}[\sigma_j^{(i)}])\rho_c dt + \kappa \sum_{l=1}^{n-k} \mathcal{D}[M_l]\rho_c dt \\ &\quad + \sqrt{\kappa} \sum_{l=1}^{n-k} \mathcal{H}[M_l] dW_j \rho_c - i \sum_{r=1}^{R} \lambda_r [F_r, \rho_c] dt. \end{aligned} \tag{3.42}$$

The number of feedback terms R needed will be less than or equal to the number of errors the code corrects against. The reason that this equality is not strict is that quantum error correcting codes can be *degenerate*, meaning that there can exist inequivalent errors that have the same effect on the state—a purely quantum mechanical property [48].

We optimize the λ_r relative to a cost function equal to the state's overlap with the codespace. For a general stabilizer code $\mathcal{C}$, the codespace projector is

$$\Pi_{\mathcal{C}} = \frac{1}{2^{n-k}} \prod_{l=1}^{n-k} (I + M_l)$$

and the rate of change of the codespace overlap due to feedback is

$$\dot{f}_{fb} = -i \ \mathrm{tr} \sum_{r=0}^{n-k} \lambda_r [\Pi_{\mathcal{C}}, F_r] \rho.$$

Maximizing this overlap subject to a maximum feedback strength λ yields the feedback coefficients

$$\lambda_r = \lambda \ \mathrm{sgn} \langle [\Pi_{\mathcal{C}}, F_r] \rangle_c \,. \tag{3.43}$$

This control solution, as for the bit-flip code, requires a controller to compute the feedback (3.43). A natural question to ask is how the scaling of the classical computation behaves. In Sec. 3.7, it is shown that the evolution of $(2^{n-k})^2$ parameters must be calculated in order to compute the feedback for an $[[n, k, d]]$ code, which at first does not seem promising. However, if one encodes mk qubits using m copies of an $[[n, k, d]]$ code, as might well be the case for a quantum memory, the SME (3.42) will not couple the dynamics of the m logical qubits; and, as in

the bit-flip case, the initial condition for the controller's integration can still be the completely mixed state in the total codespace. Then the relevant scaling for this system, the dependence on m, is linear: the number of parameters is $m(2^{n-k})^2$.

3.5 Simulation of the bit-flip code

In this section, I present the results of Monte Carlo simulations of the implementation of the protocol described in Section 3.4 for the bit-flip code.

3.5.1 Simulation details

Because the bit-flip code feedback control scheme (3.31–3.34) uses a nonlinear feedback Hamiltonian, numerical simulation is the most tractable route for its study. To obtain $\rho_c(t)$, these equations were integrated using a simple Euler integrator and a Gaussian random number generator. Stable convergent solutions were found when the dimensionless time step γdt was on the order of 10^{-6} and averaged over 10^4 quantum trajectories. As a benchmark, a typical run using these parameters took 2–8 hours on a 400 MHz Sun Ultra 2. More sophisticated Milstein [65] integrators were found to converge more quickly but required too steep a reduction in time step to achieve the same level of stability. All of these simulations began in the state $\rho_c(0) = |\bar{0}\rangle\langle\bar{0}|$, because it is maximally damaged by bit-flipping noise and therefore it yielded the most conservative results.

Two measures are used to assess the behavior of the bit-flip code feedback control scheme. The first measure is the *codeword fidelity* $F_{cw}(t) = \mathrm{tr}(\rho_c(0)\rho_c(t))$, the overlap of the state with the target codeword. This measure is appropriate when one cannot perform strong measurements and fast unitary operations, a realistic scenario for many physical systems. The quantity $F_{cw}(t)$ is compared to the fidelities of one unprotected qubit $F_1(t) = \frac{1}{2}(1+e^{-2\gamma t})$ and of three unprotected qubits $F_3(t) = (F_1(t))^3$.

The second measure is the *correctable overlap*

$$F_{corr}(t) = \mathrm{tr}(\rho_c(t)\Pi_{corr}), \tag{3.44}$$

where

$$\begin{aligned} \Pi_{corr} &= \rho_0 + XII\rho_0XII \\ &\quad + IXI\rho_0IXI + IIX\rho_0IIX \end{aligned} \tag{3.45}$$

is the projector onto the states that can be corrected back to the original codeword by discrete quantum error correction applied (once) at time t. This measure is appropriate when one can perform strong measurements and fast unitary operations, but only at discrete time intervals of length t. The quantity $F_{corr}(t)$ is compared to the fidelity $F_{\bar{3}}(t)$ obtained when, instead of using the protocol up to time t, no correction was performed until the final discrete quantum error correction at time t. As was shown in equation (3.30), the expression for $F_{\bar{3}}(t)$ may be calculated analytically; it is $F_{\bar{3}}(t) = \frac{1}{4}(2 + 3e^{-2\gamma t} - e^{-6\gamma t}) \sim 1 - 3\gamma^2 t^2$.

3.5.2 Results

The Monte Carlo simulations demonstrate that both the optimized estimate feedback scheme (3.38) and the heuristically motivated feedback scheme (3.36) effectively protect a qubit from bit-flip decoherence. Figs. 3.2 and 3.3 depict how these schemes behave for the (scaled) measurement and feedback strengths $\kappa/\gamma = 64$, $\lambda/\gamma = 128$ when averaged over 10^4 quantum trajectories. Using the first measure, one can see that at very short times, both schemes have codeword fidelities $F_{cw}(t)$ that follow the three-qubit fidelity $F_3(t)$ closely. For both schemes, $F_{cw}(t)$ improves and surpasses the fidelity of a single unprotected qubit $F_1(t)$. Indeed, perhaps the most exciting feature of these figures is that eventually $F_{cw}(t)$ surpasses $F_{\bar{3}}(t)$, the fidelity achievable by discrete quantum error correction applied at time t. In other words, continuous-time quantum error correction alone outperforms discrete

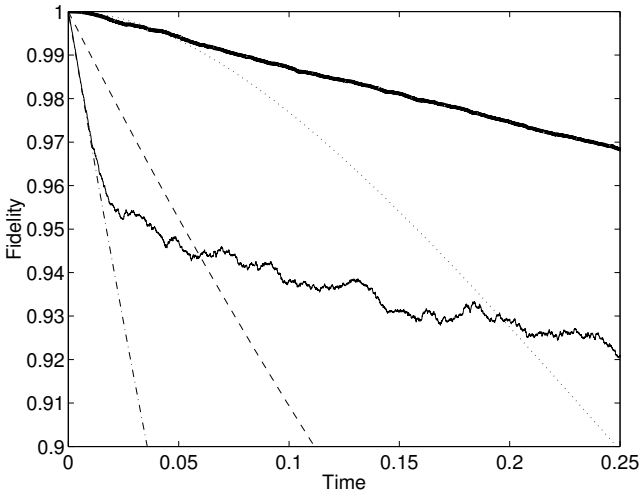

Figure 3.2: Behavior of the feedback protocol with optimized feedback (3.38) for parameters $\kappa/\gamma = 64$, $\lambda/\gamma = 128$, averaged over 10^4 quantum trajectories. The analytical curves shown are as follows: the dashed line is the fidelity of one decohering qubit, $F_1(t)$; the dashed-dotted line is the fidelity of three decohering qubits, $F_3(t)$; and the dotted line is the fidelity of an encoded qubit after one round of discrete error correction, $F_{\bar{3}}(t)$. The simulation results are as follows: the solid line is the codeword fidelity $F_{cw}(t)$, and the thick solid line is the correctable overlap $F_{corr}(t)$.

quantum error correction alone if the time between corrections is sufficiently long.

Looking at the second measure in Figs. 3.2 and 3.3, one can see that $F_{corr}(t)$ is as good as or surpasses $F_{\bar{3}}(t)$ almost everywhere. For times even as short as a tenth of a decoherence time, the effect of using (weak) continuous-time quantum error correction (CTQEC) between discrete quantum error correction cycles is quite noticeable. This improvement suggests that, even when one can approximate discrete quantum error correction but only apply it every so often, it pays to use CTQEC in between corrections. Therefore, CTQEC offers a means of improving the fidelity of a quantum memory even after the system has been isolated as well as possible and discrete quantum error correction is applied as frequently as possible.

There is a small time range from $t \cong 0.01$ to $t \cong 0.05$ for the parameters

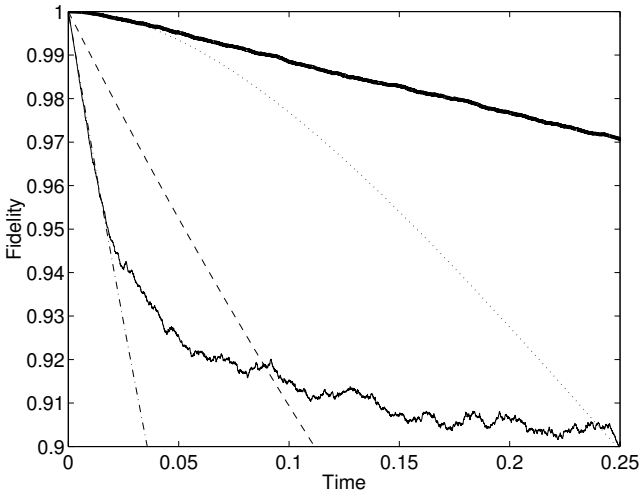

Figure 3.3: Behavior of continuous-time quantum error correction with feedback (3.36) for parameters $\kappa/\gamma = 64$, $\lambda/\gamma = 128$, averaged over 10^4 quantum trajectories. As in Fig. 3.2, the dashed line is $F_1(t)$, the dashed-dotted line is $F_3(t)$, the dotted line is $F_{\bar{3}}(t)$, the solid line is $F_{cw}(t)$ and the thick solid line is $F_{cw}(t)$. Note that this feedback is qualitatively similar to that in Fig. 3.2 but does not perform as well.

used in Fig. 3.2 in which using CTQEC before discrete quantum error correction actually underperforms not doing anything before the correction. The simulations suggest that the reason for this narrow window of deficiency is that, in the absence of CTQEC, it is possible to have two errors on a qubit (e.g., two bit flips) that cancel each other out before discrete quantum error correction is performed. In contrast, CTQEC will immediately start to correct for the first error before the second one happens, so the advantage of this sort of cancellation is lost. This view is supported by the fact that $F_{corr}(t)$ in the simulations always lies above the fidelity line obtained by subtracting such fortuitous cancellations from $F_{\bar{3}}(t)$. In any case, this window can be made arbitrarily small and pushed arbitrarily close to the beginning of CTQEC by increasing the measurement strength κ and the feedback strength λ.

In Figs. 3.2 and 3.3, the $F_{cw}(t)$ line is much more jagged than the $F_{corr}(t)$ line.

The jaggedness in both of these lines is due to statistical noise in the simulation and is reduced when averaged over more than 10^4 trajectories. The reason for the reduced noise in the $F_{corr}(t)$ line has to do with the properties of discrete quantum error correction—on average, neighboring states get corrected back to the same state by discrete quantum error correction, so noise fluctuations become smoothed out.

The improvement the optimized estimate CTQEC feedback protocol yields beyond the heuristically motivated CTQEC feedback protocol is more noticeable in $F_{cw}(t)$ than in $F_{corr}(t)$ as seen in Figs. 3.2 and 3.3. The optimized protocol acts to minimize the distance between the current state and the codespace, not between the current state and the space of states correctable back to the original codeword, so this observation is perhaps not surprising. In fact, optimizing feedback relative to $F_{corr}(t)$ is not even possible without knowing the codeword being protected. Nevertheless, the optimized protocol does perform better, so henceforth I shall restrict my discussion to it.

How CTQEC behaved when the scaled measurement strength κ/γ and feedback strength λ/γ were varied was also studied using the two measures described in Sec. 3.5.1. The first measure, the codeword fidelity $F_{cw}(t)$, crosses the unprotected qubit fidelity $F_1(t)$ at various times τ as depicted in Fig. 3.4. This time is of interest because it is the time after which the optimized protocol improves the fidelity of a qubit beyond what it would have been if it were left to itself. Increasing the scaled feedback strength λ/γ improves the CTQEC scheme and reduces τ, but the dependence on the scaled measurement strength κ/γ is not so obvious from Fig. 3.4.

By looking at cross sections of Fig. 3.4, such as at $\lambda/\gamma = 80$ as in Fig. 3.5, one can see that, for a given scaled feedback strength λ/γ, there is a minimum crossing time τ as a function of measurement strength κ/γ. In other words, there is an optimal choice of measurement strength κ/γ. This optimal choice arises because syndrome measurements, which localize states near error subspaces, compete with Hamiltonian correction operations, which coherently rotate states between the

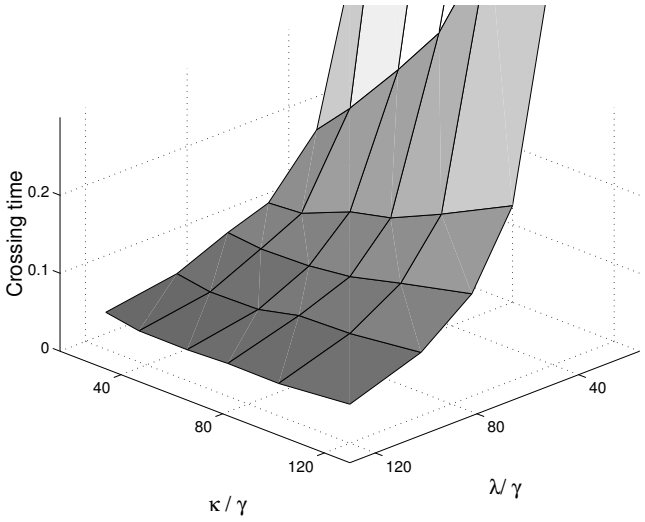

Figure 3.4: Time τ at which $F_{cw}(\tau) = F_1(\tau)$ as a function of measurement strength κ/γ and feedback strength λ/γ. This crossing time is the time after which the optimized continuous-time quantum error correction protocol improves the fidelity of a qubit beyond what it would have been if it were left to itself.

nontrivial error subspaces to the trivial error subspace. This phenomenon is a feature of continuous-time quantum error correction that is not present in discrete quantum error correction; in the former, measurement and correction are simultaneous, while in the latter, measurement and correction are separate processes that don't interfere.

In order to study how the second measure, the correctable overlap $F_{corr}(t)$, varies with κ and λ, it is instructive to examine its behavior at a particular time. Fig. 3.6 plots $F_{corr}(t)$, evaluated at the time $t = 0.2/\gamma$, as a function of κ and λ. As was found with the crossing time τ, increasing λ always improves performance, but increasing κ does not because measurement can compete with correction. Since $F_{\bar{3}}(0.2/\gamma) \cong 0.927$, for all the κ and λ plotted in Fig. 3.6, using CTQEC between discrete quantum error correction intervals of time $0.2/\gamma$ improves the reliability of the encoded data.

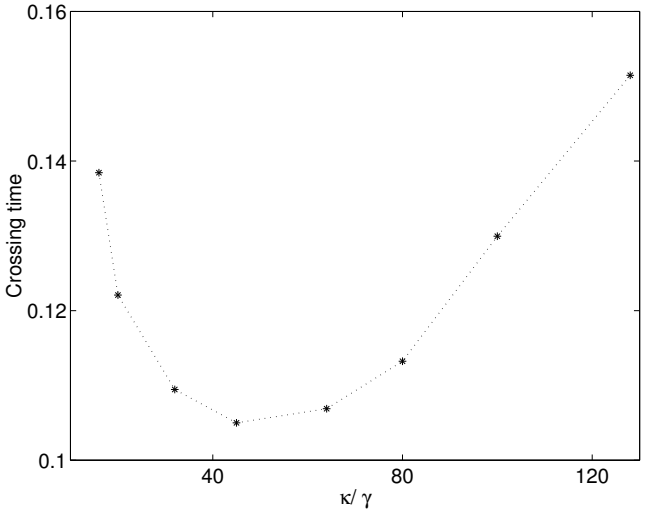

Figure 3.5: Time τ at which $F_{cw}(\tau) = F_1(\tau)$ as a function of measurement strength κ/γ, keeping correction strength fixed at $\lambda/\gamma = 80$.

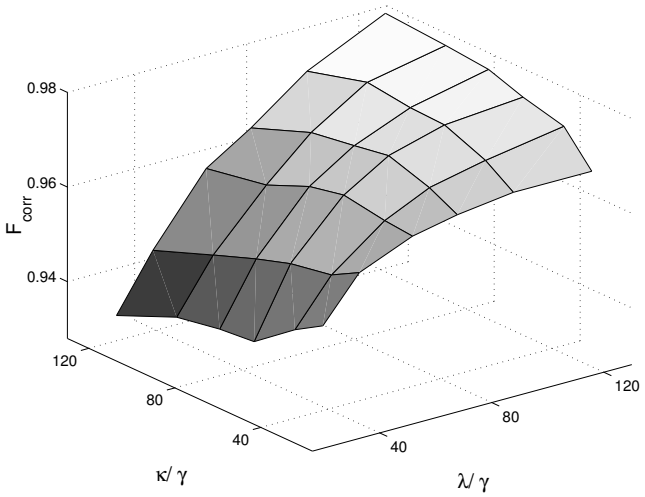

Figure 3.6: F_{corr} at $\gamma t = 0.2$ as a function of measurement strength κ/γ and feedback strength λ/γ. This quantity corresponds to the fidelity of a state given continuous error-correction up to $\gamma t = 0.2$, at which point discrete error-correction is performed.

3.6 Conclusion

Often, in realistic quantum computing architectures, weak measurements and Hamiltonian operations are likely to be the tools available to protect quantum states from decoherence. Moreover, even quantum systems in which strong measurements and fast operations are well approximated, such as ion traps [110], it is likely that these operations will only be possible at some maximum rate. Continuous-time quantum error correction (CTQEC) is able to continuously protect unknown quantum states using only weak measurements and Hamiltonian corrections and can improve the fidelity of quantum states beyond rate-limited quantum error correction. In addition, because CTQEC responds to the entire measurement record and not to instantaneous measurement results, it will not propagate errors badly and therefore has a limited inherent fault-tolerance that ordinary quantum error correction does not.

Continuous-time quantum error correction is expected to be applicable to other continuous-time quantum information processes, such as reliable state preparation and fault-tolerant quantum computation. It is also expected that this protocol will work when different continuous-time measurement tools are available, such as direct photodetection. Finally, although current computing technology has limited investigation by simulation to few-qubit versions of CTQEC, it is expected that many of the salient features found in the three-qubit bit-flip code example will persist when CTQEC is applied to larger codes.

3.7 Feedback based on the completely mixed state

Although the CTQEC scheme described in Section 3.4 does not distinguish between codewords, it is not obvious that in order to use it one does not need to first know the initial codeword to integrate the SME and calculate the relevant expectation values. Since one of the primary selling points of CTQEC is that it can protect unknown quantum states, this property is crucial to the scheme's success.

Fortunately, for a large class of stabilizer codes, the computation of the feedback can be done by assuming the initial state is the completely mixed codespace state $\rho_e = \frac{1}{2^n}\prod_{l=1}^{n-k}(I + M_l)$, which I prove here. The proof is originally due to Ahn [5].

Defining the set G for the $[[n, k, d]]$ code $\mathcal{C}$ with stabilizer $S(\mathcal{C})$ as

$$G = \{\alpha s \,|\, \alpha \in P_n, s \in S(\mathcal{C}), [s, \alpha] = 0 \text{ iff } |\alpha| \text{ is even}\}, \tag{3.46}$$

where $|\alpha|$ denotes the weight of α as defined in section 3.3.2.

The normalizer $N(S)$ for the code is defined to be the group of operators that commute with every element in $S(\mathcal{C})$. The elements of $N(S) \setminus S$ can be thought of as the *encoded operations* for the code—they move one codeword to another.

Let $g = \sigma_{i_1} \otimes ... \otimes \sigma_{i_n}$, where $i_1 \ldots i_n$ take on the values x, y, z, I and $\sigma_I = I$. Define the *Pauli basis coefficients* $R_g(\rho)$ of a density matrix ρ as follows:

$$R_g(\rho) \equiv \mathrm{tr}(\rho g)/2^n = \langle g \rangle / 2^n, \tag{3.47}$$

The following theorem shows that the conditions for the feedback to be insensitive to the initial codeword can be expressed as

1. For every R_g used in CTQEC, $g \in G$.
2. For every $g \in G$ and every ρ_1 and ρ_2 in $\mathcal{C}$, $R_g(\rho_1) = R_g(\rho_2)$.
3. Evolution under the SME couples members of the set $\{R_g | g \in G\}$ only to each other.

Theorem 3.7.1. *Let $\mathcal{C}$ be an $[[n, 1, 3]]$ [4] stabilizer code whose stabilizer $S(\mathcal{C})$ has generators of only even weight and whose encoded operations set $N(S) \setminus S$ has elements of only odd weight.*[5] *Then the conditions 1–3 above are satisfied; conse-*

[4] The restriction to $[[n, 1, 3]]$ codes is for simplicity of analysis; the proof may be extended to larger codes. Note that for an $[[n, 1, 3]]$ code, the F_l in the master equation (3.42) are all of the form $\sigma_j^{(k)}$, where this notation denotes the weight-one Pauli operator σ_j acting on qubit k.

[5] It is possible that this restriction may be able to be relaxed; however, it is sufficiently general that it holds for the most well-known codes, including the bit-flip code, the five-bit code, the

quently, continuous-time quantum error correction does not require knowledge of where the initial codeword lies in $\mathcal{C}$.

Proof. In this proof, any variable of the form α_a is an arbitrary element of P_n, and any variable of the form s_a is an arbitrary element of $S(\mathcal{C})$. Each of the conditions listed above are proven separately.

Condition 1: By construction, G contains all M of the form $M = s_i\sigma_j^{(k)}$, where $[s_i, \sigma_j^{(k)}] \neq 0$. These are precisely the operators used to compute the feedback in (3.43) for a code encoding one qubit.

Condition 2: Let $g = \alpha s \in G$ and let $\rho \in \mathcal{C}$. We know either $\alpha \in S$, $\alpha \in N(S) \setminus S$, or $\alpha \notin N(S)$. Suppose $\alpha \in S$. Then $g \in S$ acts trivially on all states in the codespace, so $R_g = 1/2^n \operatorname{tr}(\rho g) = 1/2^n$ for this case. Now suppose $\alpha \in N(S) \setminus S$. Then $[\alpha, s] = 0$, and since $\alpha s \in G$, $|\alpha|$ is even. But every element of $N(S) \setminus S$ has odd weight by hypothesis, which is a contradiction. Hence α cannot be in $N(S) \setminus S$. Finally, suppose $\alpha \notin N(S)$. Then there exists some $s' \in S$ such that $[\alpha, s'] \neq 0$; let s' be such an element. Then for $|\psi\rangle, |\phi\rangle \in \mathcal{C}$,

$$\begin{aligned} \langle\psi|\,\alpha\,|\phi\rangle &= \langle\psi|\,\alpha s'\,|\phi\rangle = -\langle\psi|\,s'\alpha\,|\phi\rangle \\ &= -\langle\psi|\,\alpha\,|\phi\rangle = 0. \end{aligned} \tag{3.48}$$

Hence for this case $R_g = 1/2^n \operatorname{tr}(\rho\alpha s) = 0$. Note that these expressions for R_g must be the same no matter where ρ is in the codespace; therefore, for every $g \in G$ and $\rho_1, \rho_2 \in \mathcal{C}$, $R_g(\rho_1) = R_g(\rho_2)$.

Condition 3: Consider dR_M, where $M \in G$. It will be shown that $dR_M = f(\{R_N | N \in G\})$ for some real function f. For any $M \in P_n$, $dR_M = \operatorname{Tr}(d\rho\ M)$, where $d\rho$ is given by the master equation (3.42). Hence the satisfaction of condition three can be demonstrated for each term of the master equation separately. First, substituting in the master equation shows that any term of the form $\mathcal{D}[c]\rho dt$

Steane code, and the nine-bit Shor code. This condition also ensures that G is *consistent*, i.e., if $\alpha_j s_k \in G$ and $\alpha_j = \alpha_n s_m$, then α_n and $s_m s_k$ also fulfill the conditions for $\alpha_n(s_m s_k)$ to be in G.

contributes either 0 or the simple exponential damping term $-2R_M$ to dR_M if M and c commute or anticommute, respectively.

As for the master equation term $\mathcal{H}[s_j]dW_j\rho$, by writing the master equation in the Pauli basis one can see that R_N contributes to dR_M through this term precisely when $Ns_j = M$ and $\{s_j, N\} \neq 0$. Since $M \in G$, one may write $M = \alpha_k s_l$ (with the appropriate restriction on $[\alpha_k, s_l]$ depending on the weight of α_k) . $N = \alpha_k s_l s_j = \alpha_k s_m$, so the condition above that $[s_j, N] = 0$ becomes $[s_j, \alpha_k s_l s_j] = (\alpha_k[s_j, s_l s_j] + [s_j, \alpha_k]s_l s_j) \Rightarrow [s_j, \alpha_k] = 0$. Therefore, $[\alpha_k, s_m] = s_l[\alpha_k, s_j] + [\alpha_k, s_l]s_j = [\alpha_k, s_l]s_j$ which is zero or not depending on the original weight of α_k. So if $M = \alpha_k s_l$ is such that $M \in G$, $N = \alpha_k s_m$ must fulfill that same condition, implying that $N \in G$ also.

Similarly, R_N contributes to dR_M through the master equation term $[\sigma_j^{(k)}, \rho]$ when $N\sigma_j^{(k)} = M$ and $[\sigma_j^{(k)}, N] \neq 0$. Now, $M \in G$ so $M = \alpha_l s_m$, again with the appropriate restriction on $[\alpha_l, s_m]$ depending on the weight of α_l. Then $N = \sigma_j^{(k)}\alpha_l s_m \equiv \alpha_n s_m$, so the condition above that $\{\sigma_j^{(k)}, N\} \neq 0$ becomes

$$\begin{aligned} \{\sigma_j^{(k)}, \sigma_j^{(k)}\alpha_l s_m\} &= \sigma_j^{(k)}[\sigma_j^{(k)}, \alpha_l]s_m + \sigma_j^{(k)}\alpha_l\{\sigma_j^{(k)}, s_m\} \\ &= \sigma_j^{(k)}\{\sigma_j^{(k)}, \alpha_l\}s_m - \sigma_j^{(k)}\alpha_l[\sigma_j^{(k)}, s_m] \\ &= 0. \end{aligned} \tag{3.49}$$

The analysis of this term can be divided into two cases. Case 1 occurs when $\sigma_j^{(k)}\alpha_l$ has weight $|\alpha_l|$, implying that $\{\alpha_l, \sigma_j^{(k)}\} = 0$. Then $\{\sigma_j^{(k)}, \sigma_j^{(k)}\alpha_l s_m\} = -\sigma_j^{(k)}\alpha_l[\sigma_j^{(k)}, s_m] = 0$, which implies that $[s_m, \alpha_n] = [s_m, \sigma_j^{(k)}]\alpha_l + \sigma_j^{(k)}[s_m, \alpha_l] = \sigma_j^{(k)}[s_m, \alpha_l]$. So $[s_m, \alpha_n] = 0$ just when $[s_m, \alpha_l] = 0$, which means that $N \in G$ since $|\alpha_n| = |\alpha_l|$.

In Case 2, $\sigma_j^{(k)}\alpha_l$ has weight $|\alpha_l \pm 1| \Rightarrow [\alpha_l, \sigma_j^{(k)}] = 0$. Then (3.49) becomes $\{\sigma_j^{(k)}, \sigma_j^{(k)}\alpha_l s_m\} = \sigma_j^{(k)}\alpha_l\{\sigma_j^{(k)}, s_m\} = 0$, which implies that $[s_m, \alpha_n] = \{s_m, \sigma_j^{(k)}\}\alpha_l + \sigma_j^{(k)}\{s_m, \alpha_l\} = \sigma_j^{(k)}\{s_m, \alpha_l\}$. So $[s_m, \alpha_n] = 0$ just when $\{s_m, \alpha_l\} = 0$, which means that $N \in G$ since $|\alpha_n| = |\alpha_l \pm 1|$. ■

In summary, the theorem demonstrates that all three of the conditions above are satisfied: all the R's used to compute the feedback are of the form $R_{N\in G}$; for a given $M \in G$, R_M will be the same for any state in the codespace; and evolution via the master equation mixes the R's of the form $R_{N\in G}$ only with each other. Therefore, CTQEC works the same for any state initially in the codespace, including the true initial state and the entirely mixed state, so it suffices to presuppose the completely mixed state as the initial condition rather than the actual (unknown) code state.

Another consequence of using the completely mixed state for feedback arises from the fact that doing so corresponds to discarding information about the state of the system. Therefore, this procedure should reduce the number of parameters needed to compute the feedback. Unfortunately, this only leads to a modest reduction in the number of parameters, which can be found by using a simple counting argument. There are $2^n/2^k = 2^{n-k}$ different error subspaces, including the no-error (code) space, and if one starts with the completely mixed state in the codespace one does not need to worry at all about any movement within any of these spaces. One only needs to worry about which error space the state is actually in, along with coherences between these spaces, so that $(2^{n-k})^2$ parameters are needed to describe the system.

Chapter 4

Topological quantum memory

Abstract

In this chapter, I present an analysis of *surface codes*, the topological quantum error-correcting codes introduced by Kitaev. In these codes, qubits are arranged in a two-dimensional array on a surface of nontrivial topology, and encoded quantum operations are associated with nontrivial homology cycles of the surface. I present new protocols for error recovery, and study the efficacy of these protocols. An order-disorder phase transition occurs in this system at a nonzero critical value of the error rate; if the error rate is below the critical value (the *accuracy threshold*), encoded information can be protected arbitrarily well in the limit of a large code block. This phase transition can be accurately modelled by a three-dimensional $\mathbb{Z}_2$ lattice gauge theory with quenched disorder. I present an estimation of the accuracy threshold, assuming that all quantum gates are *local*, that qubits can be measured rapidly, and that polynomial-size classical computations can be executed instantaneously. I also describe a robust recovery procedure that does not require measurement or fast classical processing; however, for this procedure the quantum gates are local only if the qubits are arranged in *four* or more spatial dimensions. I present procedures for encoding, measurement, and performing fault-tolerant universal quantum computation with surface codes, and argue that these codes provide a promising framework for quantum computing architectures.

The work presented in this chapter is the result of a collaboration with Dennis, Kitaev, and Preskill [29]. Large sections of this chapter were originally written by Preskill.

4.1 Introduction

The microscopic world is quantum mechanical, but the macroscopic world is classical. This fundamental dichotomy arises because a coherent quantum superposition of two readily distinguishable macroscopic states is highly unstable. The quantum state of a macroscopic system rapidly *decoheres* due to unavoidable interactions between the system and its surroundings.

Decoherence is so pervasive that it might seem to preclude subtle quantum interference phenomena in systems with many degrees of freedom. However, recent advances in the theory of quantum error correction suggest otherwise [92, 94]. Quantum states can be cleverly encoded so that the debilitating effects of decoherence, if not too severe, can be resisted. Furthermore, fault-tolerant protocols have been devised that allow an encoded quantum state to be reliably processed by a quantum computer with imperfect components [93]. In principle, then, very intricate quantum systems can be stabilized and accurately controlled.

The theory of quantum fault tolerance has shown that, even for delicate coherent quantum states, information *processing* can prevent information *loss*. In this chapter, we will study a particular approach to quantum fault tolerance that has notable advantages: in this approach, based on the *surface codes* introduced in [60, 61], the quantum processing needed to control errors has especially nice locality properties. Hence, surface codes suggest a particularly promising approach to quantum computing architecture.

One glittering achievement of the theory of quantum fault tolerance is the *threshold theorem*, which asserts that an arbitrarily long quantum computation can be executed with arbitrarily high reliability, provided that the error rates of the computer's fundamental quantum gates are below a certain critical value, the

accuracy threshold [67, 2, 62, 87, 48]. The numerical value of this accuracy threshold is of great interest for future quantum technologies, as it defines a standard that should be met by designers of quantum hardware. The critical error probability per gate p_c has been estimated as $p_c \gtrsim 10^{-4}$; very roughly speaking, this means that robust quantum computation is possible if the decoherence time of stored qubits is at least 10^4 times longer than the time needed to execute one fundamental quantum gate [52], assuming that decoherence is the only source of error.

This estimate of the accuracy threshold is obtained by analyzing the efficacy of a *concatenated code*, a hierarchy of codes within codes, and it is based on many assumptions, which will be elaborated in Sec. 4.2. Some of these assumptions are less realistic than others. For example, one assumption is that a quantum gate can act on any pair of qubits, with a fidelity that is independent of the spatial separation of the qubits. This assumption is clearly unrealistic; it is made because it greatly simplifies the analysis. Thus this estimate will be reasonable for a practical device only to the extent that the hardware designer is successful in arranging that qubits that must interact are kept close to one another. It is known that the threshold theorem still applies if quantum gates are required to be local [2, 51], but for this realistic case careful estimates of the threshold have not been carried out.

In this chapter, I will perform a quite different estimate of the accuracy threshold, based on surface codes rather than concatenated codes. This estimate applies to a device with strictly local quantum gates, if the device is controlled by a classical computer that is perfectly reliable, and whose clock speed is much faster than the clock speed of the quantum computer. In this approach, some spatial nonlocality in effect is still allowed, but all the nonlocal processing is demanded to be classical. Specifically, an error syndrome is extracted by performing local quantum gates and measurements; then a classical computation is executed to infer what quantum gates are needed to recover from error. We will assume that this classical computation, which actually requires a time bounded above by a polynomial in

the number of qubits in the quantum computer, can be executed in a constant number of time steps. Under this assumption, the existence of an accuracy threshold can be established and its value can be estimated. It will be shown that, under the assumption that the classical computation can be completed in a single time step, the critical error probability p_c per qubit and per time step satisfies $p_c \geq 1.7 \times 10^{-4}$. This estimate applies to the accuracy threshold for reliable *storage* of quantum information, rather than for reliable processing. The threshold for quantum computation is not as easy to analyze definitively, but it will be argued that its numerical value is not likely to be substantially different.

It is reasonable to believe that the principles of fault tolerance will dictate the shape of future quantum computing architectures. In Sec. 4.2, the hardware features that are conducive to fault-tolerant processing will be listed, and the design of a fault-tolerant quantum computer that incorporates surface coding will be outlined. I review the properties of surface codes in Sec. 4.3, emphasizing in particular that the qubits in the code block can be arranged in a *planar* sheet [17, 42], and that errors in the syndrome measurement complicate the recovery procedure. The core of the chapter is Sec. 4.4, where recovery from errors using surface codes is related to a statistical-mechanical model with local interactions. In the (unrealistic) case where syndrome measurements are perfect, this model becomes the two-dimensional Ising model with quenched disorder, whose phase diagram has been studied by Monte Carlo simulations. These simulations indicate that if the syndrome information is put to optimal use, error recovery succeeds with a probability that approaches one in the limit of a large code block, if and only if both phase errors and bit-flip errors occur with a probability per qubit less than about 11%. In the more realistic case where syndrome measurements are imperfect, error recovery is modelled by a three-dimensional Z_2 gauge theory with quenched disorder, whose phase diagram (to the best knowledge of me and my collaborators) has not been studied previously. The third dimension that arises can be interpreted as time—since the syndrome information cannot be trusted, one must repeat the measurement many times before one can be confident about

the correct way to recover from the errors. It will be argued that an order-disorder phase transition of this model corresponds to the accuracy threshold for quantum storage, and furthermore that the optimal recovery procedure can be computed efficiently on a classical computer. In Sec. 4.5, a rather crude lower bound on the accuracy threshold will be proved, concluding that error recovery procedure is sure to succeed in the limit of a large code block under suitable conditions: for example, if in each round of syndrome measurement, qubit phase errors, qubit bit-flip errors, and syndrome bit errors all occur with probability below 1.14%. Tighter estimates of the accuracy threshold could be obtained through numerical studies of the quenched gauge theory.

In deriving this accuracy threshold for quantum storage, it is assumed that an unlimited amount of syndrome data could be deposited in a classical memory, if necessary. But in Sec. 4.6, it will be shown that this threshold, and a corresponding accuracy threshold for quantum computation, remain intact even if the classical memory is limited to polynomial size. Then in Sec. 4.7, quantum circuits for syndrome measurement are analyzed, so that the estimate of the accuracy threshold can be reexpressed as a fidelity requirement for elementary quantum gates. The conclusion is that such a quantum memory can resist decoherence if gates can be executed in parallel, and if the qubit decoherence time is at least 6000 times longer than the time needed to execute a gate. In Sec. 4.8, it will be shown that encoded qubits can be accurately prepared and reliably measured. It will also be shown how a surface code with a small block size can be built up gradually to a large block size; this procedure allows one to enter a qubit in an unknown quantum state into this quantum memory with reasonable fidelity, and then to maintain that fidelity for an indefinitely long time. It will be explained in Sec. 4.9 how a universal set of quantum gates acting on protected quantum information can be executed fault-tolerantly.

Most of the analysis of the accuracy threshold in this chapter is premised on the assumption that qubits can be measured quickly and that classical computations can be done instantaneously and perfectly. In Sec. 4.10, these assumptions are

dropped. A recovery procedure that does not require measurement or classical computation will be presented, and a lower bound on the accuracy threshold will be inferred. Unfortunately, though, the quantum processing in this procedure is not spatially local unless the dimensionality of space is at least four. Sec. 4.11 contains some concluding remarks.

This chapter analyzes applications of surface coding to quantum memory and quantum computation that could in principle be realized in any quantum computer that meets the criteria outlined in Sec. 4.2, whatever the details of how the local quantum gates are physically implemented. It has also been emphasized [60, 61] that surface codes may point the way toward realizations of intrinsically stable quantum memories (*physical* fault tolerance). In that case, protection against decoherence would be achieved without the need for active information processing, and how accurately the protected quantum states can be processed might depend heavily on the details of the implementation.

4.2 Fault tolerance and quantum architecture

To prove that a quantum computer with noisy gates can perform a robust quantum computation, we must make some assumptions about the nature of the noise and about how the computer operates. In fact, similar assumptions are needed to prove that a classical computer with noisy gates is robust [44]. Still, it is useful to list these requirements—they should always be kept in mind when we contemplate proposed schemes for building quantum computing hardware:

- *Constant error rate.* We assume that the strength of the noise is independent of the number of qubits in the computer. If the noise increases as we add qubits, then we cannot reduce the error rate to an arbitrarily low value by increasing the size of the code block.

- *Weakly correlated errors.* Errors must not be too strongly correlated, either in space or in time. In particular, fault-tolerant procedures fail if errors act simultaneously on many qubits in the same code block. If possible, the

hardware designer should strive to keep qubits in the same block isolated from one another.

- *Parallel operation.* We need to be able to perform many quantum gates in a single time step. Errors occur at a constant rate per unit time, and we are to control these errors through information processing. We could never keep up with the accumulating errors except by doing processing in different parts of the computer at the same time.

- *Reusable memory.* Errors introduce entropy into the computer, which must be flushed out by the error recovery procedure. Quantum processing transfers the entropy from the qubits that encode the protected data to "ancilla" qubits that can be discarded. Thus fresh ancilla qubits must be continually available. The ability to erase (or replace) the ancilla quickly is an essential hardware requirement [3].

In some estimates of the threshold, additional assumptions are made. While not strictly necessary to ensure the existence of a threshold, these assumptions may be useful, either because they simplify the analysis of the threshold or because they allow us to increase its numerical value. Hence these assumptions, too, should command the attention of the prospective hardware designer:

- *Fast measurements.* It is helpful to assume that a qubit can be measured as quickly as a quantum gate can be executed. For some implementations, this may not be a realistic assumption—measurement requires the amplification of a microscopic quantum effect to a macroscopic signal, which may take a while. But by measuring a classical error syndrome for each code block, we can improve the efficiency of error recovery. Furthermore, if we can measure qubits and perform quantum gates conditioned on classical measurement outcomes, then we can erase ancilla qubits by projecting onto the $\{|0\rangle, |1\rangle\}$ basis and flipping the qubit if the outcome is $|1\rangle$.

- *Fast and accurate classical processing.* If classical processing is faster and more accurate than quantum processing, then it is beneficial to substitute classical processing for quantum processing when possible. In particular, if the syndrome is measured, then a classical computation can be executed to determine how recovery should proceed. Ideally, the classical processors that coordinate the control of the quantum computer should be integrated into the quantum hardware.

- *No leakage.* It is typically assumed that, though errors may damage the state of the computer, the qubits themselves remain accessible—they do not "leak" out of the device. In fact, at least some types of leakage can be readily detected. If leaked qubits, once detected, can be replaced easily by fresh qubits, then leakage need not badly compromise performance. Hence, a desirable feature of hardware is that leaks are easy to detect and correct.

- *Nonlocal quantum gates.* Higher error rates can be tolerated, and the estimate of the threshold is simplified, if we assume that two-qubit quantum gates can act on any pair of qubits with a fidelity independent of the distance between the qubits. However useful, this assumption is not physically realistic. What the hardware designer can and should do, though, is try to arrange that qubits that will need to interact with one another are kept close to one another. In particular, the ancilla qubits that absorb entropy should be carefully integrated into the design [51].

If we do insist that all quantum gates are local, then another desirable feature is

- *High coordination number.* A threshold theorem applies even if qubits form a one-dimensional array [2, 51]. But local gates are more effective if the qubits are arranged in three dimensions, so that each qubit has more neighbors.

Suppose, then, that we are blessed with an implementation of quantum computation that meets all of our desiderata. Qubits are arranged in a three-dimensional

lattice, and can be projectively measured quickly. Reasonably accurate quantum gates can be applied in parallel to single qubits or to neighboring pairs of qubits. Fast classical processing is integrated into the qubit array. Under these conditions planar surface codes provide an especially attractive way to operate the quantum computer fault-tolerantly.

We may envision our quantum computer as a stack of planar sheets, with a protected logical qubit encoded in each sheet. Adjacent to each logical sheet is an associated sheet of ancilla qubits that are used to measure the error syndrome of that code block; after each measurement, these ancilla qubits are erased and then immediately reused. Encoded two-qubit gates can be performed between neighboring logical sheets, and any two logical sheets in the stack can be brought into contact by performing swap gates that move the sheets through the intervening layers of logical and ancilla qubits. As a quantum circuit is executed in the stack, error correction is continually applied to each logical sheet to protect against decoherence and other errors. Portions of the stack are designated as "software factories," where special ancilla states are prepared and purified—this software is then consumed during the execution of certain quantum gates that cannot be implemented directly.

A notable feature of this design (or other fault-tolerant designs) is that most of the information processing in the device is devoted to controlling errors, rather than moving the computation forward. How accurately must the fundamental quantum gates be executed for this error control to be effective, so that our machine is computationally powerful? The goal of this chapter is to address this question.

4.3 Surface codes

We will study the family of quantum error-correcting codes introduced in [60, 61]. These codes are especially well suited for fault-tolerant implementation, because the procedure for measuring the error syndrome is highly local.

4.3.1 Toric codes

For the code originally described in [60, 61], it is convenient to imagine that the qubits are in one-to-one correspondence with the links of a square lattice drawn on a torus, or, equivalently, drawn on a square with opposite edges identified. Hence we will refer to them as "toric codes." Toric codes can be generalized to a broader class of quantum codes, with each code in the class associated with a tessellation of a two-dimensional surface. Codes in this broader class will be called "surface codes."

A surface code is a special type of "stabilizer code" [18, 47]. A (binary) stabilizer code can be characterized as the simultaneous eigenspace with eigenvalue one of a set of mutually commuting check operators (or "stabilizer generators"), where each generator is a "Pauli operator." We use the notation

$$I = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, \quad X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \tag{4.1}$$

$$Y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}, \; Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} \tag{4.2}$$

for the 2×2 identity and Pauli matrices; a Pauli operator acting on n qubits is one of the 2^{2n} tensor product operators

$$\{I, X, Y, Z\}^{\otimes n}. \tag{4.3}$$

For the toric code defined by the $L \times L$ square lattice on the torus, there are $2L^2$ links of the lattice, and hence $2L^2$ qubits in the code block. Check operators are associated with each site and with each elementary cell (or "plaquette") of the lattice, as shown in Fig. 4.1. The check operator at site s acts nontrivially on the four links that meet at the site; it is the tensor product

$$X_s = \otimes_{\ell \ni s} X_\ell \tag{4.4}$$

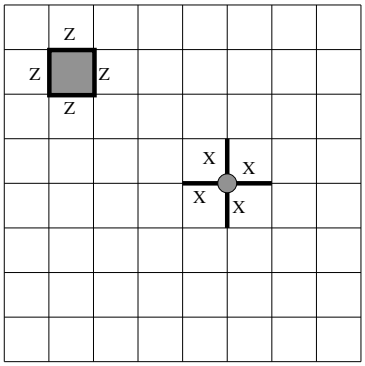

Figure 4.1: Check operators of the toric code. Each plaquette operator is a tensor product of Z's acting on the four links contained in the plaquette. Each site operator is a tensor product of X's acting on the four links that meet at the site.

acting on those four qubits, times the identity acting on the remaining qubits. The check operator at plaquette P acts nontrivially on the four links contained in the plaquette, as the tensor product

$$Z_P = \otimes_{\ell \in P} Z_\ell, \tag{4.5}$$

times the identity on the remaining links.

Although X and Z anticommute, the check operators are mutually commuting. Obviously, site operators commute with site operators, and plaquette operators with plaquette operators. Site operators commute with plaquette operators because a site operator and a plaquette operator act either on disjoint sets of links, or on sets whose intersection contains two links. In the former case, the operators obviously commute, and in the latter case, two cancelling minus signs arise when the site operator commutes through the plaquette operator. The check operators generate an Abelian group, the code's stabilizer.

The check operators can be simultaneously diagonalized, and the toric code is the space in which each check operator acts trivially. Because of the periodic

boundary conditions, each site or plaquette operator can be expressed as the product of the other $L^2 - 1$ such operators; the product of all L^2 site operators or all L^2 plaquette operators is the identity, since each link operator occurs twice in the product, and $X^2 = Z^2 = I$. There are no further relations among these operators; therefore, there are $2 \cdot (L^2 - 1)$ independent check operators, and hence two encoded qubits (the code subspace is four dimensional).

A Pauli operator that commutes with all the check operators will preserve the code subspace. What operators have this property? To formulate the answer, it is convenient to recall some standard mathematical terminology. A mapping that assigns an element of $Z_2 = \{0, 1\}$ to each link of the lattice is called a (Z_2-valued) *1-chain*. In a harmless abuse of language, we will also use the term 1-chain (or simply chain) to refer to the set of all links that are assigned the value 1 by such a mapping. The 1-chains form a vector space over Z_2—intuitively, the sum $u + v$ of two chains u and v is a disjoint union of the links contained in the two 1-chains. Similarly, 0-chains assign elements of Z_2 to lattice sites and 2-chains assign elements of Z_2 to lattice plaquettes; these also form vector spaces. A linear boundary operator ∂ can be defined that takes 2-chains to 1-chains and 1-chains to 0-chains: the boundary of a plaquette is the sum of the four links comprising the plaquette, and the boundary of a link is the sum of the two sites at the ends of the link. A chain whose boundary is trivial is called a *cycle*.

Now, any Pauli operator can be expressed as a tensor product of X's (and I's) times a tensor product of Z's (and I's). The tensor product of Z's and I's defines a Z_2-valued 1-chain, where links acted on by Z are mapped to 1 and links acted on by I are mapped to 0. This operator trivially commutes with all of the plaquette check operators, but commutes with a site operator if and only if an even number of Z's act on the links adjacent to the site. Thus, the corresponding 1-chain must be a cycle. Similarly, the tensor product of X's trivially commutes with the site operators, but commutes with a plaquette operator only if an even number of X's act on the links contained in the plaquette. This condition can be more conveniently expressed if we consider the dual lattice, in which sites and

plaquettes are interchanged; the links dual to those on which X acts form a cycle of the dual lattice. In general, then, a Pauli operator that commutes with the stabilizer of the code can be represented as a tensor product of Z's acting on a cycle of the lattice, times a tensor product of X's acting on a cycle of the dual lattice.

Cycles are of two distinct types. A 1-cycle is *homologically trivial* if it can be expressed as the boundary of a 2-chain (Fig. 4.2a). Thus, a homologically trivial cycle on our square lattice has an interior that can be "tiled" by plaquettes, and a product of Z's acting on the links of the cycle can be expressed as a product of the enclosed plaquette operators. This operator is therefore a product of the check operators—it is contained in the code stabilizer and acts trivially on the code subspace. Similarly, a product of X's acting on links that comprise a homologically trivial cycle of the dual lattice is also a product of check operators. Furthermore, *any* element of the stabilizer group of the toric code (any product of the generators) can be expressed as a product of Z's acting on a homologically trivial cycle of the lattice times X's acting on a homologically trivial cycle of the dual lattice.

But a cycle could be homologically nontrivial, that is, not the boundary of anything (Fig. 4.2b). A product of Z's corresponding to a nontrivial cycle commutes with the code stabilizer (because it is a cycle), but is not contained in the stabilizer (because the cycle is nontrivial). Therefore, while this operator preserves the code subspace, it acts nontrivially on encoded quantum information. Associated with the two fundamental nontrivial cycles of the torus, then, are the encoded operations $\bar{Z}_1$ and $\bar{Z}_2$ acting on the two encoded qubits. Associated with the two dual cycles of the dual lattice are the corresponding encoded operations $\bar{X}_1$ and $\bar{X}_2$, as shown in Fig 4.3.

A Pauli operator acting on n qubits is said to have *weight* w if the identity I acts on $n - w$ qubits and nontrivial Pauli matrices act on w qubits. The *distance* d of a stabilizer code is the weight of the minimal-weight Pauli operator that preserves the code subspace and acts nontrivially within the code subspace. If an encoded state is damaged by the action of a Pauli operator whose weight is less

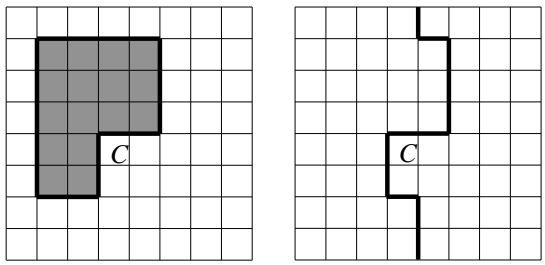

Figure 4.2: Cycles on the lattice. (a) A homologically trivial cycle bounds a region that can be tiled by plaquettes. The corresponding tensor product of Z's lies in the stabilizer of the toric code. (b) A homologically nontrivial cycle is not a boundary. The corresponding tensor product of Z's commutes with the stabilizer but is not contained in it. It is a logical operation that acts nontrivially in the code subspace.

than half the code distance, then we can recover from the error successfully by applying the minimal weight Pauli operator that returns the damaged state to the code subspace (which can be determined by measuring the check operators). For a toric code, the distance is the number of lattice links contained in the shortest homologically nontrivial cycle on the lattice or dual lattice. Thus in the case of an $L \times L$ square lattice drawn on the torus, the code distance is $d = L$.

The great virtue of the toric code is that the check operators are so simple. Measuring a check operator requires a quantum computation, but because each check operator involves just four qubits in the code block, and these qubits are situated near one another, the measurement can be executed by performing just a few quantum gates. Furthermore, the ancilla qubits used in the measurement can be situated where they are needed, so that the gates act on pairs of qubits that are in close proximity.

The observed values of the check operators provide a "syndrome" that we may use to diagnose errors. If there are no errors in the code block, then every check

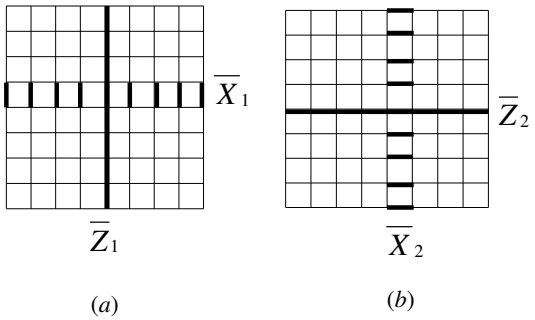

Figure 4.3: Basis for the operators that act on the two encoded qubits of the toric code. The logical operators $\bar{Z}_1$ and $\bar{Z}_2$ are tensor products of Z's associated with the fundamental nontrivial cycles of the torus constructed from links of the lattice. The complementary operators $\bar{X}_1$ and $\bar{X}_2$ are tensor products of X's associated with nontrivial cycles constructed from links of the dual lattice.

operator takes the value 1. Since each check operator is associated with a definite position on the surface, a site of the lattice or the dual lattice, we may describe the syndrome by listing all positions where the check operators take the value -1. It is convenient to regard each such position as the location of a particle, a "defect" in the code block.

If errors occur on a particular chain (a set of links of the lattice or dual lattice), then defects occur at the sites on the *boundary* of the chain. Evidently, then, the syndrome is highly ambiguous, as many error chains can share the same boundary, and all generate the same syndrome. For example, the two chains shown in Fig. 4.4 end on the same two sites. If errors occur on one of these chains, we might incorrectly infer that the errors actually occured on the other chain. Fortunately, though, this ambiguity need not cause harm. If Z errors occur on a particular chain, then by applying Z to each link of *any* chain with the same boundary as the actual error chain, we will successfully remove all defects. Furthermore, as long as the chosen chain is *homologically* correct (differs from the actual error chain

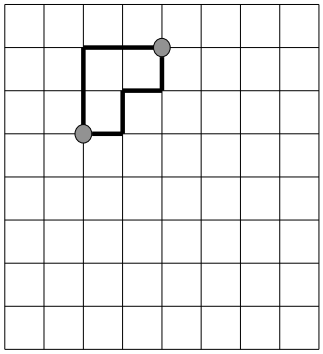

Figure 4.4: The highly ambiguous syndrome of the toric code. The two site defects shown could arise from errors on either one of the two chains shown. In general, error chains with the same boundary generate the same syndrome, and error chains that are homologically equivalent act on the code space in the same way.

by the one-dimensional boundary of a two-dimensional region), then the encoded state will be undamaged by the errors. In that event, the product of the actual Z errors and the Z's that we apply is contained in the code stabilizer and therefore acts trivially on the code block.

Heuristically, an error chain can be interpreted as a physical process in which a defect pair nucleates, and the two members of the pair drift apart. To recover from the errors, we lay down a "recovery chain" bounded by the two defect positions, which we can think of as a physical process in which the defects are brought together to reannihilate. If the defect world line consisting of both the error chain and the recovery chain is homologically trivial, then the encoded quantum state is undamaged. But if the world line is homologically nontrivial (if the two members of the pair wind around a cycle of the torus before reannihilating), then an error afflicts the encoded quantum state.

4.3.2 Planar codes

If all check operators are to be readily measured with local gates, then the qubits of the toric code need to be arranged on a topologically nontrivial surface, the torus, with the ancilla qubits needed for syndrome measurement arranged on an adjacent layer. In practice, the toroidal topology is likely to be inconvenient, especially if we want qubits residing in different tori to interact with one another in the course of a quantum computation. Fortunately, surface codes can be constructed in which all check operators are local and the qubits are arranged on planar sheets [17, 42]. The planar topology will be more conducive to realistic quantum computing architectures.

In the planar version of the surface code, there is a distinction between the check operators at the boundary of the surface and the check operators in the interior. Check operators in the interior are four-qubit site or plaquette operators, and those at the boundary are three-qubit operators. Furthermore, the boundary has two different types of edges as shown in Fig. 4.5. Along a "plaquette edge" or "rough edge," each check operator is a three-qubit plaquette operator $Z^{\otimes 3}$. Along a "site edge" or "smooth edge," each check operator is a three-qubit site operator $X^{\otimes 3}$.

As before, in order to commute with the code stabilizer, a product of Z's must act on an even number of links adjacent to each site of the lattice. Now, though, the links acted upon by Z's may comprise an *open* path that begins and ends on a rough edge. We may then say that the 1-chain comprised of all links acted upon by Z is a cycle *relative to the rough edges*. Similarly, a product of X's that commutes with the stabilizer acts on a set of links of the dual lattice that comprise a cycle relative to the smooth edges.

Cycles relative to the rough edges come in two varieties. If the chain contains an even number of the free links strung along the rough edge, then it can be tiled by plaquettes (including the boundary plaquettes), and so the corresponding product of Z's is contained in the stabilizer. We say that the relative 1-cycle is a relative

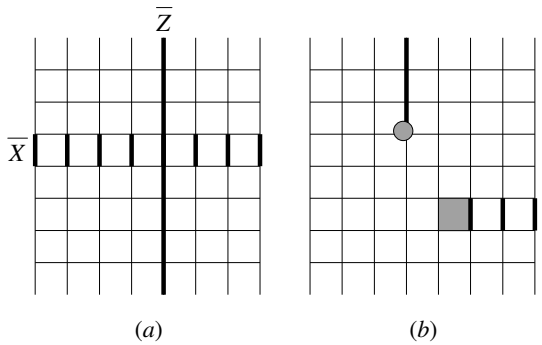

Figure 4.5: A planar quantum code. (a) At the top and bottom are the "plaquette edges" (or "rough edges"), where there are three-qubit plaquette operators, and at the left and right are the "site edges" (or "smooth edges"), where there are three-qubit site operators. The logical operation $\bar{Z}$ for the one encoded qubit is a tensor product of Z's acting on a chain running from one rough edge to the other, and the logical operation $\bar{X}$ is a tensor product of X's acting on a chain of the dual lattice running from one smooth edge to the other. For the lattice shown, the code's distance is $L = 8$. (b) Site and plaquette defects can appear singly, rather than in pairs. An isolated site defect arises from an error chain that ends at a rough edge, and an isolated plaquette defect arises from a dual error chain that ends at a smooth edge.

boundary of a 2-chain. However, a chain that stretches from one rough edge to another is not a relative boundary—it is a representative of a nontrivial relative homology class. The corresponding product of Z's commutes with the stabilizer but does not lie in it, and we may take it to be the logical operation $\bar{Z}$ acting on an encoded logical qubit. Similarly, cycles relative to the smooth edges also come in two varieties, and a product of X's associated with the nontrivial relative homology cycle of the dual lattice may be taken to be the logical operation $\bar{X}$ (see Fig. 4.5a).

A code with distance L is obtained from a square lattice, if the shortest paths from rough edge to rough edge, and from smooth edge to smooth edge, both contain L links. The lattice has $L^2 + (L-1)^2$ links, $L(L-1)$ plaquettes, and $L(L-1)$ sites. Now all plaquette and site operators are independent, which is another way to see that the number of encoded qubits is $L^2 + (L-1)^2 - 2L(L-1) = 1$.

The distinction between a rough edge and a smooth edge can also be characterized by the behavior of the defects at the boundary, as shown in Fig. 4.5b. In the toric codes, defects always appear in pairs, because every 1-chain has an even number of boundary points. But for planar codes, individual defects can appear, since a 1-chain can terminate on a rough edge. Thus a propagating site defect can reach the rough edge and disappear. But if the site defect reaches the smooth edge, it persists at the boundary. Similarly, a plaquette defect can disappear at the smooth edge, but not at the rough edge.

Let us briefly note some generalizations of the toric codes and planar codes that we have described. First, there is no need to restrict attention to lattices that have coordination number 4 at each site and plaquette. Any tessellation of a surface (and its dual tessellation) can be associated with a quantum code. Second, we may consider surfaces of higher genus. For a closed orientable Riemann surface of genus g, $2g$ qubits can be encoded—each time a handle is added to the surface, there are two new homology cycles and hence two new logical $\bar{Z}$'s. The distance of the code is the length of the shortest nontrivial cycle on lattice or dual lattice. For planar codes, we may consider a surface with e distinct rough edges separated

by e distinct smooth edges. Then $e-1$ qubits can be encoded, associated with the relative 1-cycles that connect one rough edge with any of the others. The distance is the length of the shortest path reaching from one rough edge to another, or from one smooth edge to another on the dual lattice. Alternatively, we can increase the number of encoded qubits stored in a planar sheet by punching holes in the lattice. For example, if the outer boundary of the surface is a smooth edge, and there are h holes, each bounded by a smooth edge, then h qubits are encoded. For each hole, a cycle on the lattice that encloses the hole is associated with the corresponding logical $\bar{Z}$, and a path on the dual lattice from the boundary of the hole to the outer boundary is associated with the logical $\bar{X}$.

If (say) phase errors are more common than bit-flip errors, quantum information can be stored more efficiently with an *asymmetric* planar code, such that the distance from rough edge to rough edge is longer than the distance from smooth edge to smooth edge. However, these asymmetric codes are less convenient for processing of the encoded information.

The surface codes can also be generalized to higher dimensional manifolds, with logical operations again associated with homologically nontrivial cycles. In Sec. 4.10, I will discuss a four-dimensional example.

4.3.3 Fault-tolerant recovery

A toric code defined on a lattice of linear size L has block size $2L^2$ and distance L. Therefore, if the probability of error per qubit is p, the number of errors expected in a large code block is of order pL^2, and therefore much larger than the code distance.

However, the performance of a toric code is much better than would be guessed naively based on its distance. In principle, $L/2$ errors could suffice to cause damage to the encoded information. But in fact this small number of errors can cause irrevocable damage only if the distribution of the errors is highly atypical.

If the error probability p is small, then links where errors occur ("error links") are dilute on the lattice. Long connected chains of error links are quite rare, as

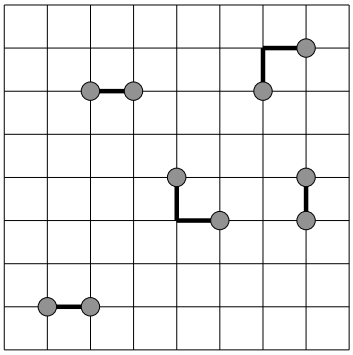

Figure 4.6: Pairs of defects. If the error rate is small and errors on distinct links are uncorrelated, then connected error chains are typically short and the positions of defects are highly correlated. It is relatively easy to guess how the defects should be paired up so that each pair is the boundary of a connected chain.

indicated in Fig. 4.6. It is relatively easy to guess a way to pair up the observed defects that is homologically equivalent to the actual error chain. Hence we expect that a number of errors that scales *linearly* with the block size can be tolerated. That is, if the error probability p per link is small enough, we expect to be able to recover correctly with a probability that approaches one as the block size increases. We therefore anticipate that there is an accuracy threshold for storage of quantum information using a toric code.

Unfortunately, life is not quite so simple, because the measurement of the syndrome will not be perfect. Occasionally, a faulty measurement will indicate that a defect is present at a site even though no defect is actually there, and sometimes an actual defect will go unobserved. Hence the population of real defects (which have strongly correlated positions) will be obscured by a population of phony "ghost defects" and "missing defects" (which have randomly distributed positions), as in Fig. 4.7.

Therefore, we should execute recovery cautiously. It would be dangerous to blithely proceed by flipping qubits on a chain of links bounded by the observed

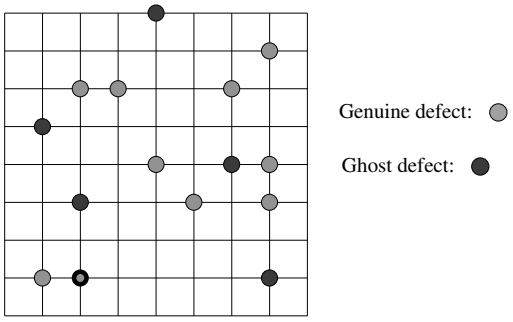

Figure 4.7: Ghost defects. Since faults can occur in the measurement of the error syndrome, the measured syndrome includes both genuine defects (lightly shaded) associated with actual errors and phony "ghost defects" (darkly shaded) that arise at randomly distributed locations. To perform recovery successfully, we need to be able to distinguish reliably between the genuine defects and the ghost defects. The position that is shaded both lightly and darkly represents a genuine defect that goes unseen due to a measurement error.

defect positions. Since a ghost defect is typically far from the nearest genuine defect, this procedure would introduce many additional errors—what was formerly a ghost defect would become a real defect connected to another defect by a long error chain. Instead we must repeat the syndrome measurement an adequate number of times to verify its authenticity. It is subtle to formulate a robust recovery procedure that incorporates repeated measurements, since further errors accumulate as the measurements are repeated and the gas of defects continues to evolve.

There are three well-studied general strategies that can be invoked to achieve robust macroscopic control of a system that is subjected to microscopic disorder. One method is to introduce a hierarchical organization in such a way that effects of noise get weaker and weaker at higher and higher levels of the hierarchy. This approach is used by Gács [44] in his analysis of robust one-dimensional classical cellular automata, and also in concatenated quantum coding [67, 2, 62, 87, 48]. A second method is to introduce more spatial dimensions. A fundamental principle of statistical physics is that local systems with higher spatial dimensionality and hence higher coordination number are more resistant to the disordering effects of fluctuations. In Sec. 4.10 this strategy will be followed in devising and analyzing a topological code that has nice locality properties in four dimensions. From the perspective of block coding, the advantage of extra dimensions is that local check operators can be constructed with a higher degree of redundancy, which makes it easier to reject faulty syndrome information.

In the bulk of this chapter I will address the issue of achieving robustness through a third strategy, namely by introducing a modest amount of nonlocality into the recovery procedure. However, all quantum processing will be demanded to be strictly local; the nonlocality will be isolated in *classical* processing. Specifically, to decide on the appropriate recovery step, a classical computation will be performed whose input is an error syndrome measured at all the sites of the lattice. This classical computation will be required to be able to be executed in a time bounded by a polynomial in the number of lattice sites. For the purpose of estimating the accuracy threshold, we will imagine that the classical calculation is

instantaneous and perfectly accurate.

This approach is guided by the expectation that quantum computers will be slow and unreliable while classical computers are fast and accurate. It is therefore advantageous to replace quantum processing by classical processing if the classical processing can accomplish the same task.

4.3.4 Surface codes and physical fault tolerance

In this chapter, the surface codes are regarded as block quantum error-correcting codes with properties that make them especially amenable to fault-tolerant quantum storage and computation. It is worth remarking that because of the locality of the check operators, these codes admit another tempting interpretation that was emphasized in [60, 61].

Consider a model physical system, with qubits arranged in a square lattice, and with a (local) Hamiltonian that can be expressed as minus the sum of the check operators of a surface code. Since the check operators are mutually commuting, we can diagonalize the Hamiltonian by diagonalizing each check operator separately, and its degenerate ground state is the code subspace. Thus, a real system that is described well enough by this model could serve as a robust quantum memory.

The model system has several crucial properties. First of all, it has a mass gap, so that its qualitative properties are stable with respect to generic weak local perturbations. Secondly, it has two types of localized quasiparticle excitations, the site defects and plaquette defects. And third, there is an exotic long-range interaction between a site defect and a plaquette defect.

The interaction between the two defects is exactly analogous to the Aharonov-Bohm interaction between a localized magnetic flux Φ and a localized electric charge Q in two-spatial dimensions. When a charge is adiabatically carried around a flux, the wave function of the system is modified by a phase $\exp(iQ\Phi/\hbar c)$ that is independent of the separation between charge and flux. Similarly, if a site defect is transported around a plaquette defect, the wave function of the system is modified by the phase -1 independent of the separation between the defects. Formally,

this phase arises because of the anticommutation relation satisfied by X and Z. Physically, it arises because the ground state of the system is very highly entangled and thus is able to support very long range quantum correlations. The protected qubits are encoded in the Aharonov-Bohm phases acquired by quasiparticles that travel around the fundamental nontrivial cycles of the surface; these could be measured in principle in a suitable quantum interference experiment.

It is useful to observe that the degeneracy of the ground state of the system is a necessary consequence of the unusual interactions among the quasiparticles [34, 108]. A unitary operator $U_{S,1}$ can be constructed that describes a process in which a pair of site defects is created, one member of the pair propagates around a nontrivial cycle C_1 of the surface, and then the pair reannihilates. Similarly a unitary operator $U_{P,2}$ can be constructed associated with a plaquette defect that propagates around a complementary nontrivial cycle C_2 that intersects C_1 once. These operators commute with the Hamiltonian H of the system and can be simultaneously diagonalized with H, but $U_{S,1}$ and $U_{P,2}$ do not commute with one another. Rather, they satisfy (in an infinite system)

$$ {U_{P,2}}^{-1}\, {U_{S,1}}^{-1}\, U_{P,2}\, U_{S,1} = -1. \tag{4.6} $$

The nontrivial commutator arises because the process in which (1) a site defect winds around C_1, (2) a plaquette defect winds around C_2 (3) the site defect winds around C_1 in the reverse direction, and (4) the plaquette defect winds around C_2 in the reverse direction, is topologically equivalent to a process in which the site defect winds once around the plaquette defect.

Because the unitary operators $U_{S,1}$ and $U_{P,2}$ do not commute, they cannot be simultaneously diagonalized—indeed applying $U_{P,2}$ to an eigenstate of $U_{S,1}$ flips the sign of the $U_{S,1}$ eigenvalue. Physically, there are two distinct ground states that can be distinguished by the Aharonov-Bohm phase that is acquired when a site defect is carried around C_1; we can change this phase by carrying a plaquette defect around C_2. Similarly, the operator $U_{S,2}$ commutes with $U_{S,1}$ and $U_{P,2}$ but

anticommutes with $U_{P,1}$. Therefore there are four distinct ground states, labelled by their $U_{S,1}$ and $U_{S,2}$ eigenvalues.

This reasoning shows that the topological interaction between site defects and plaquette defects implies that the system on an (infinite) torus has a generic four-fold ground-state degeneracy. The argument is easily extended to show that the generic degeneracy on a genus g Riemann surface is 2^{2g}. By a further extension, we see that the generic degeneracy is q^{2g} if the Aharonov-Bohm phase associated with winding one defect around another is

$$\exp(2\pi ip/q), \tag{4.7}$$

where p and q are integers with no common factor.

The same sort of argument can be applied to planar systems with a mass gap in which single defects can disappear at an edge. For example, consider an annulus in which site defects can disappear at the inner and outer edges. Then states can be classified by the Aharonov-Bohm phase acquired by a plaquette defect that propagates around the annulus, a phase that flips in sign if a site defect propagates from inner edge to outer edge. Hence there is a twofold degeneracy on the annulus. For a disc with h holes, the degeneracy is 2^h if site defects can disappear at any boundary, or q^h if the Aharonov-Bohm phase of site defect winding about plaquette defect is $\exp(2\pi ip/q)$.

These degeneracies are exact for the unperturbed model system, but will be lifted slightly in a weakly perturbed system of finite size. Loosely speaking, the effect of perturbations will be to give the defects a finite effective mass, and the lifting of the degeneracy is associated with quantum tunneling processes in which a virtual defect winds around a cycle of the surface. The amplitude A for this process has the form

$$A \sim C \exp\left(-\sqrt{2}(m^*\Delta)^{1/2}L/\hbar\right), \tag{4.8}$$

where L is the physical size of the shortest nontrivial (relative) cycle of the surface,

m^* is the defect effective mass, and Δ is the minimal energy cost of creating a defect. The energy splitting is proportional to A, and like A becomes negligible when the system is large compared to the characteristic length $l \equiv \hbar(m^*\Delta)^{-1/2}$.

In this limit, and at sufficiently low temperature, the degenerate ground state provides a reliable quantum memory. If a pair of defects is produced by a thermal fluctuation, and one of the defects wanders around a nontrivial cycle before the pair reannihilates, then the encoded quantum information will be damaged. These fluctuations are suppressed by the Boltzmann factor $\exp(-\Delta/kT)$ at low temperature. Even if defect nucleation occurs at a nonnegligible rate, we could enhance the performance of the quantum memory by continually monitoring the state of the defect gas. If the winding of defects around nontrivial cycles is detected and carefully recorded, damage to the encoded quantum information can be controlled.

4.4 The statistical physics of error recovery

One of the main objectives of this chapter is to invoke surface coding to establish an accuracy threshold for quantum computation—how well must quantum hardware perform for quantum storage, or universal quantum computation, to be achievable with arbitrarily small probability of error? In this section, rather than study the efficacy of a particular fault-tolerant protocol for error recovery, I will address whether the syndrome of a surface code is adequate in principle for protecting quantum information from error. Specifically, an order parameter that distinguishes two phases of a quantum memory will be formulated: an "ordered" phase, in which reliable storage is possible, and a "disordered phase," in which errors unavoidably afflict the encoded quantum information. Of course, this phase boundary also provides an upper bound on the accuracy threshold that can be reached by any particular protocol. The toric code and the planar surface code have the same accuracy threshold, so we may study either to learn about the other.

4.4.1 The error model

Let us imagine that in a single time step, we will execute a measurement of each stabilizer operator at each site and each plaquette of the lattice. During each time step, new qubit errors might occur. To be concrete and to simplify the discussion, we assume that all qubit errors are stochastic, and so can be assigned probabilities. (For example, errors that arise from decoherence have this property.) We will also assume that the errors acting on different qubits are independent, that bit-flip (X) errors and phase (Z) errors are uncorrelated with one another, and that X and Z errors are equally likely. Thus the error in each time step acting on a qubit with state ρ can be represented by the quantum channel

$$\begin{aligned} \rho \to \quad & (1-p)^2 I\rho I + p(1-p)X\rho X \\ & + p(1-p)Z\rho X + p^2 Y\rho Y, \end{aligned} \tag{4.9}$$

where p denotes the probability of either an X error or a Z error. It is easy to modify our analysis if some of these assumptions are relaxed; in particular, correlations between X and Z errors would not cause much trouble, since we have separate procedures for recovery from the X errors and the Z errors.

Faults can also occur in the syndrome measurement. We assume that these measurement errors are uncorrelated. We will denote by q the probability that the measured syndrome bit is faulty at a given site or plaquette.

Aside from being uncorrelated in space, the qubit and measurement errors are also assumed to be uncorrelated in time. Furthermore, the qubit and measurement errors are not correlated with one another. We assume that p and q are known quantities—the choice of recovery algorithm depends on their values. In Sec. 4.7, I will discuss how p and q can be related to more fundamental quantities, namely the fidelities of elementary quantum gates. There we will see that the execution of the syndrome measurement circuit can introduce correlations between errors. Fortunately, these correlations (which we ignore for now) do not have a big impact on the accuracy threshold.

4.4.2 Defects in spacetime

Because syndrome measurement may be faulty, it is necessary to repeat the measurement to improve our confidence in the outcome. But since new errors may arise during the repeated measurements, it is a subtle matter to formulate an effective procedure for rejecting measurement errors.

Let us suppose, for a toric block of arbitrarily large size, that we measure the error syndrome once per time step, that we monitor the block for an arbitrarily long time, and that we store all of the syndrome information that is collected. We want to address whether this syndrome information enables us to recover from errors with a probability of failure that becomes exponentially small as the size of the toric block increases. The plaquette check operators identify bit flips and the site check operators identify phase errors; therefore, we consider bit-flip and phase error recovery separately.

For analyzing how the syndrome information can be used most effectively, it is quite convenient to envision a *three-dimensional* simple cubic lattice, with the third dimension representing an integer-valued *time*. We imagine that the error operation acts at each integer-valued time t, with a syndrome measurement taking place in between each t and $t+1$. Qubits in the code block can now be associated with timelike plaquettes, those lying in the tx and ty planes. A qubit error that occurs at time t is associated with a horizontal (spacelike) link that lies in the time slice labelled by t. The outcome of the measurement of the stabilizer operator $X_s = X^{\otimes 4} = \pm 1$ at site s, performed between time t and time $t+1$, is marked on the vertical (timelike) link connecting site s at time t and site s at time $t+1$. A similar picture applies to the history of the Z_P stabilizer operators at each plaquette, but with the lattice replaced by its dual.

On some of these vertical links, the measured syndrome is erroneous. We will repeat the syndrome measurement T times in succession, and the "error history" can be described as a set of marked links on a lattice with altogether T time slices. The error history encompasses both error events that damage the qubits in the

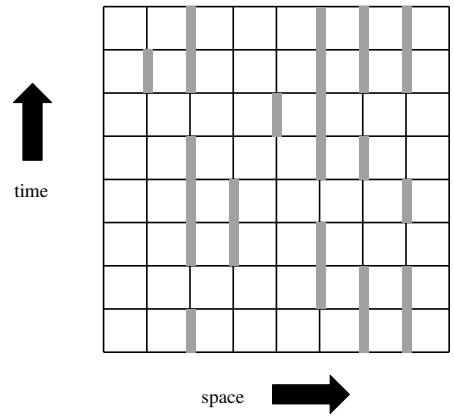

Figure 4.8: The two-dimensional lattice depicting a history of the error syndrome for the quantum repetition code, with time running upward. Each row represents the syndrome at a particular time. Qubits reside on plaquettes, and two-qubit check operators are measured at each vertical link. Links where the syndrome is nontrivial are shaded.

code block, and faults in the syndrome measurements. On the initial ($t = 0$) slice are marked all uncorrected qubit errors that are left over from previous rounds of error correction; new qubit errors that arise at a later time t ($t = 1, 2, \ldots, T-1$) are marked on horizontal links on slice t. Errors in the syndrome measurement that takes place between time t and $t+1$ are marked on the corresponding vertical links. Errors on horizontal links occur with probability p, and errors on vertical links occur with probability q.

For purposes of visualization, it is helpful to consider the simpler case of a quantum repetition code, which can be used to protect coherent quantum information from bit-flip errors if there are no phase errors (or phase errors if there are no bit-flip errors). In this case we may imagine that qubits reside on sites of a periodically identified one-dimensional lattice (*i.e.*, a circle); at each link the stabilizer generator ZZ acts on the two neighboring sites. Then there is one encoded qubit—the two-dimensional code space is spanned by the state $|000\ldots0\rangle$ with all spins "up," and the state $|111\ldots\rangle$ with all spins "down." In the case where the

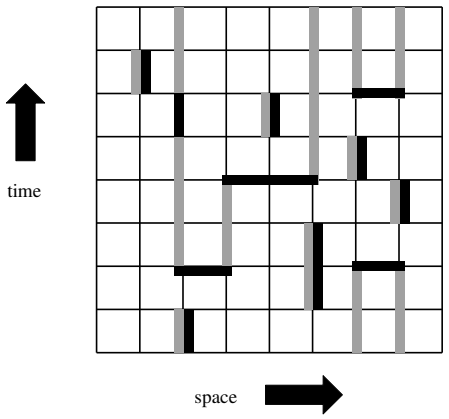

Figure 4.9: An error history shown together with the syndrome history that it generates, for the quantum repetition code. Links where errors occurred are darkly shaded, and links where the syndrome is nontrivial are lightly shaded. Errors on horizontal links indicate where a qubit flipped between successive syndrome measurements, and errors on vertical links indicate where the syndrome measurement was wrong. Vertical links that are shaded both lightly and darkly are locations where a nontrivial syndrome was found erroneously. The chain of lightly shaded links (the syndrome) and the chain of darkly shaded links (the errors) both have the same boundary.

syndrome measurement is repeated to improve reliability, we may represent the syndrome's history by associating qubits with plaquettes of a two-dimensional lattice, and syndrome bits with the timelike links, as shown in Fig. 4.8 and Fig. 4.9. Again, bit-flip errors occur on horizontal links with probability p and syndrome measurement errors occur on vertical links with probability q.

Of course, as already noted in Sec. 4.3.3, we may also use a two-dimensional lattice to represent the error configuration of the toric code, in the case where the syndrome measurements are perfect. In that case, we can collect reliable information by measuring the syndrome in one shot, and errors occur on links of the two-dimensional lattice with probability p.

4.4.3 Error chains, world lines, and magnetic flux tubes

In practice, we will always want to protect quantum information for some finite time. But for the purpose of investigating whether error correction will work effectively in principle, it is convenient to imagine that our repeated rounds of syndrome measurement extend indefinitely into the past and into the future. Qubit errors are continually occurring; as defects are created in pairs, propagate about on the lattice, and annihilate in pairs, the world lines of the defects form closed loops in spacetime. Some loops are homologically trivial and some are homologically nontrivial. Error recovery succeeds if we are able to correctly identify the homology class of each closed loop. But if a homologically nontrivial loop arises that we fail to detect, or if we mistakenly believe that a homologically nontrivial loop has been generated when none has been, then error recovery will fail. For now, let us consider this scenario in which we continue to measure the syndrome forever—in Sec. 4.6, we will consider some issues that arise when we perform error correction for a finite time.

So let us imagine a particular history extending over an indefinite number of time slices, with the observed syndrome marked on each vertical link, measurement errors marking selected vertical links, and qubit errors marking selected horizontal links. For this history we may identify several distinct 1-chains (sets of links).

We denote by S the *syndrome chain* containing all (vertical) links at which the measured syndrome is nontrivial ($X_s = -1$). We denote by E the *error chain* containing all links where errors have occurred, including both qubit errors on horizonal links and measurement errors on vertical links. Consider $S + E$, the disjoint union of S and E ($S + E$ contains the links that are in either S or E, but not both). The chain $S + E$ represents the "actual" world lines of the defects generated by qubit errors, as illustrated in Fig. 4.9. Its vertical links are those on which the syndrome would be nontrivial were it measured without error. Its horizontal links are events where a defect pair is created, a pair annihilates, or an existing defect propagates from one site to a neighboring site. Since the world lines never end, the chain $S + E$ has no boundary, $\partial(S + E) = 0$. Equivalently S and E have the same boundary, $\partial S = \partial E$.

Hence, the measured syndrome S reveals the boundary of the error chain E; we may write $E = S + C$, where C is a *cycle* (a chain with no boundary). But any other error chain $E' = S + C'$, where C' is a cycle, has the same boundary as E and therefore could have caused the same syndrome. To recover from error, we will use the syndrome information to make a hypothesis, guessing that the actual error chain was $E' = S + C'$. Now, E' may not be the same chain as E, but as long as the cycle $E + E' = C + C'$ is homologically trivial (the boundary of a surface) then recovery will be successful. If $C + C'$ is homologically nontrivial, then recovery will fail. We say that C and C' are in the same *homology class* if $C + C'$ is homologically trivial. Therefore, whether we can protect against error hinges on our ability to identify, not the cycle C, but rather the homology class of C.

Considering the set of all possible histories, let $\mathrm{prob}(E')$ denote the probability of the error chain E' (strictly speaking, we should consider the total elapsed time to be finite for this probability to be defined). Then the probability that the syndrome S was caused by any error chain $E' = S + C'$, such that C' belongs to

the homology class h, is

$$\mathrm{prob}(h|S) = \frac{\sum_{C' \in h} \mathrm{prob}(S + C')}{\sum_{C'} \mathrm{prob}(S + C')}. \tag{4.10}$$

Clearly, then, given a measured syndrome S, the optimal way to recover is to guess that the homology class h of C is the class with the highest probability according to eq. (4.10). Recovery succeeds if C belongs to this class, and fails otherwise.

We say that the probability of error per qubit lies below the accuracy threshold if and only if the recovery procedure fails with a probability that vanishes as the linear size L of the lattice increases to infinity. Therefore, below threshold, the cycle C actually belongs to the class h that maximizes eq. (4.10) with a probability that approaches one as $L \to \infty$. It is convenient to restate this criterion in a different way that makes no explicit reference to the syndrome chain S. We may write the relation between the actual error chain E and the hypothetical error chain E' as $E' = E + D$, where D is the cycle that we called $C + C'$ above. Let $\mathrm{prob}[(E+D)|E]$ denote the normalized conditional probability for error chains $E' = E + D$ that have the same boundary as E. Then, the probability of error per qubit lies below threshold if and only if, in the limit $L \to \infty$,

$$\sum_E \mathrm{prob}(E) \cdot \sum_{D \text{ nontrivial}} \mathrm{prob}[(E + D)|E] = 0. \tag{4.11}$$

Eq. (4.11) says that error chains that differ from the actual error chain by a homologically nontrivial cycle have probability zero. Therefore, the observed syndrome S is sure to point to the correct homology class, in the limit of an arbitrarily large code block.

This accuracy threshold achievable with toric codes can be identified with a phase transition in a particular statistical-physics model defined on a lattice. In a sense that I will make precise, the error chains are analogous to magnetic flux tubes in a superconductor, and the boundary points of the error chains are magnetic monopoles where these flux tubes terminate. Fixing the syndrome pins down the monopoles, and the ensemble of chains with a specified boundary can be regarded

as a thermal ensemble. As the error probability increases, the thermal fluctuations of the flux tubes increase, and at the critical temperature corresponding to the accuracy threshold, the flux tubes condense and the superconductivity is destroyed.

A similar analogy applies to the case where the syndrome is measured perfectly, and a two-dimensional system describes the syndrome on a single time slice. Then the error chains are analogous to domain walls in an Ising ferromagnet, and the boundary points of the error chains are "Ising vortices" where domain walls terminate. Fixing the syndrome pins down the vortices, and the ensemble of chains with a specified boundary can be interpreted as a thermal ensemble. As the error probability increases, the domain walls heat up and fluctuate more vigorously. At a critical temperature corresponding to the accuracy threshold, the domain walls condense and the system becomes magnetically disordered. This two-dimensional model also characterizes the accuracy threshold achievable with a quantum repetition code, if the syndrome is imperfect and the qubits are subjected only to bit-flip errors (or only to phase errors).

4.4.4 Derivation of the model

Let us establish the precise connection between our error model and the corresponding statistical-physics model. In the two-dimensional case, we consider a square lattice with links representing qubits, and assume that errors arise independently on each link with probability p. In the three-dimensional case, we consider a simple cubic lattice. Qubits reside on the timelike plaquettes, and qubit errors arise independently with probability p on spacelike links. Measurement errors occur independently with probability q on timelike links. For now, we will make the simplifying assumption that $q = p$ so that the model is isotropic; the generalization to $q \neq p$ is straightforward.

An error chain E, in either two or three dimensions, can be characterized by a function $n_E(\ell)$ that takes a link ℓ to $n_E(\ell) \in \{0, 1\}$, where $n_E(\ell) = 1$ for each link

ℓ that is occupied by the chain. Hence the probability that error chain E occurs is

$$\begin{aligned} \text{prob}(E) &= \prod_{\ell} (1-p)^{1-n_E(\ell)} p^{n_E(\ell)} \\ &= \left[\prod_{\ell}(1-p)\right] \cdot \prod_{\ell} \left(\frac{p}{1-p}\right)^{n_E(\ell)}, \end{aligned} \tag{4.12}$$

where the product is over all links of the lattice.

Now suppose that the error chain E is fixed, and we are interested in the probability distribution for all chains E' that have the same boundary as E. Note that we may express $E' = E + C$, where C is a cycle (a chain with no boundary) and consider the probability distribution for C. Then if $n_C(\ell) = 1$ and $n_E(\ell) = 0$, the link ℓ is occupied by E' but not by E, an event whose probability (aside from an overall normalization) is

$$\left(\frac{p}{1-p}\right)^{n_C(\ell)}. \tag{4.13}$$

But if $n_C(\ell) = 1$ and $n_E(\ell) = 1$, then the link ℓ is not occupied by E', an event whose probability (aside from an overall normalization) is

$$\left(\frac{1-p}{p}\right)^{n_C(\ell)}. \tag{4.14}$$

Thus a chain $E' = E + C$ with the same boundary as E occurs with probability

$$\text{prob}(E'|E) \propto \prod_{\ell} \exp\left(J_\ell u_\ell\right); \tag{4.15}$$

here we have defined

$$u_\ell = 1 - 2n_C(\ell) \in \{1, -1\}, \tag{4.16}$$

and the coupling J_ℓ assigned to link ℓ has the form

$$e^{-2J_\ell} = \begin{cases} p/(1-p), & \text{for } \ell \notin E, \\ (1-p)/p, & \text{for } \ell \in E. \end{cases} \tag{4.17}$$

Recall that the 1-chain $\{\ell | u_\ell = -1\}$ is required to be a *cycle*—it has no boundary.

It is obvious from this construction that prob$(E'|E)$ does not depend on how the chain E is chosen—it depends only on the boundary of E. We will verify this explicitly below.

The cycle condition satisfied by the u_ℓ's can be expressed as

$$\prod_{\ell \ni s} u_\ell = 1; \tag{4.18}$$

at each site s, an even number of links incident on that site have $u_\ell = -1$. It is convenient to *solve* this condition, expressing the u_ℓ's in terms of unconstrained variables. To achieve this in two dimensions, we associate with each link ℓ a link ℓ^* of the *dual lattice.* Under this duality, sites are mapped to plaquettes, and the cycle condition becomes

$$\prod_{\ell^* \in P^*} u_{\ell^*} = 1. \tag{4.19}$$

To solve the constraint, we introduce variables $\sigma_i \in \{1, -1\}$ associated with each site i of the dual lattice, and write

$$u_{ij} = \sigma_i \sigma_j \tag{4.20}$$

where i and j are nearest-neighbor sites.

Our solution to the constraint is not quite the most general possible. In the language of differential forms, we have solved the condition $du = 0$ (where u is a discrete version of a one-form, and d denotes the exterior derivative) by writing $u = d\sigma$, where σ is a zero-form. Thus our solution misses the cohomologically nontrivial closed forms, those that are not exact. In the language of homology, our solution includes all and only those cycles that are homologically trivial—that is, cycles that bound a surface.

In three dimensions, links are dual to plaquettes, and sites to cubes. The cycle

condition becomes, on the dual lattice,

$$\prod_{P^* \in C^*} u_{P^*} = 1; \tag{4.21}$$

each dual cube C^* contains an even number of dual plaquettes that are occupied by the cycle. We solve this constraint by introducing variables $\sigma_{\ell^*} \in \{1, -1\}$ on the dual links, and defining

$$u_{P^*} = \prod_{\ell^* \in P^*} \sigma_{\ell^*}. \tag{4.22}$$

In this case, we have solved a discrete version of $du = 0$, where u is a two-form, by writing $u = d\sigma$, where σ is a one-form. Once again, our solution generates only the cycles that are homologically trivial.

We have now found that, in two dimensions, the "fluctuations" of the error chains E' that share a boundary with the chain E are described by a statistical-mechanical model with partition function

$$Z[J, \eta] = \sum_{\{\sigma_i\}} \exp\left(J \sum_{\langle ij \rangle} \eta_{ij} \sigma_i \sigma_j \right), \tag{4.23}$$

where $e^{-2J} = p/(1-p)$. The sum in the exponential is over pairs of nearest neighbors on a square lattice, and $\eta_\ell \in \{1, -1\}$ is defined by

$$\eta_\ell = \begin{cases} 1, & \text{if } \ell \notin E^*, \\ -1 & \text{if } \ell \in E^*. \end{cases} \tag{4.24}$$

Furthermore if the error chains E and E' are generated by sampling the same probability distribution, then the η_ℓ's are chosen at random subject to

$$\eta_\ell = \begin{cases} 1, & \text{with probability } 1-p, \\ -1 & \text{with probability } p. \end{cases} \tag{4.25}$$

This model is the well-known "random-bond Ising model." Furthermore, the re-

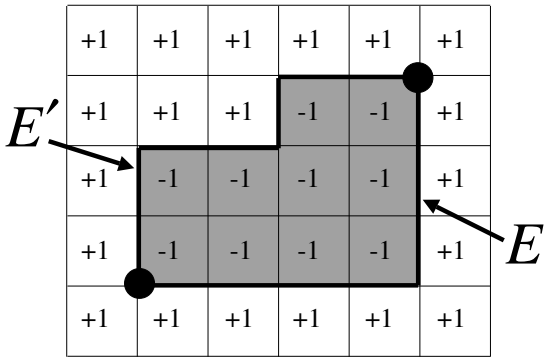

Figure 4.10: The "quenched" error chain E and the "fluctuating" error chain E', as represented in the two-dimensional random-bond Ising model. Ising spins taking values in $\{\pm 1\}$ reside on plaquettes, Ising vortices are located on the sites marked by filled circles, and the coupling between neighboring spins is antiferromagnetic along the path E that connects the Ising vortices. The links of E' comprise a domain wall connecting the vortices. The closed path $C = E + E'$ encloses a domain of spins with the value -1.

lation $e^{-2J} = p/(1-p)$ between the coupling and the bond probability defines the "Nishimori line" [81] in the phase diagram of the model, which has attracted substantial attention[1] because the model is known to have enhanced symmetry properties on this line.

Perhaps the interpretation of this random-bond Ising model can be grasped better if we picture the original lattice rather than the dual lattice, so that the Ising spins reside on plaquettes as in Fig. 4.10. The coupling between spins on neighboring plaquettes is antiferromagnetic on the links belonging to the chain E (where $\eta_\ell = -1$), meaning that it is energetically preferred for the spins to antialign at these links. At links not in E (where $\eta = 1$), it is energetically preferred for the spins to align. Thus a link ij is excited if $\eta_{ij}\sigma_i\sigma_j = -1$. We say that the excited links constitute "domain walls." In the case where $\eta_\ell = 1$ on every link, a

[1] For a recent discussion, see [54].

wall marks the boundary between two regions in which the spins point in opposite directions. Walls can never end, because the boundary of a boundary is zero.

But if the η configuration is nontrivial then the "walls" can end. Indeed each boundary point of the chain E of links with $\eta_\ell = -1$ is an endpoint of a wall, what we will call an "Ising vortex." For example, for the configuration shown in Fig 4.10, a domain wall occupies the chain E' that terminates on Ising vortices at the marked sites. The figure also illustrates that the model depends only on the boundary of the chain E, and not on other properties of the chain. To see this, imagine performing the change of variables

$$\sigma_i \to -\sigma_i, \tag{4.26}$$

on the shaded plaquettes of Fig. 4.10. A mere change of variable cannot alter the locations of the excited links—rather the effect is to shift the antiferromagnetic couplings from the chain E to a different chain E' with the same boundary.

In three dimensions, the fluctuations of the error chains that share a boundary with the specified chain E are described by a model with partition function

$$Z[J,\eta] = \sum_{\{\sigma_\ell\}} \exp(J \sum_P \eta_P u_P), \tag{4.27}$$

where $u_P = \prod_{\ell \in P} \sigma_\ell$ and

$$\eta_P = \begin{cases} 1, & \text{if } P \notin E^*, \\ -1, & \text{if } P \in E^*. \end{cases} \tag{4.28}$$

This model is a "random-plaquette" $\mathbb{Z}_2$ gauge theory in three dimensions, which, to the best of my knowledge, has not been much studied previously. Again, we are interested in the "Nishimori line" of this model where $e^{-2J} = p/(1-p)$, and p is the probability that a plaquette has $\eta_P = -1$.

In this three-dimensional model, we say that a plaquette P is excited if $\eta_P u_P = -1$. The excited plaquettes constitute "magnetic flux tubes"—these form closed

loops on the original lattice if $\eta_P = 1$ on every plaquette. But at each boundary point of the chain E on the original lattice (each cube on the dual lattice that contains an odd number of plaquettes with $\eta_P = -1$), the flux tubes can end. The sites of the original lattice (or cubes of the dual lattice) that contain endpoints of magnetic flux tubes are said to be "magnetic monopoles."

4.4.5 Order parameters

As noted, our statistical-mechanical model includes a sum over those and only those chains E' that are *homologically equivalent* to the chain E. To determine whether errors can be corrected reliably, we want to know whether chains E' in a *different* homology class than E have negligible probability in the limit of a large lattice (or code block). The relative likelihood of different homology classes is determined by the free energy difference of the classes; in the ordered phase, we anticipate that the free energy of nontrivial classes exceeds that of the trivial classes by an amount that increases linearly with L, the linear size of the lattice.

But for the purpose of finding the value of the error probability at the accuracy threshold, it suffices to consider the model in an infinite volume (where there is no nontrivial homology). In the ordered phase where errors are correctable, large fluctuations of domain walls or flux tubes are suppressed, while in the disordered phase the walls or tubes "dissolve" and cease to be well defined.

Thus, the phase transition corresponding to the accuracy threshold is a singularity, in the infinite-volume limit, in the "quenched" free energy, defined as

$$\langle \beta F[J,\eta] \rangle_p \equiv -\sum_{\{\eta\}} \mathrm{Prob}(\eta) \cdot \ln Z[J,\eta], \tag{4.29}$$

where

$$\mathrm{Prob}(\eta) = \prod_{\ell} (1-p)^{1-\eta_\ell} p^{\eta_\ell} \tag{4.30}$$

in two dimensions, or

$$\mathrm{Prob}(\eta) = \prod_{P} (1-p)^{1-\eta_P} p^{\eta_P} \tag{4.31}$$

in three dimensions. The term "quenched" signifies that, although the η chains are generated at random, we consider thermal fluctuations with the positions of the vortices or monopoles pinned down. The inverse temperature β is identical to the coupling J. We use the notation $\langle\cdot\rangle_p$ to indicate an average with respect to the quenched randomness, and we will denote by $\langle\cdot\rangle_\beta$ an average over thermal fluctuations.

There are various ways to describe the phase transition in this system, and to specify an order parameter. For example, in the two-dimensional Ising system, we may consider a "disorder parameter" $\Phi(x)$ that inserts a single Ising vortex at a specified position x. To define this operator, we must consider either an infinite system or a finite system with a boundary; on the torus, Ising vortices can only be inserted in pairs. But for a system with a boundary, we can consider a domain wall with one end at the boundary and one end in the bulk. In the *ferromagnetic* phase, the cost in free energy of introducing an additional vortex at x is proportional to L, the distance from x to the boundary. Correspondingly we find

$$\langle\langle\Phi(x)\rangle_\beta\rangle_p = 0 \tag{4.32}$$

in the limit $L \to \infty$. The disorder parameter vanishes because we cannot introduce an isolated vortex without creating an infinitely long domain wall. In the disordered phase, an additional vortex can be introduced at finite free energy cost, and hence

$$\langle\langle\Phi(x)\rangle_\beta\rangle_p \neq 0. \tag{4.33}$$

On the torus, we may consider an operator that inserts, not a semi-infinite domain wall terminating on a vortex, but instead a domain wall that winds about a cycle of the torus. Again, in the ferromagnetically ordered phase, the cost in free energy of inserting the domain wall will be proportional to L, the minimal length of a cycle. Specifically, in our two-dimensional Ising spin model, consider choosing

an η-chain and evaluating the corresponding partition function

$$Z[J,\eta] = \exp[-\beta F(J,\eta)]. \tag{4.34}$$

Now choose a set of links C of the original lattice that constitute a nontrivial cycle wound around the torus, and replace $\eta_\ell \to -\eta_\ell$ for the corresponding links of the dual lattice, $\ell \in C^*$. Evaluate, again, the partition function, obtaining

$$Z_C[J,\eta] = \exp[-\beta F_C(J,\eta)]. \tag{4.35}$$

Then the free energy cost of the domain wall is given by

$$\beta F_C(J,\eta) - \beta F(J,\eta) = -\ln\left(\frac{Z_C[J,\eta]}{Z[J,\eta]}\right). \tag{4.36}$$

After averaging over $\{\eta\}$, this free energy cost diverges as $L \to \infty$ in the ordered phase, and converges to a constant in the disordered phase.

There is also a dual order parameter that vanishes in the disordered phase—the spontaneous magnetization of the Ising spin system. Strictly speaking, the defining property of the non-ferromagnetic disordered phase is that spin correlations decay with distance, so that

$$\lim_{r\to\infty} \langle\langle \sigma_0 \sigma_r \rangle_\beta \rangle_p = 0 \tag{4.37}$$

in the disordered phase. Correspondingly, the mean squared magnetization per site

$$m^2 \equiv N^{-2} \sum_{i,j} \langle\langle \sigma_i \sigma_j \rangle_\beta \rangle_p, \tag{4.38}$$

where i, j are summed over all spins and N is the total number of spins, approaches a nonzero constant as $N \to \infty$ in the ordered phase, and approaches zero as a positive power of $1/N$ in the disordered phase.

Similarly in our three-dimensional gauge theory, there is a disorder parameter that inserts a single magnetic monopole, which we may think of as the end of a semi-infinite flux tube. Alternatively, we may consider the free energy cost of in-

serting a flux tube that wraps around the torus, which is proportional to L in the magnetically ordered phase. In the three-dimensional model, the partition function $Z_C[J, \eta]$ in the presence of a flux tube wrapped around the nontrivial cycle C of the original lattice is obtained by replacing $\eta_P \to -\eta_P$ on the plaquettes dual to the links of C. The magnetically ordered phase is called a "Higgs phase" or a "superconducting phase." The magnetically disordered phase is called a "confinement phase" because in this phase introducing an isolated electric charge has a infinite cost in free energy, and electric charges are confined in pairs by electric flux tubes.

An order parameter for the Higgs-confinement transition is the Wilson loop operator

$$W(C) = \prod_{\ell \in C} \sigma_\ell \tag{4.39}$$

associated with a closed loop C of links on the lattice. This operator can be interpreted as the insertion of a charged particle source whose world line follows the path C. In the confinement phase, this world line becomes the boundary of the world sheet of an electric flux tube, so that the free energy cost of inserting the source is proportional to the minimal area of a surface bounded by C; that is,

$$-\langle \ln \langle W(C) \rangle_\beta \rangle_p \tag{4.40}$$

increases like the area enclosed by the loop C in the confinement phase, while in the Higgs phase it increases like the perimeter of C.[2]

In the case $q \neq p$, our gauge theory becomes anisotropic—p controls the coupling and the quenched disorder on the timelike plaquettes, while q controls the coupling and the quenched disorder on the spacelike plaquettes. The tubes of flux

[2] A subtle point is that the relevant Wilson loop operator differs from that considered in Sec. 10 of [6]. In that reference, the Wilson loop was modified so that the "Dirac strings" connecting the monopoles would be invisible. But in our case, the Dirac strings have a physical meaning (they comprise the chain E) and we are genuinely interested in how far the physical flux tubes (comprising the chain E') fluctuate away from the Dirac strings!

in $E + E'$ will be stretched in the time direction for $q > p$ and compressed in the time direction for $q < p$. Correspondingly, spacelike and timelike Wilson loops will decay at different rates. Still, one expects that (for $0 < q < 1/2$) a single phase boundary in the p–q plane separates the region in which both timelike and spacelike Wilson loops decay exponentially with area (confinement phase) from the region in which both timelike and spacelike Wilson loops decay exponentially with perimeter. In the limit $q \to 0$, flux on the spacelike plaquettes becomes completely suppressed, and the timelike plaquettes on distinct time slices decouple, each described by the two-dimensional spin model described earlier. Similarly, in the limit $p \to 0$, the gauge theory reduces to decoupled one-dimensional spin models extending in the vertical direction, with a critical point at $q = 1/2$.

4.4.6 Accuracy threshold

What accuracy threshold can be achieved by surface codes? We have found that in the case where the syndrome is measured perfectly ($q = 0$), the answer is determined by the value of critical point of the two-dimensional random-bond Ising model on the Nishimori line. This value has been determined by numerically evaluating the domain wall free energy; a recent result of Honecker et al. is [55]

$$p_c = .1094 \pm .0002. \tag{4.41}$$

and an even more recent result of Merz and Chalker is [76]

$$p_c = .1093 \pm .0002. \tag{4.42}$$

A surface code is a Calderbank-Shor-Steane (CSS) code, meaning that each stabilizer generator is either a tensor product of X's or a tensor product of Z's [19, 95]. If X errors and Z errors each occur with probability p, then it is known that CSS codes exist with asymptotic rate $R \equiv k/n$ (where n is the block size and k is the number of encoded qubits) such that error recovery will succeed with

probability arbitrarily close to one, where

$$R = 1 - 2H_2(p); \tag{4.43}$$

here $H_2(p) = -p \log_2 p - (1-p)\log_2(1-p)$ is the binary Shannon entropy. This rate hits zero when p has the value

$$p_c = .1100, \tag{4.44}$$

which agrees with eq. (4.41) within statistical errors. Thus the critical error probability is (at least approximately) the same regardless of whether we allow arbitrary CSS codes or restrict to those with a locally measurable syndrome. This result is analogous to the property that the classical repetition code achieves reliable recovery from bit-flip errors for any error probability $p < 1/2$, the value for which the Shannon capacity hits zero. Note that eq. (4.41) can also be interpreted as a threshold for the quantum repetition code, in the case where the bit-flip error rate and the measurement error rate are equal ($p = q$).

If measurement errors are incorporated, then the accuracy threshold achievable with surface codes is determined by the critical point along the Nishimori line of the three-dimensional Z_2 gauge theory with quenched randomness. In that model the measurement error probability q (the error weight for vertical links) and the bit-flip probability p (the error weight for horizontal links) are independent parameters. It seems that numerical studies of this quenched gauge theory have not been done previously, even in the isotropic case; work on this problem is in progress.

Since recovery is more difficult with imperfect syndrome information than with perfect syndrome information, the numerical data on the random-bond Ising model indicate that $p_c < .11$ for any $q > 0$. For the case $p = q$, we will derive the lower bound $p_c \geq .0114$ in Sec. 4.5.

4.4.7 Free energy versus energy

In either the two-dimensional model (if $q = 0$) or the three-dimensional model (if $q > 0$), the critical error probability along the Nishimori line provides a criterion for whether it is possible in principle to perform flawless recovery from errors. In practice, we would have to execute a classical computation, with the measured syndrome as input, to determine how error recovery should proceed. The defects revealed by the syndrome measurement can be brought together to annihilate in several homologically distinct ways; the classical computation determines which of these "recovery chains" should be chosen.

We can determine the right homology class by computing the free energy for each homology class, and choosing the one with minimal free energy. In the ordered phase (error probability below threshold), the correct sector will be separated in free energy from other sectors by an amount linear in L, the linear size of the lattice.

The computation of the free energy could be performed by, for example, the Monte Carlo method. It should be possible to identify the homology class that minimizes the free energy in a time polynomial in L, unless the equilibration time of the system is exponentially long. Such a long equilibration time would be associated with spin-glass behavior—the existence of a large number of metastable configurations. In the random-bond Ising model, spin glass behavior is expected in the disordered phase, but not in the ferromagnetically ordered phase corresponding to error probability below threshold. Thus, we expect that in the two-dimensional model the correct recovery procedure can be computed efficiently for any $p < p_c$. Similarly, it is also reasonable to expect that, for error probability below threshold, the correct recovery chain can be found efficiently in the three-dimensional model that incorporates measurement errors.

In fact, there is reason to expect that when the error probability is below threshold, we can recover successfully by finding a recovery chain that minimizes *energy* rather than free energy. Nishimori [82] notes that along the Nishimori line, the free

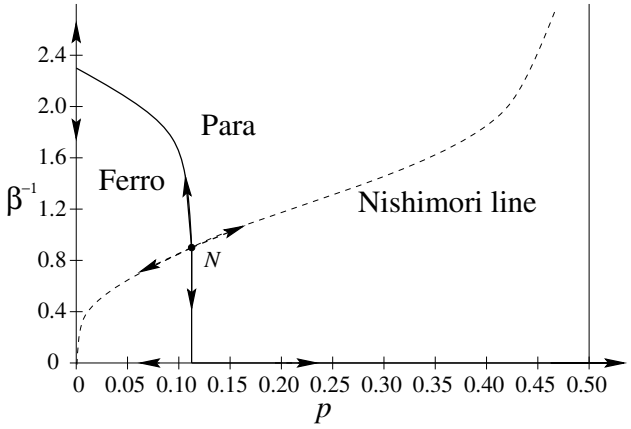

Figure 4.11: The phase diagram of the random-bond Ising model, with the temperature β^{-1} on the vertical axis and the probability p of an antiferromagnetic bond on the horizontal axis. The solid line is the boundary between the ferromagnetic (ordered) phase and the paramagnetic (disordered) phase. The dotted line is the Nishimori line $e^{-2\beta} = p/(1-p)$, which crosses the phase boundary at the Nishimori point N. From the point N to the horizontal axis, the phase boundary is *vertical*.

energy $\langle \beta F[J] \rangle_p$ coincides with the *entropy of frustration*; that is, the *Shannon entropy* of the distribution of Ising vortices, which does not depend on temperature. (He considered the isotropic two-dimensional model, but his argument applies just as well to our three-dimensional gauge theory, or to the anisotropic model with $q \neq p$.) Thus, the singularity of the free energy on the Nishimori line can be regarded as a singularity of this Shannon entropy, which is a purely geometrical effect having nothing to do with thermal fluctuations. This led Nishimori to suggest that the boundary between the ferromagnetic and paramagnetic phases below the Nishimori point (p_c, T_c) is vertical in the p–T plane (see Fig. 4.11). Kitatani also conjectured this verticality property, motivated by an "appropriate condition" [64]. If this conjecture were true, then the threshold p_{c0} at $T = 0$ would be the same as p_c, so energy-minimization recovery would be just as effective as free energy-minimization recovery.

Recent numerical results suggest that this conjecture is false, namely Kawashima and Aoki [57] compute the zero-temperature critical bond concentration for the random-bond Ising model to be

$$p_{c0} \simeq .105 \pm .002, \tag{4.45}$$

and Wang [104] computes the critical bond concentration to be

$$p_{c0} \simeq .1030 \pm .0002 \tag{4.46}$$

for the random-bond Ising model and

$$p_{c0} \simeq .0295 \pm .0002 \tag{4.47}$$

for the random-plaquette gauge model. While these results point to interesting new physics for these models [105], and prevent the threshold from being tightly calculated by an energy-minimization recovery algorithm, they are still useful in establishing bounds on the accuracy threshold. Nishimori [81] convincingly argued

that the phase boundary cannot extend to any value of of p greater than p_c and Le Doussal and Harris [32] argued that the tangent to the phase boundary is vertical at the Nishimori point. Hence, $p_{c0} < p_c$, so p_{c0} is a *lower bound* on the accuracy threshold.

Minimizing the energy has advantages. For one, the minimum energy configuration is the minimum weight chain with a specified boundary, which we know can be computed in a time polynomial in L using the perfect matching algorithm of Edmonds [33, 8]. Also, the minimum energy is easier to work with analytically—in Sec. 4.5 we will derive a rigorous bound on the accuracy threshold in our error model, by considering the efficacy of the energy minimization procedure in the three-dimensional model.

4.5 Chains of minimal weight

4.5.1 The most probable world line

As argued in Sec. 4.4.7, an effective way to use the error syndrome in our three-dimensional model is to construct an error chain that has the minimal "energy"—that is, we select from among all error chains that have the same boundary as the syndrome chain S, the single chain $E_{\min}$ that has the highest probability. In this Section, we will study the efficacy of this procedure, and obtain a lower bound on the accuracy threshold for quantum storage.

An error chain E with H horizontal links and V vertical links occurs with probability (aside from an overall normalization)

$$\left(\frac{p}{1-p}\right)^H \left(\frac{q}{1-q}\right)^V , \tag{4.48}$$

where p is the qubit error probability and q is the measurement error probability. Thus we choose $E_{\min}$ to be the chain with

$$\partial E_{\min} = \partial S \tag{4.49}$$

that has the *minimal* value of

$$H \cdot \log\left(\frac{1-p}{p}\right) + V \cdot \log\left(\frac{1-q}{q}\right); \tag{4.50}$$

we minimize the effective length (number of links) of the chain, but with horizontal and vertical links given different linear weights for $p \neq q$. If the minimal chain is not unique, one of the minimal chains is selected randomly.

Given the measured syndrome, and hence its boundary ∂S, the minimal chain $E_{\min}$ can be determined on a classical computer, using standard algorithms, in a time bounded by a polynomial of the number of lattice sites [33, 8]. If p and q are small, so that the lattice is sparsely populated by the sites contained in ∂S, this algorithm typically runs quite quickly. We assume this classical computation can be performed instantaneously and flawlessly.

4.5.2 A bound on chain probabilities

Recovery succeeds if our hypothesis $E_{\min}$ is homologically equivalent to the actual error chain E that generated the syndrome chain S, and fails otherwise. Hence, we wish to bound the likelihood of homologically nontrivial paths appearing in $E + E_{\min}$.

Consider a particular cycle on our spacetime lattice (or in fact any connected path, whether or not the path is closed). Suppose that this path contains H horizontal links and V vertical links. How likely is it that $E + E_{\min}$ contains this particular set of links?

For our particular path with H horizontal links and V vertical links, let H_m, V_m be the number of those links contained in $E_{\min}$, and let H_e, V_e be the number of those links contained in E (Cf. Fig. 4.12). These quantities obey the relations

$$H_m + H_e \geq H, \quad V_m + V_e \geq V, \tag{4.51}$$

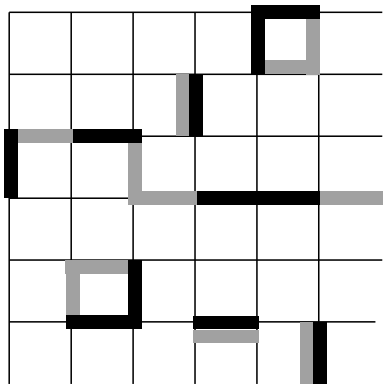

Figure 4.12: The error chain E (darkly shaded) and one possible choice for the chain $E_{\min}$ (lightly shaded), illustrated for a 6×6 torus in two dimensions. In this case $E + E_{\min}$ contains a homologically nontrivial cycle of length 8, which contains $H_e = 4$ links of E and $H_m = 4$ links of $E_{\min}$.

and so it follows that

$$\left(\frac{p}{1-p}\right)^{H_m}\left(\frac{q}{1-q}\right)^{V_m}\cdot\left(\frac{p}{1-p}\right)^{H_e}\left(\frac{q}{1-q}\right)^{V_e} \leq \left(\frac{p}{1-p}\right)^{H}\left(\frac{q}{1-q}\right)^{V}. \tag{4.52}$$

Furthermore, our procedure for constructing $E_{\min}$ ensures that

$$\left(\frac{p}{1-p}\right)^{H_e}\left(\frac{q}{1-q}\right)^{V_e} \leq \left(\frac{p}{1-p}\right)^{H_m}\left(\frac{q}{1-q}\right)^{V_m}. \tag{4.53}$$

This must be so because the e links and the m links share the same boundary; were eq. (4.53) not satisfied, we could replace the m links in $E_{\min}$ by the e links and thereby increase the value of $[p/(1-p)]^{H_m}[q/(1-q)]^{V_m}$. Combining the inequalities eq. (4.52) and eq. (4.53), we obtain

$$\left(\frac{p}{1-p}\right)^{H_e}\left(\frac{q}{1-q}\right)^{V_e} \leq \left[\left(\frac{p}{1-p}\right)^{H}\left(\frac{q}{1-q}\right)^{V}\right]^{1/2}. \tag{4.54}$$

What can we say about the probability $\mathrm{Prob}(H,V)$ that a particular connected path with (H,V) horizontal and vertical links is contained in $E+E_{\min}$? There are altogether 2^{H+V} ways to distribute errors (links contained in E) at locations on the specified chain—each link either has an error or not. And once the error locations are specified, the probability for errors to occur at those particular locations is

$$\begin{aligned} & p^{H_e}(1-p)^{H-H_e} q^{V_e}(1-q)^{V-V_e} \\ = \;& (1-p)^H (1-q)^V \left(\frac{p}{1-p}\right)^{H_e} \left(\frac{q}{1-q}\right)^{V_e}. \end{aligned} \tag{4.55}$$

But with those chosen error locations, the cycle can be in $E+E_{\min}$ only if eq. (4.54) is satisfied. Combining these observations, we conclude that

$$\mathrm{Prob}(H,V) \leq 2^{H+V} \left(\tilde{p}^H \tilde{q}^V\right)^{1/2}, \tag{4.56}$$

where

$$\tilde{p} = p(1-p), \quad \tilde{q} = q(1-q). \tag{4.57}$$

We can now bound the probability that $E+E_{\min}$ contains any connected path with (H,V) links (whether an open path or a cycle) by counting such paths. We may think of the path as a walk on the lattice (in the case of a cycle we randomly choose a point on the cycle where the walk begins and ends). Actually, our primary interest is not in how long the walk is (how many links it contains), but rather in how far it wanders—in particular we are interested in whether a closed walk is homologically nontrivial. The walks associated with connected chains of errors visit any given *link* at most once, but it will suffice to restrict the walks further, to be *self-avoiding walks* (SAW's)—those that visit any given *site* at most once (or in the case of a cycle, revisit only the point where the walk starts and ends). This restriction proves adequate for our purposes, because given any open error walk that connects two sites, we can always obtain an SAW by eliminating some closed loops of links from that walk. Similarly, given any homologically nontrivial closed walk, we can obtain a closed SAW (a *self-avoiding polygon*, or SAP) by eliminating

some links.

If we wish to consider the probability of an error per unit time in the encoded state, we may confine our attention to SAW's that lie between two time slices separated by the finite time T. (In fact, I will explain in Sec. 4.6 why we can safely assume that $T = O(L)$.) Such an SAW can begin at any one of $L^2 \cdot T$ lattice sites of our three-dimensional lattice (and in the case of an SAP, we may arbitrarily select one site that it visits as its "starting point.") If $n_{\rm SAP}(H, V)$ denotes the number of SAP's with (H, V) links and a specified starting site, then the probability $\mathrm{Prob}_{\rm SAP}(H, V)$ that $E + E_{\rm min}$ contains any SAP with (H, V) links satisfies

$$\mathrm{Prob}_{\rm SAP}(H, V) \leq L^2 T \cdot n_{\rm SAP}(H, V) \cdot 2^{H+V} \left(\tilde{p}^H \tilde{q}^V\right)^{1/2}. \tag{4.58}$$

The upper bound eq. (4.58) will be the foundation of the results that follow.

The encoded quantum information is damaged if $E + E_{\rm min}$ contains homologically nontrivial paths. At a minimum, the homologically nontrivial (self-avoiding) path must contain at least L horizontal links. Hence we can bound the failure probability as

$$\begin{aligned} \mathrm{Prob}_{\rm fail} &\leq \sum_V \sum_{H \geq L} \mathrm{Prob}_{\rm SAP}(H, V) \\ &\leq L^2 T \sum_V \sum_{H \geq L} n_{\rm SAP}(H, V) \cdot (4\tilde{p})^{H/2} (4\tilde{q})^{V/2}. \end{aligned} \tag{4.59}$$

4.5.3 Counting anisotropic self-avoiding walks

We will obtain bounds on the accuracy threshold for reliable quantum storage with toric codes by establishing conditions under which the upper bound eq. (4.59) rapidly approaches zero as L gets large. For this analysis, we will need bounds on the number of self-avoiding polygons with a specified number of horizontal and vertical links.

One such bound is obtained if we ignore the distinction between horizontal and vertical links. The first step of an SAP on a simple (hyper)cubic lattice in d

dimensions can be chosen in any of $2d$ directions, and each subsequent step in at most $2d-1$ directions, so for walks containing a total of ℓ links we obtain

$$n_{\mathrm{SAP}}^{(d)}(\ell) \leq 2d(2d-1)^{\ell-1}, \quad d \text{ dimensions.} \tag{4.60}$$

Some tighter bounds are known [101, 74] in the cases $d = 2, 3$:

$$n_{\mathrm{SAP}}^{(2)}(\ell) \leq P_2(\ell)(\mu_2)^{\ell}, \quad \mu_2 \approx 2.638, \tag{4.61}$$

and

$$n_{\mathrm{SAP}}^{(3)}(\ell) \leq P_3(\ell)(\mu_3)^{\ell}, \quad \mu_3 \approx 4.684, \tag{4.62}$$

where $P_{2,3}(\ell)$ are polynomials.

Since an SAP with H horizontal and V vertical links has $\ell = H + V$ total links, we may invoke eq. (4.62) together with eq. (4.59) to obtain

$$\begin{aligned} &\mathrm{Prob}_{\mathrm{fail}} \\ &\leq L^2 T \sum_V \sum_{H \geq L} P_3(H+V) \cdot (4\mu_3^2\ \tilde{p})^{H/2} (4\mu_3^2\ \tilde{q})^{V/2}. \end{aligned} \tag{4.63}$$

Provided that

$$\tilde{p} < (4\mu_3^2)^{-1}, \quad \tilde{q} < (4\mu_3^2)^{-1}, \tag{4.64}$$

we have

$$(4\mu_3^2\ \tilde{p})^{H/2} \cdot (4\mu_3^2\ \tilde{q})^{V/2} \leq (4\mu_3^2\ \tilde{p})^{L/2}, \tag{4.65}$$

for every term appearing in the sum. Since there are altogether $2L^2T$ horizontal links and L^2T vertical links on the lattice, the sum over H, V surely can have at most $2L^4T^2$ terms, so that

$$\mathrm{Prob}_{\mathrm{fail}} < Q_3(L,T) \cdot (4\mu_3^2\ \tilde{p})^{L/2} \tag{4.66}$$

where $Q_3(L,T)$ is a polynomial. To ensure that quantum information can be stored with arbitrarily good reliability, it will suffice that $\mathrm{Prob}_{\mathrm{fail}}$ becomes arbitrarily

small as L gets large (with T increasing no faster than a polynomial of L). Thus eq. (4.64) is sufficient for reliable quantum storage. Numerically, the accuracy threshold is surely attained provided that

$$\tilde{p}, \tilde{q} < (87.8)^{-1} = .0113, \tag{4.67}$$

or

$$p, q < .0114. \tag{4.68}$$

Not only does eq. (4.66) establish a lower bound on the accuracy threshold; it also shows that, below threshold, the failure probability decreases exponentially with L, the square root of the block size of the surface code.

Eq. (4.68) bounds the accuracy threshold in the case $p = q$, where the sum in eq. (4.59) is dominated by isotropic walks with $V \sim H/2$. But for $q < .0114$, higher values of p can be tolerated, and for $q > .0114$, there is still a threshold, but the condition on p is more stringent. To obtain stronger results than eq. (4.68) from eq. (4.59), we need better ways to count anisotropic walks, with a specified ratio of V to H.

One other easy case is the $q \to 0$ limit (perfect syndrome measurement), where the only walks that contribute are two-dimensional SAP's confined to a single time slice. Then we have

$$\mathrm{Prob}_{\mathrm{fail}} < Q_2(L,T) \cdot (4\ \mu_2^2\ \tilde{p})^{L/2} \tag{4.69}$$

(where $Q_2(L,T)$ is a polynomial) provided that

$$\tilde{p} = p(1-p) < (4\mu_2^2)^{-1} \approx (27.8)^{-1} = .0359, \tag{4.70}$$

or

$$p < .0373; \tag{4.71}$$

the threshold value of p can be relaxed to at least .0373 in the case where syndrome measurements are always accurate.

This estimate of p_c is considerably smaller than the value $p_c \simeq .1094 \pm .0002$ quoted in Sec. 4.4.6, obtained from the critical behavior of the random-bond Ising model. That discrepancy is not a surprise, considering the crudeness of our arguments in this section. If one accepts the results of the numerical studies of the random-bond Ising model, and Nishimori's argument that the phase boundary of the model is vertical, then apparently constructing the minimum weight chain is a more effective procedure than our bound indicates.

One possible way to treat the case $q \neq p$ would be to exploit an observation due to de Gennes [27], which relates the counting of SAP's to the partition function of a classical $O(N)$ spin model in the limit $N \to 0$. This spin model is anisotropic, with nearest-neighbor couplings J_H on horizontal links and J_V on vertical links, and its (suitably rescaled) free energy density has the high-temperature expansion

$$f(J_H, J_V) = \sum_{H,V} n_{\mathrm{SAP}}(H, V) \, (J_H)^H \, (J_V)^V \, . \tag{4.72}$$

This expansion converges in the disordered phase of the spin system, but diverges in the magnetically ordered phase. Thus, the phase boundary of the spin system in the J_H–J_V plane can be translated into an upper bound on the storage accuracy threshold in the p–q plane, through the relations

$$\tilde{p} = J_H^2/4, \quad \tilde{q} = J_V^2/4, \tag{4.73}$$

obtained by comparing eq. (4.72) and eq. (4.59).

To bound the failure probability for a planar code rather than the toric code, we should count the "relative polygons" that stretch from one edge of the lattice to the opposite edge. This change has no effect on the estimate of the threshold.

4.6 Error correction for a finite time interval

In estimating the threshold for reliable *storage* of encoded quantum information, we found it convenient to imagine that we perform error syndrome measurement

forever, without any beginning or end. Thus $S + E$ is a cycle (where S is the syndrome chain and E is the error chain) containing the closed world lines of the defects. Though some of these world lines may be homologically nontrivial, resulting in damage to the encoded qubits, we can recover from the damage successfully if the chain $S + E'$ (where E' is our estimated error chain) is homologically equivalent to $S + E$. The analysis is simplified because we need to consider only the errors that have arisen during preceding rounds of syndrome measurement, and need not consider any pre-existing errors that were present when the round of error correction began.

However, if we wish to perform a *computation* acting on encoded toric blocks, the analysis is more complicated. In our analysis of the storage threshold, we assumed that the complete syndrome history of an encoded block is known. But when two blocks interact with one another in the execution of a quantum gate, the defects in each block may propagate to the other block. Then to assemble a complete history of the defects in any given block, we would need to take into account the measured syndrome of all the blocks in the "causal past" of the block in question. In principle this is possible. But in practice, the required classical computation would be far too complex to perform efficiently—in T parallelized time steps, with two-qubit gates acting in each step, it is conceivable that defects from as many as 2^T different blocks could propagate to a given block. Hence, if we wish to compute fault-tolerantly using toric codes, we will need to intervene and perform recovery repeatedly. Since the syndrome measurement is imperfect and the defect positions cannot be precisely determined, errors left over from one round of error correction may cause problems in subsequent rounds.

Intuitively, it should not be necessary to store syndrome information for a very long period to recover successfully, because correlations decay exponentially with time in our statistical-mechanical model. To take advantage of this property, we must modify our recovery procedure.

4.6.1 Minimal-weight chains

Consider performing syndrome measurement T times in succession (starting at time $t = 0$), generating syndrome chain S and error chain E. Let the error chain E contain any qubit errors that were already present when the syndrome measurements began. Then the chain $S + E$ consisting of all defect world lines contains both closed loops and open paths that end on the final time slice—we say that $S + E$ is closed relative to the final time slice, or $\partial_{\rm rel}(S + E) = 0$. The open connected paths contained in $S + E$ are of two types: pairs of defects created prior to $t = 0$ that have persisted until $t = T$ (if the world line contains links on the initial time slice), and pairs of defects created after $t = 0$ that have persisted until $t = T$ (if the world line contains no links on the initial slice).

The syndrome S could have been caused by any error chain E' with the same *relative* boundary as E. To reconstruct the world lines, we should choose an E' that is likely given the observed S. A reasonable procedure is to choose the chain E' with $\partial_{\rm rel}E' = \partial_{\rm rel}S$ that minimizes the weight eq. (4.50).

The chain $S + E'$ can be projected onto the final time slice—the projected chain $\Pi(S + E')$ contains those and only those horizonal links that are contained in $S + E'$ on an odd number of time slices. Of course, E' has the same projection as $S + E'$; the syndrome chain S contains only vertical links so that its projection is trivial. The projection $\Pi(E')$ is our hypothesis about which links have errors on the final time slice. After $\Pi(E')$ is constructed, we may perform X's or Z's on these links to compensate for the presumed damage. Note that, to construct E', we do not need to store all of S in our (classical) memory—only the relative boundary of S is needed.

Actually, any homologically trivial closed loops in $\Pi(E')$ are harmless and can be safely ignored. Each homologically nontrivial world line modifies the encoded information by the logical operation $\bar{X}$ or $\bar{Z}$. Thus, after the hypothetical closed world lines are reconstructed, we may compensate for the homologically nontrivial closed loops by applying $\bar{X}$ and/or $\bar{Z}$ as needed. Projecting the open world lines

in E' onto the final time slice produces a pairing of the presumed positions of surviving defects on the final slice. These defects are removed by performing Z's or X's along a path connecting the pair that is homologically equivalent to the projected chain that connects them. Thus, this recovery step in effect brings the paired defects together to annihilate harmlessly.

Of course, our hypothesis E' won't necessarily agree exactly with the actual error chain E. Thus $E + E'$ contains open chains bounded by the final time slice. Where these open chains meet the final time slice, defects remain that our recovery procedure has failed to remove.

4.6.2 Overlapping recovery method

The procedure of constructing the minimal-weight chain E' with the same *relative* boundary as S is not as effective as the procedure in which we continue to measure the syndrome forever. In the latter case, we are in effect blessed with additional information about where monopoles will appear in the future, at times later than T, and that additional information allows us to make a more accurate hypothesis about the defect world lines. However, we can do nearly as well if we use a procedure that stores the syndrome history for only a finite time, if we recognize that the older syndrome is more trustworthy than the more recent syndrome. In our statistical physics model, the fluctuating closed loops in $E + E'$ do not grow indefinitely large in either space or in time. Therefore, we can reconstruct an E' that is homologically equivalent to E *quasilocally in time*—to pair up the monopoles in the vicinity of a given time slice, we do not need to know the error syndrome at times that are much earlier or much later.

So, for example, imagine measuring the syndrome $2T$ times in succession (starting at time $t = 0$), and then constructing E' with the same relative boundary as S. The chain E' can be split into two disjoint subchains, as indicated in Fig. 4.13. The first part consists of all connected chains that terminate on two monopoles, where both monopoles lie in the time interval $0 \le t < T$; call this part $E'_{\rm old}$. The rest of E' we call $E'_{\rm keep}$. To recover, we flip the links in the projection $\Pi(E'_{\rm old})$,

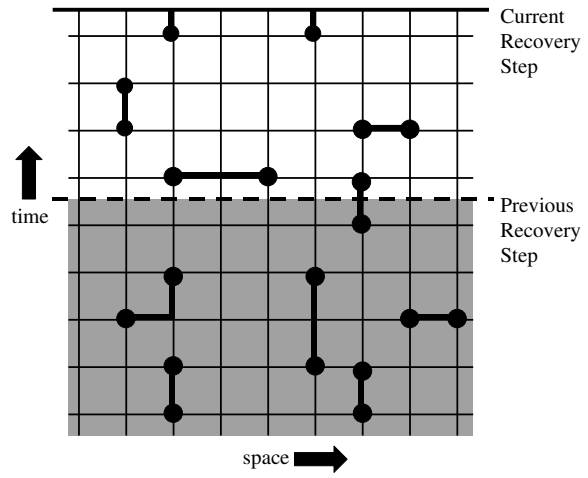

Figure 4.13: The "overlapping recovery" method, shown schematically. All monopoles (boundary points of the error syndrome chain) are indicated as filled circles, including both monopoles left over from earlier rounds of error recovery (those in the shaded region below the dotted line) and monopoles generated after the previous round (those in the unshaded region above the dotted line). Also shown is the minimum weight chain E' that connects each monopole to either another monopole or to the current time slice. The chain E' contains $E'_{\rm old}$, whose boundary lies entirely in the shaded region, and the remainder $E'_{\rm keep}$. In the current recovery step, errors are corrected on the horizontal links of $E'_{\rm old}$, and its boundary is then erased from the recorded syndrome history. The boundary of $E'_{\rm keep}$ is retained in the record, to be dealt with in a future recovery step.

after which we may erase from memory our record of the monopoles connected by $E'_{\rm old}$; only $E'_{\rm keep}$ (indeed only the relative boundary of $E'_{\rm keep}$) will be needed to perform the next recovery step.

In the next step we measure the syndrome another T times in succession, from $t = 2T$ to $t = 3T - 1$. Then we choose our new E' to be the minimal-weight chain whose boundary relative to the new final time slice is the union of the relative boundary of S in the interval $2T \le t < 3T$ and the relative boundary of $E'_{\rm keep}$ left over from previous rounds of error correction. We will call this procedure the "overlapping recovery method" because the minimal-weight chains that are constructed in successive steps occupy overlapping regions of spacetime.

If we choose T to be large compared to the characteristic correlation time of our statistical physics model, then only rarely will a monopole survive for more than one round, and the amount of syndrome information we need to store will surely be bounded. Furthermore, for such T, this overlapping recovery method will perform very nearly as well as if an indefinite amount of information were stored.

The time T should be chosen large enough so that connected chains in $E + E'$ are not likely to extend more than a distance T in the time direction. Arguing as in Sec. 4.5.3 (and recalling that the number $n_{\rm SAW}(\ell)$ of self-avoiding walks of length ℓ differs from the number $n_{\rm SAP}(\ell)$ of self-avoiding polygons of length ℓ by a factor polynomial in ℓ), we see that a connected chain containing H horizontal links and V vertical links occurs with a probability

$$\mathrm{Prob}(H, V) \le Q'_3(H, V)(4\mu_3^2\tilde{p})^{H/2}(4\mu_3^2\tilde{q})^{V/2}, \tag{4.74}$$

where $Q'_3(H, V)$ is a polynomial. Furthermore, a connected chain with temporal extent T must have at least $V = 2T$ vertical links if both ends of the chain lie on the final time slice. Therefore the probability $\mathrm{Prob}(H, V)$ is small compared to the failure probability eq. (4.66), so that our procedure with finite memory differs in efficacy from the optimal procedure with infinite memory by a negligible amount,

provided that

$$T \gg \frac{L}{2} \cdot \frac{\log(4\mu_3^2\tilde{p})^{-1}}{\log(4\mu_3^2\tilde{q})^{-1}}. \tag{4.75}$$

In particular, if the measurement error and qubit error probabilities are comparable ($q \simeq p$), it suffices to choose $T \gg L$, where L is the linear size of the lattice.

Thus we see that the syndrome history need not be stored indefinitely for our recovery procedure to be robust. The key to fault tolerance is that we should not overreact to syndrome information that is potentially faulty. In particular, if we reconstruct the world lines of the defects and find open world lines that do not extend very far into the past, it might be dangerous to accept the accuracy of these world lines and respond by bringing the defects together to annihilate. But world lines that persist for a time comparable to L are likely to be trustworthy. In our overlapping recovery scheme, we take action to remove only these long-lived defects, leaving those of more recent vintage to be dealt with in the next recovery step.

4.6.3 Computation threshold

Our three-dimensional model describes the history of a single code block; hence its phase transition identifies a threshold for reliable storage of quantum information. Analyzing the threshold for reliable quantum *computation* is more complex, because we need to consider interactions between code blocks.

When two encoded blocks interact through the execution of a gate, errors can propagate from one block to another, or potentially from one qubit in a block to another qubit in the same block. It is important to keep this error propagation under control. We will discuss in Sec. 4.9 how a universal set of fault-tolerant quantum gates can be executed on encoded states. For now let us consider the problem of performing a circuit consisting of CNOT gates acting on pairs of encoded qubits. The encoded CNOT gate with block 1 as its control and block 2 as its target can be implemented *transversally*—that is, by performing CNOT gates in parallel, each acting on a qubit in block 1 and the corresponding qubit in block

2. A CNOT gate propagates bit-flip errors from control to target and phase errors from target to control. Let us first consider the case in which storage errors occur at a constant rate, but errors in the gates themselves can be neglected.

Suppose that a transversal CNOT gate is executed at time $t = 0$, propagating bit-flip errors from block 1 to block 2, and imagine that we wish to correct the bit-flip errors in block 2. We suppose that many rounds of syndrome measurement are performed in both blocks before and after $t = 0$. Denote by S_1 and S_2 the syndrome chains in the two blocks, and by E_1 and E_2 the error chains. Due to the error propagation, the chain $S_2 + E_2$ in block 2 has a nontrivial boundary at the $t = 0$ time slice. Therefore, to diagnose the errors in block 2 we need to modify our procedure.

We may divide each syndrome chain and error chain into two parts, a portion lying in the past of the $t = 0$ time slice, and a portion lying in its future. Then the chain

$$\begin{aligned} &S_{1,\text{before}} + S_{2,\text{before}} + S_{2,\text{after}} \\ &+E_{1,\text{before}} + E_{2,\text{before}} + E_{2,\text{after}} \end{aligned} \tag{4.76}$$

has a trivial boundary. Therefore, we can estimate $E_{1,\text{before}} + E_{2,\text{before}} + E_{2,\text{after}}$ by constructing the minimal chain with the same boundary as $S_{1,\text{before}} + S_{2,\text{before}} + S_{2,\text{after}}$. Furthermore, because of the error propagation, it is $E_{1,\text{before}} + E_{2,\text{before}} + E_{2,\text{after}}$ whose horizontal projection identifies the damaged links in block 2 after $t = 0$.

If in each block the probability of error per qubit and per time step is p, while the probability of a syndrome measurement error is q, then the error chain $E_{1,\text{before}} + E_{2,\text{before}} + E_{2,\text{after}}$ has in effect been selected from a distribution in which the error probabilities are $(2p(1-p), 2q(1-q))$ before the gate, and (p, q) after the gate. Obviously, these errors are no more damaging than if the error probabilities had been $(2p(1-p), 2q(1-q))$ at all times, both before and after $t = 0$. Therefore, if (p, q) lies below the accuracy threshold for accurate storage, then error rates

$(2p(1-p), 2q(1-q))$ will be below the accuracy threshold for a circuit of CNOT gates.

Of course, the transversal CNOT might itself be prone to error, damaging each qubit with probability p_{CNOT}, so that the probability of error is larger on the $t = 0$ slice than on earlier or later slices. However, increasing the error probability from p to $p + p_{\mathrm{CNOT}}$ on a single slice is surely no worse than increasing the probability of error to $p + p_{\mathrm{CNOT}}$ on all slices. For a given q, there is a threshold value $p_c(q)$, such that for $p < p_c(q)$ a circuit of CNOT's is robust if the gates are flawless; then the circuit with imperfect gates is robust provided that $p + p_{\mathrm{CNOT}} < p_c(q)$.

By such reasoning, we can infer that the accuracy threshold for quantum computation is comparable to the threshold for reliable storage, differing by factors of order one. Furthermore, below threshold, the probability of error in an encoded gate decreases exponentially with L, the linear size of the lattice. Therefore, to execute a quantum circuit that contains T gates with reasonable fidelity, we should choose $L = O(\log T)$, so that the block size $2L^2$ of the code is $O(\log^2 T)$.

4.7 Quantum circuits for syndrome measurement

In our model with uncorrelated errors, in which qubit errors occur with probability p per time step and measurement errors occur with probability q, we have seen in Sec. 4.4 that it is possible to identify a sharp phase boundary between values of the parameters such that error correction is sure to succeed in the limit of a large code block, and values for which error correction need not succeed. How can we translate this accuracy threshold, expressed as a phase boundary in the p–q plane, into a statement about how well the hardware in our quantum memory must perform in order to protect quantum states effectively? The answer really depends on many details about the kinds of hardware that are potentially at our disposal. For purposes of illustration, we will relate p and q to the error probabilities for the fundamental gates in a particular computational model.

4.7.1 Syndrome measurement

Whenever a check operator X_s or Z_P is measured, a quantum circuit is executed in which each of the qubits occurring in the check operator interacts with an ancilla, and then the ancilla is measured to determine the result. Our task is to study this quantum circuit to determine how the faults in the circuit contribute to p and to q. To start we must decide what circuit to study.

For many quantum codes, the design of the syndrome measurement circuit involves subtleties. If the circuit is badly designed, a single error in the ancilla can propagate to many qubits in the code block, compromising the effectiveness of the error correction procedure. To evade this problem, Shor [93] and Steane [96] proposed two different methods for limiting the propagation of error from ancilla to data in the measurement of the check operators of a stabilizer code. In Shor's method, to extract each bit of the error syndrome, an ancilla "cat state" is prepared that contains as many qubits as the weight of the check operator. The ancilla interacts with the data code block, and then each qubit of the ancilla is measured; the value of the check operator is the parity of the measurement outcomes. In Steane's method, the ancilla is prepared as an encoded block (containing as many qubits as the length of the code). The ancilla interacts with the data, each qubit in the ancilla is measured, and a classical parity check matrix is applied to the measurement outcomes to extract the syndrome. In either scheme, each ancilla qubit interacts with only a single qubit in the data, so that errors in the ancilla cannot seriously damage the data. The price we pay is the overhead involved in preparing the ancilla states and verifying that the preparation is correct.

We could use the Shor method or the Steane method to measure the stabilizer of a surface code, but it is best not to. We can protect against errors more effectively by using just a single ancilla qubit for the measurement of each check operator, avoiding all the trouble of preparing and verifying ancilla states. The price we pay is modest—a single error in the ancilla might propagate to become two errors in the data, but we'll see that these correlated errors in the data are not so damaging.

So we imagine placing a sheet of ancilla qubits above the qubits of a planar code block. Directly above the site s is the ancilla qubit that will be used to measure X_s, and directly above the center of the plaquette P is the ancilla qubit that will be used to measure Z_P. We suppose that CNOT gates can be executed acting on a data qubit and its neighboring ancilla qubits. The circuits for measuring the plaquette operator $Z^{\otimes 4}$ and the site operator $X^{\otimes 4}$ are shown in Fig. 4.14:

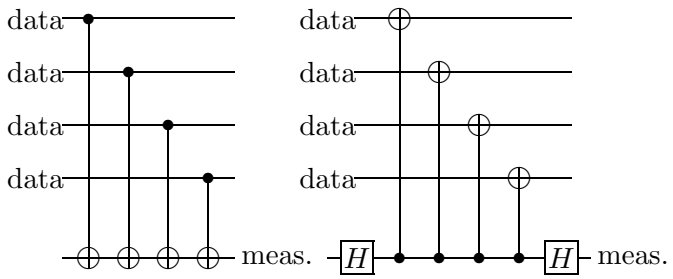

Figure 4.14: Circuits for measurement of the plaquette ($Z^{\otimes 4}$) and site ($X^{\otimes 4}$) stabilizer operators.

We have included the Hadamard gates in the circuit for measuring the site operator to signify that the ancilla qubit is initially prepared in the $X = 1$ state, and the final measurement is a measurement of X, while in the case of the plaquette operator measurement the ancilla is prepared in the $Z = 1$ state and Z is measured at the end. But we will suppose that our computer can measure X as easily as it can measure Z; hence in both cases the circuit is executed in six time steps (including preparation and measurement), and there is really no Hadamard gate.

4.7.2 Syndrome errors and data errors

We will assume that all errors in the circuit are stochastic (for example, they could be errors caused by decoherence). We will consider both "storage errors" and "gate errors." In each time step, the probability that a "resting" qubit is damaged will be denoted p_s. For simplicity, we will assume that an error, when it occurs, is one of the Pauli operators X, Y, or Z. (The analysis of the circuit is easily generalized to more general models of stochastic errors.) In our analysis, we will always make

a maximally pessimistic assumption about which error occurred at a particular position in the circuit. If a gate acts on a qubit in a particular time step, we will assume that there is still a probability p_s of a storage error in that step, plus an additional probability of error due to the execution of the gate. We denote the probability of an error in the two-qubit CNOT gate by p_{CNOT}; the error is a tensor product of Pauli operators, and again we will always make maximally pessimistic assumptions about which error occurs at a particular position in the circuit. If a storage error and gate error occur in the same time step, we assume that the gate error acts first, followed by the storage error. When a single qubit is measured in the $\{|0\rangle, |1\rangle\}$ basis, p_m is the probability of obtaining the incorrect outcome. (If a storage error occurs during a measurement step, we assume that the error precedes the measurement.) And when a fresh qubit is acquired in the state $|0\rangle$, p_p denotes the probability that its preparation is faulty (it is $|1\rangle$ instead).

In a single cycle of syndrome measurement, each data qubit participates in the measurement of four stabilizer operators: two site operators and two plaquette operators. Each of these measurements requires four time steps (excluding the preparation and measurement steps), as a single ancilla qubit is acted upon by four sequential CNOT's. But to cut down the likelihood of storage errors, we can execute the four measurement circuits in parallel, so that every data qubit participates in a CNOT gate in every step. For example, for each plaquette and each site, we may execute CNOT gates that act on the four edges of the plaquette or the four links meeting at the site in the counterclockwise order north-west-south-east. The CNOT gates that act on a given data qubit, then, alternate between CNOT's with the data qubit as control and CNOT's with the data qubit as target, as indicated in Fig. 4.15.

For either a site check operator or a plaquette check operator, the probability that the measurement is faulty is

$$q_{\text{single}} = p_p + 4p_{\text{CNOT}} + 6p_s + p_m + \text{h. o.}, \tag{4.77}$$

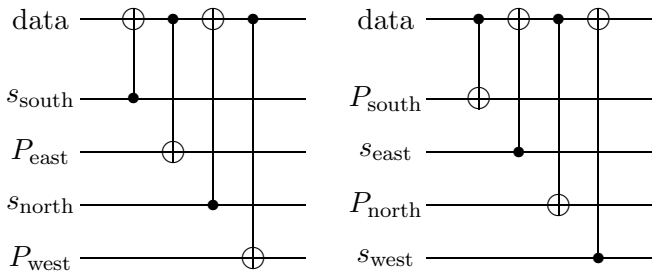

Figure 4.15: Gates acting on a given qubit in a complete round of syndrome measurement. Data qubits on links with a north-south orientation participate successively in measurements of check operators at the site to the south, the plaquette to the east, the site to the north, and the plaquette to the west. Qubits on links with an east-west orientation participate successively in measurements of check operators at the plaquette to the south, the site to the east, the plaquette to the north, and the site to the west.

where "+ h. o." denotes terms of higher than linear order in the fundamental error probabilities. The measurement can fail if any one of the CNOT gates has an error, if a storage error occurs during any of the six time steps needed to execute the circuit (including the preparation and measurement step), or because of a fault in the initial preparation or final measurement of the ancilla qubit. By omitting the higher order terms we are actually *overestimating* q. For example, p_s is the probability that a storage error occurs in the first time step, disregarding whether or not additional errors occur in the circuit.

I have used the notation q_{single} in eq. (4.77) to emphasize that this is an estimate of the probability of an isolated error on a vertical (timelike) link. More troublesome are syndrome measurement errors that are correlated with qubit errors. These arise if, say, a qubit suffers a Z error that is duly recorded in the syndrome measurement of one of the two adjoining sites but not the other. In our spacetime picture, then, there is a timelike plaquette with an error on one of its horizontal links and one of its vertical links. We will refer to this type of correlated error as a "vertical hook"—hook because the two links with errors meet at a 90°

angle, and vertical because one of the links is vertical (and to contrast with the case of a horizontal hook which will be discussed later).

We can estimate the probability of a vertical hook on a specified timelike plaquette by considering the circuits in Fig. 4.15. The qubit in question participates in the measurement of two site check operators, through the two CNOT gates in the circuit in which the data qubit is the target of the CNOT. A vertical hook can arise due to a fault that occurs in either of these CNOT gates or at a time in between the execution of these gates. Hence the probability of a vertical hook is

$$q_{\text{hook}} = 3p_{\text{CNOT}} + 2p_s + \text{h. o.}; \tag{4.78}$$

faults in any of three different CNOT gates, or storage errors in either of two time steps, can generate the hook. Note that the hook on the specified plaquette has a unique orientation; the first of the two site operator measurements that the data qubit participated in is the one that fails to detect the error. Of course, the same formula for q_{hook} applies if we are considering the measurement of the plaquette operators rather than the site operators.

A CNOT gate propagates X errors from control qubit to target qubit, and Z errors from target to control. Thus we don't have to worry about a vertical hook that arises from an error in an ancilla bit that propagates to the data. For example, if we are measuring a plaquette operator, then X errors in the ancilla damage the syndrome bit while Z errors in the ancilla propagate to the data; the result is a vertical error in the X-error syndrome that is correlated with a horizontal Z-error in the data. This correlation is not problematic because we deal with X errors and Z errors separately. However, propagation of error from ancilla to data also generates correlated horizontal errors that we need to worry about. In the measurement of, say, the plaquette operator $Z_P = Z^{\otimes 4}$, Z errors (but not X errors) can feed back from the ancilla to the data. Feeding back four Z's means no error at all, because $Z^{\otimes 4}$ is in the code stabilizer, and feeding back three Z's generates the error $IZZZ$, which is equivalent to the single Z error

$ZIII$. Therefore, the only way to get a double qubit error from a single fault in the circuit is through an error in the second or third CNOT, or through an ancilla storage error in between the second and third CNOT. (The second CNOT might apply Z to the ancilla but not to the data, and that Z error in the ancilla can then feed back to two data qubits, or the third CNOT could apply Z to both ancilla and data, and the Z error in the ancilla can then feed back to one other data qubit.) Because of the order we have chosen for the execution of the CNOT's, this double error, when it occurs, afflicts the southeast corner of the plaquette (or equivalently the northwest corner, which has the same boundary). We will refer to this two-qubit error as a "horizontal hook," because the two horizontal errors meet at a 90° angle. Similarly, error propagation during the measurement of the site operator X_s can produce X errors on the north and west links meeting at that site. One should emphasize that the only correlated XX or ZZ errors that occur with a probability linear in the fundamental error probabilities are these hooks. This is a blessing—correlated errors affecting two collinear links would be more damaging.

Feedback from the measurement of a plaquette operator can produce ZZ hooks but not XX hooks, and feedback from the measurement of a site operator can produce XX hooks but not ZZ hooks. Thus, in each round of syndrome measurement, the probability of a ZZ hook at a plaquette or an XX hook at a site is

$$p_{\text{hook}} = 2p_{\text{CNOT}} + p_s + \text{h. o.} \tag{4.79}$$

(Remember that a "hook" means two Z's or two X's; in addition, an error in a single CNOT gate could induce, say, an X error in the data and a Z error in the ancilla that subsequently feeds back, but correlated X and Z errors will not cause us any trouble.)

Now we need to count the ways in which a single error can occur in the data during a round of syndrome measurement. First suppose that we measure a single plaquette operator Z_P, and consider the scenarios that lead to a single Z error in

the data. The Z error can arise either because a gate or storage error damages the data qubit directly, or because an error in the ancilla feeds back to the data. Actually, single errors occur with slightly different probabilities for different data qubits acted on by the circuit. The worst case occurs for the first and last qubit acted on by the circuit; the probability that the circuit produces a single error that acts on the first (or last) qubit is

$$\begin{aligned} p_{\text{single,Z}}^{Z_P,1} &= p_{\text{single,Z}}^{Z_P,4} \\ &= p_{\text{CNOT}} + 6p_s \ + p_{\text{CNOT}} + p_s + \text{h. o.} \end{aligned} \tag{4.80}$$

The first two terms arise from gate errors and storage errors that damage the data qubit directly. For the first qubit, the last two terms arise from the case in which a Z error in the ancilla is fed back to the data by each of the last three CNOT's—the resulting $IZZZ$ error is equivalent to a $ZIII$ error because $ZZZZ$ is in the code stabilizer. For the fourth qubit, the last two terms arise from an error fed back by the last CNOT gate in the circuit. On the other hand, for the second and third qubit acted on by the circuit, it isn't possible for just a single error to feed back; *e.g.*, if the error feeds back to the third qubit, it will feed back to the fourth as well, and the result will be a hook instead of a single error. Hence, the probability of a single error acting on the second or third qubit is

$$p_{\text{single,Z}}^{Z_P,2} = p_{\text{single,Z}}^{Z_P,3} = p_{\text{CNOT}} + 6p_s + \text{h. o.}; \tag{4.81}$$

there is no feedback term. If we are measuring a site operator X_s, then X errors might feed back from the ancilla to the data, but Z errors will not. Therefore, for each of the four qubits acted on by the circuit, the probability that a single Z error results from the execution of the circuit, acting on that particular qubit, is

$$p_{\text{single,Z}}^{X_s} = p_{\text{CNOT}} + 6p_s + \text{h. o.}; \tag{4.82}$$

again there is no feedback term.

In a single round of syndrome measurement, each qubit participates in the measurement of four check operators, two site operators and two plaquette operators. For the plaquette operator measurements, depending on the orientation of the link where the qubit resides, the qubit will be either the first qubit in one measurement and the third in the other, or the second in one and the fourth in the other. Either way, the total probability of a single Z error arising that afflicts that qubit is

$$\begin{aligned} p_{\text{single}} &= 4p_{\text{CNOT}} + 6p_s + p_{\text{CNOT}} + p_s + \text{h. o.} \\ &= 5p_{\text{CNOT}} + 7p_s + \text{h. o.}, \end{aligned} \tag{4.83}$$

with the $4p_{\text{CNOT}} + 6p_s$ arising from direct damage to the qubit and the $p_{\text{CNOT}} + p_s$ from feedback due to one of the four check operator measurements. The same equation applies to the probability of a single X error arising at a given qubit in a single round of syndrome measurement.

4.7.3 Error-chain combinatorics

With both single errors and hooks to contend with, it is more complicated to estimate the failure probability, but we can still obtain useful upper bounds. In fact, the hooks don't modify the estimate of the accuracy threshold as much as might have been naively expected. Encoded information is damaged if $E + E_{\min}$ contains a homologically nontrivial (relative) cycle, which can wrap around the code block with either a north-south or east-west orientation. Either way, the cycle contains at least L links all with the *same* orientation, where L is the linear size of the lattice. A horizontal hook introduces two errors with *different* orientations, which is not as bad as two errors with the same orientation. Similarly, a vertical hook contains only one horizontal error.

There are two other reasons why the hooks do not badly compromise the effectiveness of error correction. While single errors can occur with any orientation, horizontal hooks can appear only on the northwest corner of a plaquette (hooks

on southeast corners are equivalent to hooks on northwest corners and should not be counted separately), and vertical hooks on timelike plaquettes have a unique orientation, too. Therefore, hooks have lower "orientational entropy" than the single errors, which means that placing hooks on self-avoiding walks reduces the number of walks of a specified length. And finally, $p_{\rm hook}$ is smaller than $p_{\rm single}$, and $q_{\rm hook}$ is smaller than $q_{\rm single}$, which further reduces the incentive to include hooks in $E + E_{\rm min}$.

We will suppose that $E_{\rm min}$ is constructed by the same procedure as before, by minimizing the weight

$$H \log p_{\rm single}^{-1} + V \log q_{\rm single}^{-1}. \tag{4.84}$$

To simplify later expressions, we have replaced $p/(1-p)$ by p here, which will weaken our upper bound on the failure probability by an insignificant amount. Note that our procedure finds the most probable chain under the assumption that only single errors occur (no hooks). If $p_{\rm hook}$ and $q_{\rm hook}$ are assumed to be known, then in principle we could retool our recovery procedure by taking these correlated errors into account in the construction of $E_{\rm min}$. To keep things simple we won't attempt to do that. Then, as before, for any connected subchain of $E + E_{\rm min}$ with H horizontal links and V vertical links, the numbers H_e and V_e of horizontal and vertical links of the subchain that are contained in E must satisfy

$$p_{\rm single}^{H_e} q_{\rm single}^{V_e} \leq p_{\rm single}^{H/2} q_{\rm single}^{V/2}. \tag{4.85}$$

To bound the failure probability, we wish to count the number of ways in which a connected chain with a specified number of horizontal links can occur in $E+E_{\rm min}$, keeping in mind that the error chain E could contain hooks as well as single errors. Notice that a hook might contribute only a single link to $E+E_{\rm min}$, if one of the links contained in the hook is also in $E_{\rm min}$. But since $p_{\rm hook} < p_{\rm single}$ and $q_{\rm hook} < q_{\rm single}$, we will obtain an upper bound on the failure probability if we pessimistically assume that all of the errors in $E + E_{\rm min}$ are either two-link hooks occurring with probabilities $p_{\rm hook}, q_{\rm hook}$ or single errors occurring with probabilities $p_{\rm single}, q_{\rm single}$.

If the H_e horizontal errors on a connected chain include $H_{\rm hook}$ horizontal hooks and $V_{\rm hook}$ vertical hooks, then there are $H_e - 2H_{\rm hook} - V_{\rm hook}$ single horizontal errors and $V_e - V_{\rm hook}$ single vertical errors; once the locations of the hooks and the single errors are specified, the probability that errors occur at those locations is no larger than

$$\begin{aligned}
&(p_{\rm single})^{H_e - 2H_{\rm hook} - V_{\rm hook}} (p_{\rm hook})^{H_{\rm hook}} \\
&\quad \cdot (q_{\rm single})^{V_e - V_{\rm hook}} (q_{\rm hook})^{V_{\rm hook}} \\
&< p_{\rm single}^{H/2} \left(\frac{p_{\rm hook}}{p_{\rm single}^2} \right)^{H_{\rm hook}} q_{\rm single}^{V/2} \left(\frac{q_{\rm hook}}{p_{\rm single} q_{\rm single}} \right)^{V_{\rm hook}} .
\end{aligned} \tag{4.86}$$

Because a horizontal hook contains two errors with different orientations, it will be convenient to distinguish between links oriented east-west and links oriented north-south. We denote by H_1 the number of horizontal links in the connected chain with east-west orientation and by H_2 the number of horizontal links with north-south orientation; then clearly

$$H_{\rm hook} \leq H_1, \quad H_{\rm hook} \leq H_2. \tag{4.87}$$

To estimate the threshold, we will bound the probability that our connected chain has $H_1 \geq L$; of course, the same expression bounds the probability that $H_2 \geq L$.

For a specified connected chain, suppose that altogether H_e of the horizontal links and V_e of the vertical links have errors, and that there are $H_{\rm hook}$ horizontal hooks and $V_{\rm hook}$ vertical hooks, so that there are $H_e - 2H_{\rm hook} - V_{\rm hook}$ single horizontal errors and $V_e - V_{\rm hook}$ single vertical errors. In how many ways can we distribute the hooks and single errors along the path? Since each horizontal hook contains a link with north-south orientation, there are no more than $\binom{H_2}{H_{\rm hook}}$ ways to choose the locations of the horizontal hooks; similarly there are no more than $\binom{V}{V_{\rm hook}}$ ways to choose the locations of the vertical hooks.[3] Then there are no

[3] Actually, we have given short shrift here to a slight subtlety. Once we have decided that a vertical hook will cover a particular vertical link, there may be two ways to place the hook—it

more than $2^{H_1+H_2-2H_{\rm hook}-V_{\rm hook}}$ ways to place the single horizontal errors among the remaining horizontal links, and no more than $2^{V-V_{\rm hook}}$ ways to place the single vertical errors among remaining $V - V_{\rm hook}$ vertical links on the chain. Now consider counting the self-avoiding paths starting at a specified site, where the path is constructed from hooks, single errors, and the links of $E_{\rm min}$. Whenever we add a horizontal hook to the path there are at most two choices for the orientation of the hook, and whenever we add a vertical hook there are at most four choices; hence there are no more than $2^{H_{\rm hook}}4^{V_{\rm hook}}$ ways to choose the orientations of the hooks. For the remaining $H_1+H_2-2H_{\rm hook}+V-2V_{\rm hook}$ links of the path, the orientation can be chosen in no more than 5 ways. Hence, the total number of paths with a specified number of horizontal links, horizontal hooks, vertical links, and vertical hooks is no more than

$$\begin{pmatrix} H_2 \\ H_{\rm hook} \end{pmatrix} \quad \begin{pmatrix} V \\ V_{\rm hook} \end{pmatrix} \cdot 2^{H_1+H_2-2H_{\rm hook}-V_{\rm hook}} 2^{V-V_{\rm hook}} \cdot 2^{H_{\rm hook}} 4^{V_{\rm hook}} \cdot 5^{H_1+H_2-2H_{\rm hook}+V-2V_{\rm hook}}. \tag{4.88}$$

Combining this counting of paths with the bound eq. (4.86) on the probability of each path, we conclude that the probability that $E + E_{\rm min}$ contains a connected path with specified starting site, containing H_1 links with east-west orientation, H_2 links with north-south orientation, V vertical links, $H_{\rm hook}$ horizontal hooks, and $V_{\rm hook}$ vertical hooks is bounded above by

$$\begin{pmatrix} H_2 \\ H_{\rm hook} \end{pmatrix} \left(\frac{p_{\rm hook}}{50 p_{\rm single}^2} \right)^{H_{\rm hook}} (100 p_{\rm single})^{(H_1+H_2)/2} \cdot \begin{pmatrix} V \\ V_{\rm hook} \end{pmatrix} \left(\frac{q_{\rm hook}}{25 p_{\rm single} q_{\rm single}} \right)^{V_{\rm hook}} \cdot (100 q_{\rm single})^{V/2}. \tag{4.89}$$

might cover either one of two adjacent horizontal links. However, for the hook to be free to occupy either position, the orientation of the second horizontal link must be chosen in one of only two possible ways. Thus the freedom to place the hook in two ways is more than compensated by the reduction in the orientational freedom of the other horizontal link by a factor of 2/5, and can be ignored. A similar remark applies to horizontal hooks.

Here H_{hook} can take any value from zero to H_2, and V_{hook} can take any value from zero to V. We can sum over H_{hook} and V_{hook}, to obtain an upper bound on the probability of a chain with an unspecified number of hooks:

$$\begin{aligned}&(100p_{\text{single}})^{(H_1+H_2)/2}\left(1+\frac{p_{\text{hook}}}{50p_{\text{single}}^2}\right)^{H_2}\\&\cdot(100q_{\text{single}})^{V/2}\left(1+\frac{q_{\text{hook}}}{25p_{\text{single}}q_{\text{single}}}\right)^{V}.\end{aligned}\tag{4.90}$$

Finally, since a path can begin at any of L^2T sites, and since there are two types of homologically nontrivial cycles, the probability of failure $\text{Prob}_{\text{fail}}$ satisfies the bound

$$\begin{aligned}\text{Prob}_{\text{fail}} &< 2L^2T\sum_{H_1\geq L}(100p_{\text{single}})^{H_1/2}\\&\cdot\sum_{H_2\geq 0}\left[100p_{\text{single}}\left(1+\frac{p_{\text{hook}}}{50p_{\text{single}}^2}\right)^2\right]^{H_2/2}\\&\cdot\sum_{V\geq 0}\left[100q_{\text{single}}\left(1+\frac{q_{\text{hook}}}{25p_{\text{single}}q_{\text{single}}}\right)^2\right]^{V/2}.\end{aligned}\tag{4.91}$$

This sum will be exponentially small for large L provided that

$$\begin{aligned}&p_{\text{single}}<\frac{1}{100},\quad q<\frac{1}{100},\\&p_{\text{hook}}<5\;p_{\text{single}}^2\left(\frac{1}{\sqrt{p_{\text{single}}}}-10\right),\\&q_{\text{hook}}<\frac{5}{2}\;p_{\text{single}}q_{\text{single}}\left(\frac{1}{\sqrt{q_{\text{single}}}}-10\right).\end{aligned}\tag{4.92}$$

Of course, making p_{single} and q_{single} smaller can only make things better. Our conditions on p_{hook} and q_{hook} in eq. (4.92) are not smart enough to know this—for p_{single} sufficiently small, we find that making it still smaller gives us a *more* stringent condition on p_{hook}, and similarly for q_{hook}. Clearly, this behavior is an artifact of our approximations. Thus, for a given p_{single} and q_{single}, we are free to choose any smaller values of p_{single} and q_{single} in order to obtain more liberal

conditions on $p_{\rm hook}$ and $q_{\rm hook}$ from eq. (4.92). Our expression that bounds $p_{\rm hook}$ achieves its maximum for $p_{\rm single} = (3/40)^2$, and for fixed $p_{\rm single}$, our expression that bounds $q_{\rm hook}$ achieves its maximum for $q_{\rm single} = (1/20)^2$. We therefore conclude that for recovery to succeed with a probability that approaches one as the block size increases, it suffices that

$$\begin{aligned} p_{\rm single} &< \frac{9}{1600}, \quad q_{\rm single} < \frac{1}{400}, \\ p_{\rm hook} &< \frac{3}{32} \cdot \frac{9}{1600}, \quad q_{\rm hook} < \frac{1}{16} \cdot \frac{9}{1600}. \end{aligned} \tag{4.93}$$

Comparing to our expressions for $q_{\rm single}$, $p_{\rm single}$, and $p_{\rm hook}$, we see that, unless $q_{\rm single}$ is dominated by preparation or measurement errors, these conditions are all satisfied provided that

$$q_{\rm hook} = 3p_{\rm CNOT} + 2p_s < 3.5 \times 10^{-4}. \tag{4.94}$$

If the probability of a CNOT error is negligible, then we obtain a lower bound on the critical error probability for storage errors,

$$(p_s)_c > 1.7 \times 10^{-4}. \tag{4.95}$$

In view of the crudeness of our combinatorics, we believe that this estimate is rather conservative, if one accepts the assumptions of our computational model.

4.8 Measurement and encoding

4.8.1 Measurement

At the conclusion of a quantum computation, we need to measure some qubits. If the computation is being executed fault tolerantly, this means measuring an encoded block. How can we perform this measurement fault tolerantly?

Suppose we want to measure the logical operator $\bar{Z}$; that is, measure the encoded block in the basis $\{|\bar{0}\rangle, |\bar{1}\rangle\}$. If we are willing to destroy the encoded block,

we first measure Z for each qubit in the block, projecting each onto the basis $\{|0\rangle, |1\rangle\}$. Were there no errors in the code block at the time of the measurement, and were all measurements of the individual qubits performed flawlessly, then we could choose any homologically nontrivial path on the lattice and evaluate the parity of the outcomes for the links along that path. Even parity indicates that the encoded block is in the state $|\bar{0}\rangle$, odd parity the state $|\bar{1}\rangle$.

But the code block *will* contain some errors (not too many, we hope), and some of the measurements of the individual qubits *will* be faulty. Since a single bit flip along the path could alter the parity of the measurement outcomes, we need to devise a fault-tolerant procedure for translating the observed values of the individual qubits into a value of the encoded qubit.

One such procedure is to evaluate the parity $Z^{\otimes 4}$ of the measurement outcomes at each plaquette of the lattice, determining the locations of all plaquette defects. These defects can arise either because defects were already present in the code block before the measurement, or they could be introduced by the measurement itself. It is useful and important to recognize that the defects introduced by the measurement do not pose any grave difficulties. An isolated measurement error at a single link will produce two neighboring defects on the plaquettes that contain that link. Widely separated defects can arise from the measurement only if there are many correlated measurement errors.

Therefore we can apply a suitable classical algorithm to remove the defects, for example, by choosing a chain of minimal total length that is bounded by the defect locations, which can be found in a polynomial-time classical computation. Flipping the bits on this chain corrects the errors in the measurement outcomes, so that we can then proceed to evaluate the parity along a nontrivial cycle. Assuming sufficiently small rates for the qubit and measurement errors, the encoded qubit will be evaluated correctly, with a probability of error that is exponentially small for large block size.

We can measure $\bar{X}$ by the same procedure, by measuring X for each qubit, and evaluating all site operators $X^{\otimes 4}$ from the outcomes. After removal of the site

defects by flipping bits appropriately, $\bar{X}$ is the parity along a nontrivial cycle of the dual lattice.

To measure $\bar{Z}$ of a code block without destroying the encoded state, we can prepare an ancilla block in the encoded state $|\bar{0}\rangle$, and perform a bitwise CNOT from the block to be measured into the ancilla. Then we can measure the ancilla by the destructive procedure just described. A nondestructive measurement of $\bar{X}$ is executed similarly.

4.8.2 Encoding of known states

At the beginning of a quantum computation, we need to prepare encoded qubits in eigenstates of the encoded operations, for example, the state $|\bar{0}\rangle$ of the planar code, a $\bar{Z} = 1$ eigenstate. If syndrome measurement were perfectly reliable, the state $|\bar{0}\rangle$ could be prepared quickly by the following method: Start with the state $|0\rangle^{\otimes n}$ where n is the block size of the code. This is the simultaneous eigenstate with eigenvalue 1 of all plaquette stabilizer operators $Z_P = Z^{\otimes 4}$ and of the logical operator $\bar{Z}$, but not of the site stabilizer operators $X_s = X^{\otimes 4}$. Then measure all the site operators. Since the site operators commute with the plaquette operators and the logical operators, this measurement does not disturb their values. About half of the site measurements have outcome $X_s = 1$ and about half have outcome $X_s = -1$; to obtain the state $|\bar{0}\rangle$, we must remove all of the site defects (sites where $X_s = -1$). Thus we select an arbitrary 1-chain whose boundary consists of the positions of all site defects, and we apply Z to each link of this chain, thereby imposing $X_s = 1$ at each site. In carrying out this procedure, we might apply $\bar{Z}$ to the code block by applying Z to a homologically nontrivial path, but this has no effect since the state is a $\bar{Z} = 1$ eigenstate.

Unfortunately, syndrome measurement is not perfectly reliable; therefore this procedure could generate long *open* chains of Z errors in the code block. To keep the open chains under control, we need to repeat the measurement of both the X and Z syndromes of order L times (where L is the linear size of the lattice), and use our global recovery method. Then the initial configuration of the defects will

be "forgotten" and the error chains in the code block will relax to the equilibrium configuration in which long open chains are highly unlikely. The probability of an $\bar{X}$ error that causes a flip of the encoded state will be exponentially small in L. We can prepare the encoded state with $\bar{X} = 1$ by the dual procedure, starting with the state $[\frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)]^{\otimes n}$.

4.8.3 Encoding of unknown states

Quantum error-correcting codes can protect *unknown* coherent quantum states. This feature is crucial in applications to quantum computation—the operator of a quantum computer need not "monitor" the encoded quantum state to keep the computation on track. But to operate a quantum computer, we don't typically need to *encode* unknown quantum states. It is sufficient to initialize the computer by encoding known states, and then execute a known quantum circuit.

Still, a truly robust "quantum memory" should be able to receive an unknown quantum state and store it indefinitely. But given any nonzero rate of decoherence, to store an unknown state for an indefinitely long time we need to encode it using a code of indefinitely long block size. How, then, can we expect to encode the state before it decoheres?

The key is to encode the state quickly, providing some measure of protection, while continuing to build up toward larger code blocks. Concatenated codes provide one means of achieving this. We can encode, perform error correction, then encode again at the next level of concatenation. If the error rates are small enough, encoding can outpace the errors so that we can store the unknown state in a large code block with reasonable fidelity.

The surface codes, too, allow us to build larger codes from smaller codes and so to protect unknown states effectively. The key to enlarging the code block is that a code corresponding to one triangulation of a surface can be transformed into a code corresponding to another triangulation.

For example, we can transform one surface code to another using local moves shown in Fig. 4.16:

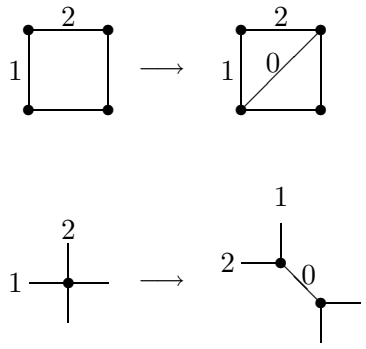

Figure 4.16: Two basic moves that modify the triangulation of a surface by adding a link: splitting a plaquette, and splitting a vertex.

Links can be added to (or removed from) the triangulation in either of two ways—one way adds a new plaquette, the other adds a new site. Either way, the new triangulation corresponds to a new code with an additional qubit in the code block and an additional stabilizer generator.

When a new plaquette is added, the new code stabilizer is obtained from the old one by adding the new plaquette operator

$$Z_1 Z_2 Z_0 \tag{4.96}$$

and by modifying the site operators with the replacements

$$X_1 \to X_1 X_0, \quad X_2 \to X_2 X_0. \tag{4.97}$$

When a new site is added, the stabilizer is modified similarly, but with X's and Z's interchanged:

$$X_1 X_2 X_0, \tag{4.98}$$

is a new stabilizer generator, and the existing plaquette operators are modified as

$$Z_1 \to Z_1 Z_0, \quad Z_2 \to Z_2 Z_0. \tag{4.99}$$

To add a plaquette or a site to a stabilizer code, we prepare the additional qubit in a $Z_0 = 1$ or $X_0 = 1$ eigenstate, and then execute the circuit shown in Fig. 4.17. We recall that, acting by conjugation, a CNOT gate changes a tensor product of Pauli operators acting on its control and target according to

$$IZ \leftrightarrow ZZ, \quad XI \leftrightarrow XX; \tag{4.100}$$

that is, the CNOT transforms an IZ eigenstate to a ZZ eigenstate and an XI eigenstate to an XX eigenstate, while leaving ZI and IX eigenstates invariant. The circuit in Fig. 4.17 with qubit 0 as target, then, transforms the site operators as in eq. (4.97) while also implementing

$$Z_0 \to Z_1 Z_2 Z_0. \tag{4.101}$$

The initial $Z_0 = 1$ eigenstate is transformed into a state that satisfies the plaquette parity checks of the new triangulation. Similarly the circuit in Fig. 4.17 with qubit 0 as control implements eq. (4.99) as well as

$$X_0 \to X_1 X_2 X_0; \tag{4.102}$$

the circuit transforms the $X_0 = 1$ eigenstate into a state that satisfies the new site parity checks.

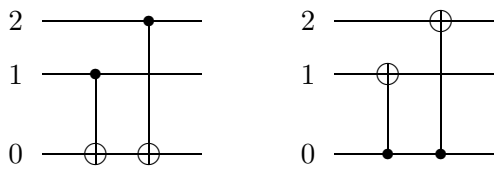

Figure 4.17: Circuits that implement the two basic moves of Fig. 4.16. The circuit with qubit 0 as the target of the CNOT's adds a plaquette; the circuit with qubit 0 as the control of the CNOT's adds a site.

Of course, these circuits are reversible; they can be used to extricate qubits

from a stabilizer code instead of adding them.

If planar codes are used, we can lay out the qubits in a planar array. Starting with a small encoded planar block in the center, we can gradually add new qubits to the boundary using the moves shown in Fig. 4.18:

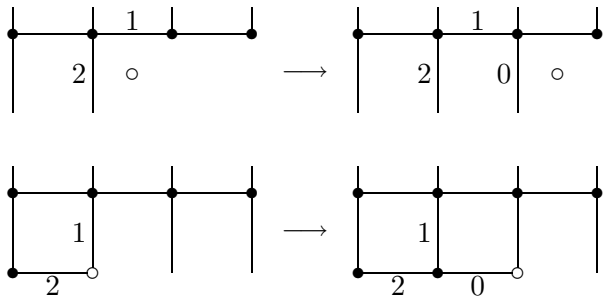

Figure 4.18: The same circuits as in Fig. 4.17 can also be used to build up a planar code by adding a link at the boundary. Sites or plaquettes marked by open circles do not correspond to stabilizer operators.

These moves add a new three-qubit plaquette or site operator, and can also be implemented by the circuits of Fig. (4.17).

A procedure that transforms a distance-L planar code to a distance-$(L+1)$ code is shown in Fig. 4.19. By adding a new row of plaquette operators, we transform what was formerly a smooth edge into a rough edge, and by adding a new row of site operators we transform a rough edge to a smooth edge. We start the row of plaquettes by adding a two-qubit plaquette operator to the corner via the transformations

$$Z_0 \to Z_1 Z_0, \quad X_1 \to X_1 X_0, \tag{4.103}$$

which can be implemented by a single CNOT; similarly, we start a row of sites by adding a two-qubit site operator with

$$X_0 \to X_1 X_0, \quad Z_1 \to Z_1 Z_0. \tag{4.104}$$

Then a new row of boundary stabilizer operators can be "zipped" into place.

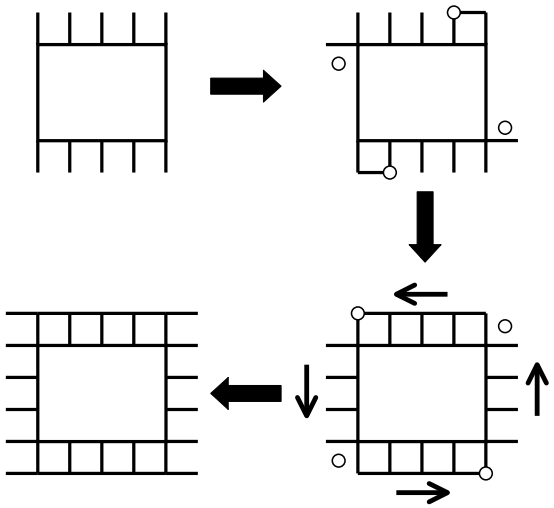

Figure 4.19: Building a distance-$(L+1)$ planar code by adding qubits to a distance-L planar code. (Here, $L = 5$.) In the first step, new two-qubit stabilizer operators are added in the corners with single CNOT's; in subsequent steps, three-qubit stabilizer operators are added with double CNOT's. The last step promotes the corner operators to three-qubit operators.

As is typical of encoding circuits, this procedure can propagate errors badly; a single faulty CNOT can produce a long row of qubit errors (a widely separated pair of defects) along the edge of the block. To ensure fault tolerance, we must measure the boundary stabilizer operators frequently during the procedure. Examining the syndrome record, we can periodically identify the persistent errors and remove them before proceeding to add further qubits.

4.9 Fault-tolerant quantum computation

We will now consider how information protected by planar surface codes can be processed fault-tolerantly. Our objective is to show that a universal set of fault-tolerant encoded quantum gates can be realized using only local quantum gates

among the fundamental qubits and with only polynomial overhead. We will describe one gate set with this property [60, 62]. This construction suffices to show that there is an accuracy threshold for quantum computation using surface codes: each gate in our set can be implemented acting on encoded states with arbitrarily good fidelity, in the limit of a large code block. The calculation of the numerical value of this computation threshold remains an open problem. Better implementations of fault-tolerant quantum computation can probably be found, requiring less overhead and yielding a better threshold.

We choose the basis introduced by Shor [93], consisting of four gates. Three of these generate the "symplectic" or "normalizer" group, the finite subgroup of the unitary group that, acting by conjugation, takes tensor products of Pauli operators to tensor products of Pauli operators. Of these three, two are single-qubit gates: the Hadamard gate

$$H = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}, \tag{4.105}$$

which acts by conjugation on Pauli operators according to

$$H : X \leftrightarrow Z, \tag{4.106}$$

and the phase gate

$$P \equiv \Lambda(i) = \begin{pmatrix} 1 & 0 \\ 0 & i \end{pmatrix}, \tag{4.107}$$

which acts by conjugation on Pauli operators according to

$$P : X \to Y, \quad Z \to Z. \tag{4.108}$$

The third generator of the normalizer group is the two-qubit CNOT $= \Lambda(X)$ gates, which acts by conjugation on Pauli operators according to

$$\begin{aligned} \text{CNOT} : \quad & XI \to XX, \qquad IX \to IX, \\ & ZI \to ZI, \qquad IZ \to ZZ. \end{aligned} \tag{4.109}$$

Quantum computation in the normalizer group is no more powerful than classical computation [49]. To realize the full power of quantum computing we need to complete the basis with a gate outside the normalizer group. This gate can be chosen to be the three-qubit Toffoli gate $T \equiv \Lambda^2(X)$, which acts on the standard three-qubit orthonormal basis $\{|a, b, c\rangle\}$ as

$$T : |a, b, c\rangle \rightarrow |a, b, c \oplus ab\rangle. \tag{4.110}$$

4.9.1 Normalizer gates for surface codes

CNOT gate

Implementing normalizer computation on planar codes is relatively simple. First of all, a planar surface code is a Calderbank-Shor-Steane [19, 95] (CSS) code, and as for any CSS code with a single encoded qubit, an encoded CNOT can be performed *transversally*—in other words, if simultaneous CNOT's are executed from each qubit in one block to the corresponding qubit in the other block, the effect is to execute the encoded CNOT [50]. To see this, we first need to verify that the transversal CNOT preserves the code space, *i.e.*, that its action by conjugation preserves the code's stabilizer. This follows immediately from eq. (4.109), since each stabilizer generator is either a tensor product of X's or a tensor product of Z's. Next we need to check that $\mathrm{CNOT}^{\otimes n}$ acts on the encoded operations $\bar{X}$ and $\bar{Z}$ as in eq. (4.109), which also follows immediately since $\bar{Z}$ is a tensor product of Z's and $\bar{X}$ is a tensor product of X's.

Hadamard gate

What about the Hadamard gate? In fact, applying the bitwise operation $H^{\otimes n}$ does not preserve the code space; rather it maps the code space of one planar code to that of another, different, planar code. If the stabilizer generators of the initial code are site operators X_s and plaquette operators Z_P, then the action of

the bitwise Hadamard is

$$H^{\otimes n} : X_s \to Z_s, \quad Z_P \to X_P \tag{4.111}$$

Compared to the initial code, the stabilizer of the new code has sites and plaquettes interchanged. We may reinterpret the new code as a code with X_s and Z_P check operators, but defined on a lattice dual to the lattice of the original code. If the original lattice has its "rough" edges at the north and south, then the new lattice has its rough edges at the east and west. We will refer to the two codes as the "north-south" (NS) code and the "east-west" (EW) code. As indicated in Fig. 4.20, the action of $H^{\otimes n}$ on the encoded operations $\bar{X}$ and $\bar{Z}$ of the NS code is

$$H^{\otimes n} : \bar{X}_{\mathrm{NS}} \to \bar{Z}_{\mathrm{EW}}, \quad \bar{Z}_{\mathrm{NS}} \to \bar{X}_{\mathrm{EW}}. \tag{4.112}$$

If we rigidly rotate the lattice by 90°, the EW code is transformed back to the NS code. Hence, the overall effect of a bitwise Hadamard and a 90° rotation is an encoded Hadamard $\bar{H}$.

Of course, a physical rotation of the lattice might be inconvenient in practice! Instead, we will suppose that "peripheral" qubits are available at the edge of the code block, and that we have the option of incorporating these qubits into the block or ejecting them from the block using the method described in Sec. 4.8.3. After applying the bitwise Hadamard, transforming the $L \times L$ NS code to the EW code, we add $L-1$ plaquettes to the northern edge and $L-1$ sites to the western edge, while removing $L-1$ plaquettes on the east and $L-1$ sites on the south. This procedure transforms the block back to the NS code, but with the qubits shifted by half a lattice spacing to the north and west—we'll call this shifted code the NS$'$ code. Furthermore, this modification of the boundary transforms the logical operations $\bar{Z}_{\mathrm{EW}}$ and $\bar{X}_{\mathrm{EW}}$ of the EW code to the operations $\bar{Z}_{\mathrm{NS}'}$ and $\bar{X}_{\mathrm{NS}'}$ of the NS$'$ code. The overall effect, then, of the bitwise Hadamard followed by the

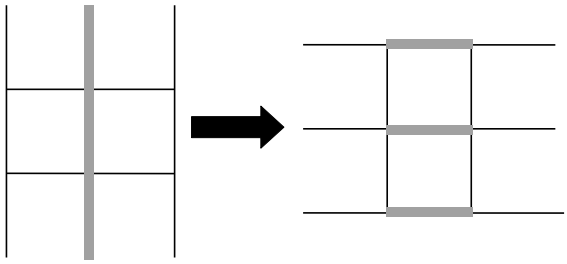

Figure 4.20: Action of the bitwise Hadamard gate on the planar code. If Hadamard gates are applied simultaneously to all the qubits in the block, an "NS code" with rough edges at the north and south is transformed to an "EW code" with rough edges at the east and west; the encoded operation $\bar{Z}_{\rm NS}$ of the NS code is transformed to $\bar{X}_{\rm EW}$ of the EW code, and $\bar{X}_{\rm NS}$ is transformed to $\bar{Z}_{\rm EW}$.

boundary modification is the operation

$$\bar{X}_{\rm NS} \to \bar{Z}_{\rm NS'}, \quad \bar{Z}_{\rm NS} \to \bar{X}_{\rm NS'}. \tag{4.113}$$

In principle, we could complete the encoded Hadamard gate by physically shifting the qubits half a lattice spacing to the south and east, transforming the NS$'$ code back to the NS code. One way to execute this shift might be to swap the qubits of the NS$'$ with qubits located at the corresponding sites of the NS lattice. If we prefer to avoid the additional quantum processing required by the swaps, then what we can do instead is associate a classical flag bit with each code block, recording whether the number of Hadamard gates that have been applied in our circuit to that logical qubit is even or odd, and hence whether the logical qubit is encoded in the NS code or the NS$'$ code. This classical bit is consulted whenever the circuit calls for a Hadamard or CNOT acting on the block. If we perform a Hadamard on a qubit that is initially encoded with the NS$'$ code, we add qubits on the south

and east while removing them from the north and west, returning to the NS code. The CNOT gates are performed transversally between blocks that are both in the NS code or both in the NS$'$ code; that is, each qubit in one layer interacts with the corresponding qubit directly below it in the next layer. But if one block is in the NS code and the other is in the NS$'$ code, then each qubit in one layer interacts with the qubit in the next layer that is half a lattice spacing to north and west. Note that the modification of the boundary requires a number of computation steps that is linear in L.

Phase gate

For implementation of the phase gate P, note that if we can execute CNOT and H then we can also construct the "controlled-(iY)" gate

$$\Lambda(iY) = \Lambda(ZX) = (IH) \cdot \Lambda(X) \cdot (IH) \cdot \Lambda(X). \tag{4.114}$$

Hence it suffices to be able to prepare an eigenstate $|+\rangle$ or $|-\rangle$ of Y,

$$Y|\pm\rangle = \pm|\pm\rangle; \tag{4.115}$$

if we prepare an ancilla in the state $|+\rangle$, and apply a CNOT with the data as its control and the ancilla as its target, the effect on the data is the same as $\Lambda(i) = P$. If the ancilla is the state $|-\rangle$, then we apply $\Lambda(-i) = P^{-1}$ to the data instead.

Now, it is not obvious how to prepare a large toric block in an eigenstate of the encoded Y with good fidelity. Fortunately, we can nevertheless use a CNOT and an ancilla to implement P, thanks to a trick that works because P is the only gate in our set that is not real. Consider a circuit that applies the unitary transformation U to the data if the ancilla has actually been prepared in the state $|+\rangle$. Then if $|+\rangle$ were replaced by $|-\rangle$, this same circuit would apply the complex conjugate unitary U^*, since each P in the circuit would be replaced by P^*.

Instead of a Y eigenstate, suppose we prepare the ancilla in any encoded state we please, for example, $|\bar{0}\rangle$. And then we use this same ancilla block, and a CNOT,

every time a P is to be executed. The state of the ancilla can be expressed as a linear combination $a|+\rangle + b|-\rangle$ of the Y eigenstates, and our circuit, acting on the initial state $|\psi\rangle$ of the data, yields

$$a|+\rangle \otimes U|\psi\rangle + b|-\rangle \otimes U^*|\psi\rangle. \tag{4.116}$$

Now, at the very end of a quantum computation, we will need to make a measurement to read out the final result. Let A denote the observable that we measure. The expectation value of A will be

$$\langle A\rangle = |a|^2\langle\psi|U^\dagger A U|\psi\rangle + |b|^2\langle\psi|U^\dagger A^T U|\psi\rangle, \tag{4.117}$$

where A^T denotes the transpose of A. Without losing any computational power, we may assume that the observable A is real ($A = A^T$)—for example, it could be $\frac{1}{2}(I-Z)$ acting on one of our encoded blocks. Then we get the same answer for the expectation value of A as if the ancilla had been prepared as $|+\rangle$ (or $|-\rangle$); hence our fault-tolerant procedure successfully simulates the desired quantum circuit.

Since there is just one ancilla block that must be used each time the P gate is executed, this block has to be swapped into the position where it is needed, a slowdown that is linear in the width of the quantum circuit that is being simulated.

Thus we have described a way to perform fault-tolerant normalizer computation for planar surface codes. We envision, then, a quantum computer consisting of a stack of planar sheets, with a logical qubit residing in each sheet. Each logical sheet has associated with it an adjacent sheet of ancilla qubits that are used to measure the check operators of the surface code; after each measurement, these ancilla qubits are refreshed in place and then reused. The quantum information in one sheet can be swapped with that in the neighboring sheet through the action of local gates. To perform a logical CNOT between two different logical qubits in the stack, we first use swap gates to pass the qubits through the intervening sheets of logical and ancilla qubits and bring them into contact, then execute the transversal CNOT between the two layers, and then use swap gates to return the

logical qubits to their original positions. By inserting a round of error correction after each swap or logical operation, we can execute a normalizer circuit reliably.

4.9.2 State purification and universal quantum computation

Now we need to consider how to complete our universal gate set by adding the Toffoli gate. As Shor observed [93], implementation of the gate can be reduced to the problem of preparing a particular three-qubit state, which may be chosen to be

$$|\psi\rangle_{\rm anc} = 2^{-3/2} \sum_{a,b,c\in\{0,1\}} (-1)^{abc} |a\rangle_1 |b\rangle_2 |c\rangle_3; \tag{4.118}$$

this state is the simultaneous eigenstate of three commuting symplectic operators: $\Lambda(Z)_{1,2}X_3$ and its two cyclic permutations, where $\Lambda(Z)$ is the two-qubit conditional phase gate

$$\Lambda(Z) : |a,b\rangle \to (-1)^{ab} |a,b\rangle. \tag{4.119}$$

Shor's method for constructing this state involved the preparation and measurement of an unprotected n-qubit cat state, where n is the block size of the code. But this method cannot be used for a toric code on a large lattice, because the cat state is too highly vulnerable to error.

Fortunately, there is an alternative procedure for constructing the needed encoded state with high fidelity—*state purification*. Suppose that we have a supply of noisy copies of the state $|\psi\rangle_{\rm anc}$. We can carry out a purification protocol to distill from our initial supply of noisy states a smaller number of states with much better fidelity [58, 28]. In this protocol, normalizer gates are applied to a pair of noisy copies, and then one member of the pair is measured. Based on the outcome of the measurement, the other state is either kept or discarded. If the initial ensemble of states approximates the $|\psi\rangle_{\rm anc}$ with adequate fidelity, then as purification proceeds, the fidelity of the remaining ensemble converges rapidly toward one.

For this procedure to work, it is important that our initial states are not *too* noisy—there is a purification threshold. Therefore, to apply the purification method to toric codes, we will need to build up the size of the toric block gradually,

as in the procedure for encoding unknown states described in Sec. 4.8.3. We start out by encoding $|\psi\rangle_{\rm anc}$ on a small planar sheet of qubits, with a fidelity below the purification threshold. Then we purify for a while to improve the fidelity, and build on the lattice to increase the size of the code block. By building and purifying as many times as necessary, we can construct a copy of the ancilla state that can be used to execute the Toffoli gate with high fidelity.

The time needed to build up the encoded blocks is quadratic in L, and the number of rounds of purification needed is linear in L, if we wish to reach a fidelity that is exponentially small in L. Thus the overhead incurred in our implementation of the Toffoli gate is polynomial in the block size.

We have now assembled all the elements of a fault-tolerant universal quantum computer based on planar surface codes. The computer is a stack of logical qubits, and it contains "software factories" where the ancilla states needed for execution of the Toffoli gate are prepared. Once prepared, these states can be transported through swapping to the position in the stack where the Toffoli gate is to be performed.

4.10 A local algorithm in four dimensions

In our recovery procedure, we have distinguished between quantum and classical computation. Measurements are performed to collect syndrome information about errors that have accumulated in the code block, and then a fast and reliable classical computer processes the measured data to infer what recovery step is likely to remove most of the errors. These procedures are fault tolerant because the quantum computation needed to measure the syndrome is highly local. But the classical computation not so local—the algorithm for constructing the chain of minimal weight requires as input the syndrome history of the entire code block.

It would be preferable to replace this procedure by one in which measurements and classical processing are eliminated, and all of the processing is local quantum processing. Can we devise a stable quantum memory based on topological coding

such that rapid measurements of the syndrome are not necessary?

Heuristically, errors create pairs of defects in the code block, and trouble may arise if these defects diffuse apart and annihilate other defects, eventually generating homologically nontrivial defect world lines. In principle, we could protect the encoded quantum information effectively if there is a strong attractive interaction between defects that prevents them from wandering apart. A recovery procedure that simulates such interactions was discussed in Ref. [28]. For that procedure, an accuracy threshold can be established, but only if the interactions have arbitrarily long range, in which case the order-disorder transition in the code block is analogous to the Kosterlitz-Thouless transition in a two-dimensional Coulomb gas. But to simulate these infinite-range interactions, nonlocal processing is still required.

A similar problem confronts the proposal [61, 83, 41] to encode quantum information in a configuration of widely separated nonabelian anyons. Errors create anyons in pairs, and the encoded information is endangered if these "thermal anyons" diffuse among the anyons that encode the protected quantum state. In principle, a long-range attractive interaction among anyons might control the diffusion, but this interaction might also interfere with the exchanges of anyons needed to process the encoded state. In any case, a simulation of the long-range dynamics involves nonlocal processing.

I will now describe a procedure for recovery that, at least mathematically, requires no such nonlocal processing of quantum or classical information. With this procedure, based on "locally available" quantum information, we can infer a recovery step that is more likely to remove errors than add new ones. Because the procedure is local we can dispense with measurement without degrading its performance very much—measurements followed by quantum gates conditioned on measurement outcomes can be replaced my unitary transformations acting on the data qubits and on nearby ancilla qubits. But since we will still need a reservoir where we can dispose the entropy introduced by random errors, we will continue to assume as usual that the ancilla qubits can be regularly refreshed as needed.

Unfortunately, while our procedure is local in the mathematical sense that

recovery operations are conditioned on the state of a small number of "nearby" qubits, we do not know how to make it *physically* local in a space of fewer than four dimensions.

4.10.1 Repetition code in two dimensions

The principle underlying our local recovery procedure can be understood if we first consider the simpler case of a repetition code. We can imagine that the code block is a periodically identified one-dimensional lattice of binary spins, with two codewords corresponding to the configurations with all spins up or all spins down. To diagnose errors, we can perform a local syndrome measurement by detecting whether each pair of neighboring spins is aligned or anti-aligned, thus finding the locations of defects where the spin orientation flips.

To recover, we need to bring these defects together in pairs to annihilate. One way to do this is to track the history of the defects for a while, assembling a record S of the measured syndrome, and then find a minimum-weight chain E' with the same boundary, in order to reconstruct hypothetical world lines of the defects. But in that case the processing required to construct E' is nonlocal.

The way to attain a local recovery procedure is to increase the dimensionality of the lattice. In two dimensions, errors will generate droplets of flipped spins (as in Fig. 4.21), and the local syndrome measurement will detect the boundary of the droplet. Thus the defects now form one-dimensional closed loops, and our recovery step should be designed to reduce the total length of such defects. Local dynamical rules can easily be devised that are more likely to shrink a loop than stretch it, just as it is possible to endow strings with local dynamics (tension and dissipation) that allow the strings to relax. Thus in equilibrium, very long loops will be quite rare. If the error rate is small enough, then the droplets of flipped spins will typically remain small, and the encoded information will be well protected.

That the two-dimensional version of the repetition code is more robust than the one-dimensional version illustrates a central principle of statistical mechanics—that order is more resistant to fluctuations in higher dimensions. The code block

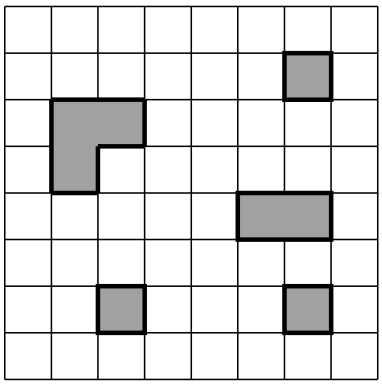

Figure 4.21: Droplets of flipped qubits in the two-dimensional quantum repetition code. Qubits reside on plaquettes, and the qubits that have been flipped are shaded. Thick links are locations of "defects" where the error syndrome is nontrivial because neighboring qubits are anti-aligned. The defects form closed loops that enclose the droplets.

is described by an Ising spin model, and while the one-dimensional Ising model is disordered at any nonzero temperature, the two-dimensional Ising model remains ordered up to a nonvanishing critical temperature. From the perspective of coding theory, the advantage of the two-dimensional version is that the syndrome is highly redundant. If we check each pair of nearest-neighbor spins to see if they are aligned or anti-aligned, we are collecting more information than is really needed to diagnose all the errors in the block. Hence there is a constraint that must be satisfied by a valid syndrome, namely that the boundary of a droplet can never end; therefore, errors in the syndrome can be detected. Of course, physically, the stability of the ordered state of the Ising model in more than one dimension is the reason that magnetic memories are robust in Nature.

4.10.2 Toric code in four dimensions

The defects detected by the measurement of the stabilizer operators of a two-dimensional toric code are also pointlike objects, and error recovery is achieved by

bringing the defects together to annihilate. We can promote the annihilation by introducing an effective long-range interaction between defects, but a more local alternative procedure is to increase the dimensionality of the lattice.

So consider a *four-dimensional* toric code. Qubits are associated with each plaquette. With each link is associated the six-qubit stabilizer operator $X_\ell = X^{\otimes 6}$ acting on the six plaquettes that contain the link, and with each cube is associated the six-qubit stabilizer operator $Z_C = Z^{\otimes 6}$ acting on the six plaquettes contained in the cube. Thus the four-dimensional code maintains the duality between phase and flip errors that we saw in two dimensions. The encoded $\bar{Z}$ or $\bar{X}$ operation is constructed from Z's or X's acting on a homologically nontrivial surface of the lattice or dual lattice respectively. Z errors on a connected open surface generate a closed loop of defects on the boundary of the surface, and X errors on a connected open surface of the dual lattice generate defects on a set of cubes that form a closed loop on the dual lattice. As in the two-dimensional case, there is a "hyperplanar" version of the code that can be defined on a four-dimensional region with a boundary.

Now we want to devise a recovery procedure that will encourage the defect loops to shrink and disappear. Assuming that syndrome measurements are employed, a possible procedure for controlling phase errors can be described as follows: First, the stabilizer operator X_ℓ is measured at each link, and a record is stored of the outcome. We say that each link with $X_\ell = -1$ is occupied by a string, and each link with $X_\ell = 1$ is unoccupied. We choose a set of nonoverlapping plaquettes (with no link shared by two plaquettes in the set), and based on the syndrome for the links of that plaquette, decide whether or not to flip the plaquette (by applying a Z). If three or four of the plaquette's links are occupied by string, we always flip the plaquette. If zero or one link is occupied, we never flip it. And if two links are occupied, we flip the plaquette with probability 1/2. Then in the next time step, we again measure the syndrome, and decide whether to flip another nonoverlapping set of plaquettes. And so on.

Naturally, we also measure the bit-flip syndrome—Z_C on every cube—in each

time step. The procedure for correcting the bit-flip errors is identical, with the lattice replaced by the dual lattice, and X replaced by Z.

Of course the measurement is not essential. A simple reversible computation can imprint the number of string bits bounding a plaquette on ancilla qubits, and subsequent unitary gates controlled by the ancilla can "decide" whether to flip the plaquette. Note that a CNOT that is applied with probability 1/2, needed in the event that the plaquette has two string bits on its boundary, can be realized by a Toffoli gate, where one of the control qubits is a member of a Bell pair so that the control takes the value 1 with probability 1/2.

This recovery procedure has the property that, if it is perfectly executed and no further errors occur during its execution, it will never increase the total length of string on the lattice, but it will sometimes reduce the length. Indeed, if it is applied repeatedly while no further errors occur, it will eventually eliminate every string. We have chosen to make the procedure nondeterministic in the case where there are two string bits on a plaquette, because otherwise the procedure would have closed orbits—some string configurations would oscillate indefinitely rather than continuing to shrink and annihilate. With the nondeterministic procedure, a steady state can be attained only when all the strings have disappeared.

Actually, following the ideas of Toom [98], it is possible to devise *anisotropic deterministic* procedures that also are guaranteed to remove all strings. These procedures, in fact, remove the strings more efficiently than our nondeterministic one, but are a little more difficult to analyze.

Of course, the recovery procedure will not really be executed flawlessly, and further errors will continue to accumulate. Still, as error recovery is performed many times, an equilibrium will eventually be attained in which string length is being removed by recovery as often as it is being created by new errors. If the error rates are small enough, the equilibrium population of long string loops will be highly suppressed, so that the encoded quantum information will be well protected.

Eventually, say at the conclusion of a computation, we will want to measure encoded qubits. This measurement procedure does have a nonlocal component (as

the encoded information is topological), and for this purpose only we will assume that a reliable classical computer is available to help with the interpretation of the measured data. To measure the logical operator $\bar{Z}$, say, we first measure every qubit in the code block. Then we apply a classical parity check, evaluating Z_C for each cube of the lattice, thereby generating a configuration of closed defect loops on the dual lattice. To complete the measurement, we first eliminate the defects by applying flips to a set of plaquettes bounded by each loop. Then we can evaluate the product of Z's associated with a homologically nontrivial surface to find the value of $\bar{Z}$.

Of course, when we eliminate the defects, we need to make sure that we choose correctly among the homologically inequivalent surfaces bounded by the observed strings. One way to do so, which is unlikely to fail when qubit and measurement error probabilities are small, is to invoke the relaxation algorithm formulated above to the classical measurement outcome. Since our classical computer is reliable, the algorithm eventually removes all strings, and then the value of $\bar{Z}$ can be determined.

4.10.3 Accuracy threshold

To evaluate the efficacy of the local recovery method, we need to find the equilibrium distribution of defects. This equilibrium configuration is not so easily characterized, but it will suffice to analyze a less effective algorithm that does attain a simple steady state—the heat bath algorithm. To formulate the heat bath algorithm, suppose that strings carry an energy per lattice unit length that we may normalize to one, and suppose that each plaquette is in contact with a thermal reservoir at inverse temperature β. In each time step, plaquettes are updated, with the change in the string length bounding a plaquette governed by the Boltzmann probability distribution. Thus survival or creation of a length-4 loop is suppressed by the factor

$$\frac{\text{Prob}(0 \to 4)}{\text{Prob}(0 \to 0)} = \frac{\text{Prob}(4 \to 4)}{\text{Prob}(4 \to 0)} = e^{-4\beta}. \tag{4.120}$$

Similarly, the probability of a plaquette flip when the length of bounding string is 3 or 1 satisfies

$$\frac{\text{Prob}(1 \to 3)}{\text{Prob}(1 \to 1)} = \frac{\text{Prob}(3 \to 3)}{\text{Prob}(3 \to 1)} = e^{-2\beta}. \tag{4.121}$$

In the case of a plaquette with two occupied links, we again perform the flip with probability 1/2. As before, this ensures ergodicity—any initial configuration has some nonvanishing probability of reaching any final configuration.

Damage to encoded information arises from string "world sheets" that are homologically nontrivial. At low temperature, string loops are dilute and failure is unlikely, but at a critical temperature the strings "condense," and the encoded data are no longer well protected. The critical temperature is determined by a balance between Boltzmann factor $e^{-\beta\ell}$ suppressing a string of length ℓ and the string entropy. The abundance of self-avoiding closed loops of length ℓ behaves like [74]

$$n_{\text{SAW}}^{(4)}(\ell) \sim P_4(\ell)(\mu_4)^\ell, \quad \mu_4 \approx 6.77, \tag{4.122}$$

in $d = 4$ dimensions, where $P_4(\ell)$ is a polynomial. Thus, large loops are rare when the sum

$$\sum_\ell n_{\text{SAW}}^{(4)}(\ell) e^{-\beta\ell} \sim \sum_\ell P_4(\ell) \left(\mu_4 e^{-\beta}\right)^\ell \tag{4.123}$$

converges, and the system is surely ordered for $e^{-\beta} < \mu_4^{-1}$. Thus the critical inverse temperature β_c satisfies

$$e^{-\beta_c} \geq (\mu_4)^{-1}. \tag{4.124}$$

Now, our local recovery procedure will not be precisely a heat bath algorithm. But like the heat bath algorithm it is more likely to destroy string than create it, and we can bound its performance by assigning to it an effective temperature. For example, if no new errors arise and the algorithm is perfectly executed, it will with probability one remove a length-4 string loop bounding a plaquette. In practice, though, the plaquette may not flip when the recovery computation is performed, either because of a fault during its execution, or because other neighboring pla-

quettes have flipped in the meantime. Let us denote by q_4 the probability that a plaquette, occupied by four string bits at the end of the last recovery step, does not in fact flip during the current step. Similarly, let q_3 denote the probability that a plaquette with three string bits fails to flip, and let q_1, q_0 denote the probabilities that plaquettes containing one or zero string bits *do* flip. These quantities can all be calculated, given the quantum circuit for recovery and a stochastic error model.

Now we can find a positive quantity q such that

$$\begin{aligned} q_0, q_4 &\leq q/(1+q), \\ q_1, q_3 &\leq \sqrt{q}/(1+\sqrt{q}). \end{aligned} \tag{4.125}$$

Comparing to eqs. (4.120,4.121), we see that our recovery algorithm is at least as effective as a heat bath algorithm with the equivalent temperature

$$e^{-4\beta} = q; \tag{4.126}$$

in equilibrium strings of length ℓ are therefore suppressed by a factor no larger than $e^{-\beta\ell} = q^{\ell/4}$. From our estimate of the critical temperature eq. (4.124), we then obtain a lower bound on the critical value of q:

$$q_c \geq (\mu_4)^{-4} \approx 4.8 \times 10^{-4}. \tag{4.127}$$

This quantum system with local interactions has an accuracy threshold.

A local procedure that controls the errors in a quantum memory is welcome, but it is disheartening that four spatial dimensions are required. Of course, the four-dimensional code block can be projected to $d < 4$ dimensions, but then interactions among four-dimensional neighbors become interactions between qubits that are distance $L^{(4-d)/d}$ apart, where L is the linear size of the lattice. In a three-dimensional version of the toric code, we can place qubits on plaquettes, and associate check operators with links and cubes. Thus, phase error defects are strings and bit-flip error defects are point particles, or vice versa. Then we can

recover locally (without measurement or classical computation) from either the phase errors or the bit-flip errors, but not both.

In fewer than four spatial dimensions, how might we devise an intrinsically stable quantum memory, analogous to a magnetic domain with long-range order that encodes a robust classical bit? Perhaps we can build a two-dimensional material with a topologically degenerate ground state, such that errors create point defects that have infinite-range attractive interactions. That system's quasi-long-range order at nonzero temperature could stabilize an arbitrary coherent superposition of ground states.

4.11 Conclusions

In foreseeable quantum computers, the quantum gates that can be executed with good fidelity are likely to be *local* gates—only interactions between qubits that are close to one another will be accurately controllable. Therefore, it is important to contemplate the capabilities of large-scale quantum computers in which all gates are local in three-dimensional space. It is also reasonable to imagine that future quantum computers will include some kind of integrated classical processors, and that the classical processors will be much more accurate and much faster than the quantum processors.

Such considerations have led to this chapter's investigation of the efficacy of quantum error correction in a computational model in which all quantum gates are local, classical computations of polynomial size can be done instantaneously and with perfect accuracy, and measurement of a qubit can be done as quickly as the execution of a quantum gate.

These conditions are ideally suited for the use of topological quantum error-correcting codes, such that all quantum computations needed to extract an error syndrome have excellent locality properties. Indeed, I have shown that if the two-dimensional surface codes introduced in [60, 61] are used, then an accuracy threshold for quantum storage can be established, and its numerical value can

be estimated. This accuracy threshold can be interpreted as a critical point of a three-dimensional lattice gauge theory with quenched randomness, where the third dimension represents time. There is also an accuracy threshold for universal quantum computation, but it has not been calculated carefully.

Topological codes provide a compelling framework for controlling errors in a quantum system via local quantum processing; for this reason, these codes should figure prominently in the future evolution of quantum technologies. In any case, the analysis in this chapter amply illustrates that principles from statistical physics and topology can be fruitfully applied to the daunting task of accurately manipulating intricate quantum states.

Chapter 5

Quantum measurement algorithms

Abstract

In this chapter, I will describe a new class of quantum algorithms for solving combinatorial search problems. These algorithms, called *quantum measurement algorithms*, use only a sequence of measurements to achieve their goal. Quantum measurement algorithms are similar in spirit to quantum adiabatic algorithms, in that both are designed to keep quantum information in an eigenstate of a time-varying operator. Indeed, I will show that one may view a quantum measurement algorithm as a polynomial simulation of a quantum adiabatic algorithm. I will also show how to achieve a quadratic speedup for Grover's unstructured search problem with a quantum search algorithm that uses only two measurements.

The work presented in this chapter is the result of a collaboration with Childs, Deotto, Farhi, Gutmann, and Goldstone [22].

5.1 Introduction

In the conventional circuit model of quantum computation, a program for a quantum computer consists of a discrete sequence of unitary gates chosen from a fixed

set. The memory of the quantum computer is a collection of qubits initially prepared in some definite state. After a sequence of unitary gates is applied, the qubits are measured in the computational basis to give the result of the computation, a classical bit string.

This description of a quantum computer has been used to formulate quantum algorithms that outperform classical methods, notably Shor's factoring algorithm [91] and Grover's algorithm for unstructured search [53]. Subsequent development of quantum algorithms has focused primarily on variations of the techniques introduced by Shor and Grover. One way to motivate new algorithmic ideas is to consider alternative (but in general, equivalent) descriptions of the way a quantum computer operates. For example, the technique of quantum computation by adiabatic evolution [38] is most easily described by a quantum computer that evolves continuously according to a time-varying Hamiltonian.

Another model of quantum computation allows measurement at intermediate stages. Indeed, recent work has shown that *measurement alone* is universal for quantum computation: one can efficiently implement a universal set of quantum gates using only measurements (and classical processing) [79, 40, 70, 88]. In this chapter, we describe an algorithm for solving combinatorial search problems that consists only of a sequence of measurements. Using a straightforward variant of the quantum Zeno effect (see, for example, [4, 90]), we show how to keep the quantum computer in the ground state of a smoothly varying Hamiltonian $H(s)$. This process can be used to solve a computational problem by encoding the solution to the problem in the ground state of the final Hamiltonian.

The organization of the chapter is as follows. In Section 5.2, we present the algorithm in detail and describe how measurement of $H(s)$ can be performed on a digital quantum computer. In Section 5.3, we estimate the running time of the algorithm in terms of spectral properties of $H(s)$. Then, in Section 5.4, we discuss how the algorithm performs on Grover's unstructured search problem and show that by a suitable modification, Grover's quadratic speedup can be achieved by the measurement algorithm. Finally, in Section 5.5, we discuss the relation-

ship between the measurement algorithm and quantum computation by adiabatic evolution.

5.2 The measurement algorithm

5.2.1 Adiabatic algorithms by the Zeno effect

Our algorithm is conceptually similar to quantum computation by adiabatic evolution [38], a general method for solving combinatorial search problems using a quantum computer. Both algorithms operate by remaining in the ground state of a smoothly varying Hamiltonian $H(s)$ whose initial ground state is easy to construct and whose final ground state encodes the solution to the problem. However, whereas adiabatic quantum computation uses Schrödinger evolution under $H(s)$ to remain in the ground state, the present algorithm uses *only* measurement of $H(s)$.

In general, we are interested in searching for the minimum of a function $h(z)$ that maps n-bit strings to positive real numbers. Many computational problems can be cast as minimization of such a function; for specific examples and their relationship to adiabatic quantum computation, see [38, 23]. Typically, we can restrict our attention to the case where the global minimum of $h(z)$ is unique. Associated with this function, we can define a *problem Hamiltonian* H_P through its action on computational basis states:

$$H_P|z\rangle = h(z)|z\rangle\,. \tag{5.1}$$

Finding the global minimum of $h(z)$ is equivalent to finding the ground state of H_P. If the global minimum is unique, then this ground state is nondegenerate.

To reach the ground state of H_P, we begin with the quantum computer prepared in the ground state of some other Hamiltonian H_B, the *beginning Hamiltonian*. Then we consider a one-parameter family of Hamiltonians $H(s)$ that interpolates smoothly from H_B to H_P for $s \in [0,1]$. In other words, $H(0) = H_B$ and

$H(1) = H_P$, and the intermediate $H(s)$ is a smooth function of s. One possible choice is linear interpolation,

$$H(s) = (1-s)H_B + sH_P\,. \tag{5.2}$$

Now we divide the interval $[0,1]$ into M subintervals of width $\delta = 1/M$. So long as the interpolating Hamiltonian $H(s)$ is smoothly varying and δ is small, the ground state of $H(s)$ will be close to the ground state of $H(s+\delta)$. Thus, if the system is in the ground state of $H(s)$ and we measure $H(s+\delta)$, the post-measurement state is very likely to be the ground state of $H(s+\delta)$. If we begin in the ground state of $H(0)$ and successively measure $H(\delta), H(2\delta), \ldots, H((M-1)\delta), H(1)$, then the final state will be the ground state of $H(1)$ with high probability, assuming δ is sufficiently small.

5.2.2 The system-meter model

To complete our description of the quantum algorithm, we must explain how to measure the operator $H(s)$. The technique we use is motivated by von Neumann's description of the measurement process [103]. In this description, measurement is performed by coupling the system of interest to an ancillary system, which we call the *pointer*. Suppose that the pointer is a one-dimensional free particle and that the system-pointer interaction Hamiltonian is $H(s) \otimes p$, where p is the momentum of the particle. Furthermore, suppose that the mass of the particle is sufficiently large that we can neglect the kinetic term. Then the resulting evolution is

$$e^{-itH(s)\otimes p} = \sum_a \left[|E_a(s)\rangle\langle E_a(s)| \otimes e^{-itE_a(s)p} \right], \tag{5.3}$$

where $|E_a(s)\rangle$ are the eigenstates of $H(s)$ with eigenvalues $E_a(s)$, and we have set $\hbar = 1$. Suppose we prepare the pointer in the state $|x=0\rangle$, a narrow wave packet centered at $x = 0$. Since the momentum operator generates translations in

position, the above evolution performs the transformation

$$|E_a(s)\rangle \otimes |x = 0\rangle \rightarrow |E_a(s)\rangle \otimes |x = tE_a(s)\rangle. \tag{5.4}$$

If we can measure the position of the pointer with sufficiently high precision that all relevant spacings $x_{ab} = t|E_a(s) - E_b(s)|$ can be resolved, then measurement of the position of the pointer—a fixed, easy-to-measure observable, independent of $H(s)$—effects a measurement of $H(s)$.

5.2.3 Digitizing the algorithm

Von Neumann's measurement protocol makes use of a continuous variable, the position of the pointer. To turn it into an algorithm that can be implemented on a fully digital quantum computer, we can approximate the evolution (5.3) using r quantum bits to represent the pointer [109, 114]. The full Hilbert space is thus a tensor product of a 2^n-dimensional space for the system and a 2^r-dimensional space for the pointer. We let the computational basis of the pointer, with basis states $\{|z\rangle\}$, represent the basis of momentum eigenstates. The label z is an integer between 0 and $2^r - 1$, and the r bits of the binary representation of z specify the states of the r qubits. In this basis, the digital representation of p is

$$p = \sum_{j=1}^{r} 2^{-j} \frac{1 - \sigma_z^{(j)}}{2}\,, \tag{5.5}$$

a sum of diagonal operators, each of which acts on only a single qubit. Here $\sigma_z^{(j)}$ is the Pauli z operator on the jth qubit. As we will discuss in the next section, we have chosen to normalize p so that

$$p|z\rangle = \frac{z}{2^r}|z\rangle\,, \tag{5.6}$$

which gives $\|p\| \sim 1$. If $H(s)$ is a sum of terms, each of which acts on at most k qubits, then $H(s) \otimes p$ is a sum of terms, each of which acts on at most $k + 1$

qubits. As long as k is a fixed constant independent of the problem size n, such a Hamiltonian can be simulated efficiently on a quantum computer [71]. In the momentum eigenbasis, the initial state of the pointer is

$$|x = 0\rangle = \frac{1}{2^{r/2}} \sum_{z=0}^{2^r-1} |z\rangle \,. \tag{5.7}$$

The measurement is performed by evolving under $H(s) \otimes p$ for a total time τ. We discuss how to choose τ in the next section. After this evolution, the position of the simulated pointer could be measured by measuring the qubits that represent it in the x basis, i.e., the Fourier transform of the computational basis. However, note that our algorithm only makes use of the post-measurement state of the system, not of the measured value of $H(s)$. In other words, only the reduced density matrix of the system is relevant. Thus it is not actually necessary to perform a Fourier transform before measuring the pointer, or even to measure the pointer at all. When the system-pointer evolution is finished, one can immediately re-prepare the pointer in its initial state $|x = 0\rangle$ and begin the next measurement.

As an aside, note that the von Neumann measurement procedure described above is identical to the well-known phase estimation algorithm for measuring the eigenvalues of a unitary operator [59, 26], which can also be used to produce eigenvalues and eigenvectors of a Hamiltonian [1]. This connection has been noted previously in [114], and it has been pointed out that the measurement is a non-demolition measurement in [99]. In the phase estimation problem, we are given an eigenvector $|\psi\rangle$ of a unitary operator U and asked to determine its eigenvalue $e^{-i\phi}$. The algorithm uses two registers, one that initially stores $|\psi\rangle$ and one that will store an approximation of the phase ϕ. The first and last steps of the algorithm are Fourier transforms on the phase register. The intervening step is to perform the transformation

$$|\psi\rangle \otimes |z\rangle \to U^z |\psi\rangle \otimes |z\rangle \,, \tag{5.8}$$

where $|z\rangle$ is a computational basis state. If we take $|z\rangle$ to be a momentum eigenstate with eigenvalue z (i.e., if we choose a different normalization than in (5.6))

and let $U = e^{-iHt}$, this is exactly the transformation induced by $e^{-i(H\otimes p)t}$. Thus we see that the phase estimation algorithm for a unitary operator U is exactly von Neumann's prescription for measuring $i \ln U$.

5.3 Running time

The running time of the measurement algorithm is the product of M, the number of measurements, and τ, the time per measurement. Even if we assume perfect projective measurements, the algorithm is guaranteed to keep the computer in the ground state of $H(s)$ only in the limit $M \to \infty$, so that $\delta = 1/M \to 0$. Given a finite running time, the probability of finding the ground state of H_P with the last measurement will be less than 1. To understand the efficiency of the algorithm, we need to determine how long we must run as a function of n, the number of bits on which the function h is defined, so that the probability of success is not too small. In general, if the time required to achieve a success probability greater than some fixed constant (e.g., $\frac{1}{2}$) is poly(n), we say the algorithm is efficient, whereas if the running time grows exponentially, we say it is not.

To determine the running time of the algorithm, we consider the effect of the measurement process on the reduced density matrix of the system. Here, we simply motivate the main result; for a detailed analysis, see Section 5.6.

Let $\rho^{(j)}$ denote the reduced density matrix of the system after the jth measurement; its matrix elements are

$$\rho^{(j)}_{ab} = \langle E_a(j\delta)|\rho^{(j)}|E_b(j\delta)\rangle \,. \tag{5.9}$$

The interaction with the digitized pointer effects the transformation

$$|E_a(s)\rangle \otimes |z\rangle \to e^{-iE_a(s)zt/2^r}|E_a(s)\rangle \otimes |z\rangle \,. \tag{5.10}$$

Starting with the pointer in the state (5.7), evolving according to (5.10), and

tracing over the pointer, the quantum operation induced on the system is

$$\rho_{ab}^{(j+1)} = \kappa_{ab}^{(j)} \sum_{c,d} U_{ac}^{(j)} \rho_{cd}^{(j)} U_{bd}^{(j)*} , \tag{5.11}$$

where the unitary transformation relating the energy eigenbases at $s = j\delta$ and $s = (j+1)\delta$ is

$$U_{ab}^{(j)} = \langle E_b((j+1)\delta) | E_a(j\delta) \rangle \tag{5.12}$$

and

$$\kappa_{ab}^{(j)} = \frac{1}{2^r} \sum_{z=0}^{2^r-1} e^{i[E_b(j\delta) - E_a(j\delta)]zt/2^r} . \tag{5.13}$$

Summing this geometric series, we find

$$\left| \kappa_{ab}^{(j)} \right|^2 = \left| \kappa([E_b(j\delta) - E_a(j\delta)]t/2) \right|^2 , \tag{5.14}$$

where

$$|\kappa(x)|^2 = \frac{\sin^2 x}{4^r \sin^2(x/2^r)} . \tag{5.15}$$

This function is shown in Fig. 5.1 for the case $r = 4$. It has a sharp peak of unit height and width of order 1 at the origin, and identical peaks at integer multiples of $2^r\pi$.

If the above procedure were a perfect projective measurement, then we would have $\kappa_{ab} = 0$ whenever $E_a \neq E_b$. Assuming (temporarily) that this is the case, we find

$$\rho_{00}^{(j+1)} \geq \left| U_{00}^{(j)} \right|^2 \rho_{00}^{(j)} \tag{5.16}$$

with the initial condition $\rho_{00}^{(0)} = 1$ and $\rho_{ab}^{(0)} = 0$ otherwise. Perturbation theory gives

$$\begin{aligned} \left| U_{00}^{(j)} \right|^2 &= 1 - \delta^2 \sum_{a \neq 0} \left. \frac{|\langle E_a(s) | \frac{\mathrm{d}H}{\mathrm{d}s} | E_0(s) \rangle|^2}{(E_0(s) - E_a(s))^2} \right|_{s=j\delta} \\ &\quad + O(\delta^3) \qquad (5.17) \\ &\geq 1 - \frac{\Gamma(j\delta)^2 \, \delta^2}{g(j\delta)^2} + O(\delta^3) , \qquad (5.18) \end{aligned}$$

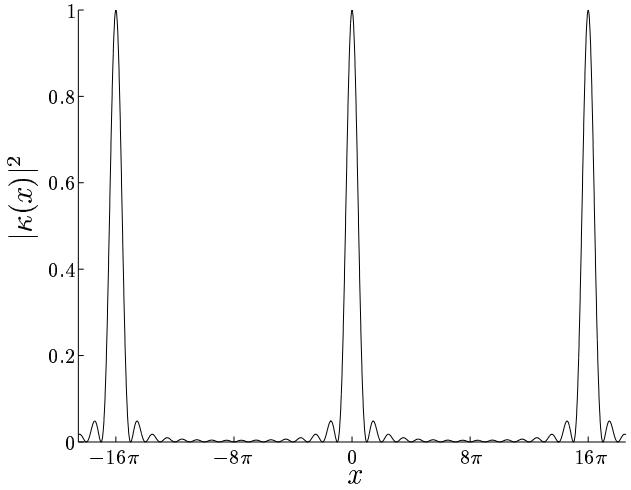

Figure 5.1: The function $|\kappa(x)|^2$ for $r = 4$.

where

$$\Gamma(s)^2 = \langle E_0(s)|(\tfrac{\mathrm{d}H}{\mathrm{d}s})^2|E_0(s)\rangle - \langle E_0(s)|\tfrac{\mathrm{d}H}{\mathrm{d}s}|E_0(s)\rangle^2 \tag{5.19}$$

and

$$g(s) = E_1(s) - E_0(s) \tag{5.20}$$

is the energy gap between the ground and first excited states. If we let

$$\Gamma = \max_{s\in[0,1]} \Gamma(s) \tag{5.21}$$

$$g = \min_{s\in[0,1]} g(s)\,, \tag{5.22}$$

then according to (5.16), the probability of being in the ground state after the last measurement is at least

$$\rho_{00}^{(M)} \geq \left[1 - \frac{\Gamma^2}{M^2 g^2} + O(M^{-3})\right]^M \tag{5.23}$$

$$= \exp\left(-\frac{\Gamma^2}{Mg^2}\right) + O(M^{-2})\,. \tag{5.24}$$

The probability of success is close to 1 provided

$$M \gg \frac{\Gamma^2}{g^2}\,. \tag{5.25}$$

When H_B and H_P are both sums of poly(n) terms, each of which acts nontrivially on at most a constant number of qubits, it is easy to choose an interpolation such as (5.2) so that Γ is only poly(n). Thus we are mainly interested in the behavior of g, the *minimum gap* between the ground and first excited states. We see that for the algorithm to be successful, the total number of measurements must be much larger than $1/g^2$.

However, the simulated von Neumann procedure is not a perfect projective measurement. Thus, we must determine how long the system and pointer should interact so that the measurement is sufficiently good. The analysis in Section 5.6 shows it is necessary that $|\kappa_{01}^{(j)}|^2$ be bounded below 1 by a constant for all j. In other words, to sufficiently resolve the difference between the ground and first excited states, we must decrease the coherence between them by a fixed fraction per measurement. The width of the central peak in Fig. 5.1 is of order 1, so it is straightforward to show that to have $|\kappa(x)|^2$ less than, say, $1/2$, we must have $x \geq O(1)$. This places a lower bound on the system-pointer interaction time of

$$\tau \geq \frac{O(1)}{g} \tag{5.26}$$

independent of r, the number of pointer qubits. (Note that the same bound also holds in the case of a continuous pointer with a fixed resolution length.)

Putting these results together, we find that the measurement algorithm is successful if the total running time, $T = M\tau$, satisfies

$$T \gg \frac{\Gamma^2}{g^3} \quad \text{(measurement)}\,. \tag{5.27}$$

This result can be compared to the corresponding expression for quantum compu-

tation by adiabatic evolution[1],

$$T \gg \frac{\Gamma}{g^2} \quad \text{(adiabatic)}\,. \tag{5.28}$$

Note that the same quantity appears in the numerator of both expressions; in both cases, Γ accounts for the possibility of transitions to all possible excited states.

The adiabatic and measurement algorithms have qualitatively similar behavior: if the gap is exponentially small, neither algorithm is efficient, whereas if the gap is only polynomially small, both algorithms are efficient. However, the measurement algorithm is slightly slower: whereas adiabatic evolution runs in a time that grows like $1/g^2$, the measurement algorithm runs in a time that grows like $1/g^3$. To see that this comparison is fair, recall that we have defined the momentum in (5.5) so that $||p|| \sim 1$, which gives $||H(s)|| \sim ||H(s) \otimes p||$. Alternatively, we can compare the number η of few-qubit unitary gates needed to simulate the two algorithms on a conventional quantum computer. Using the Lie product formula

$$e^{A+B} \simeq (e^{A/m} e^{B/m})^m\,, \tag{5.29}$$

which is valid provided $m \gg \|A\|^2 + \|B\|^2$, we find $\eta = O(1/g^4)$ for adiabatic evolution and $\eta = O(1/g^6)$ for the measurement algorithm, in agreement with the previous comparison.

The argument we have used to motivate (5.27) is explained in greater detail in Section 5.6. There, we also consider the number of qubits, r, that must be used to represent the pointer. We show that if the gap is only polynomially small in n, it is always sufficient to take $r = O(\log n)$. However, we argue that generally, a single qubit will suffice.

[1]The adiabatic bound is actually somewhat weaker than this in the numerator; the gap-dependence is what is significant for most situations.

5.4 The Grover problem

5.4.1 Oracle formulation

The unstructured search problem considered by Grover is to find a particular unknown n-bit string w (the marked state, or the *winner*) using only queries of the form "is z the same as w?" [53]. In other words, one is trying to minimize a function

$$h_w(z) = \begin{cases} 0 & z = w \\ 1 & z \neq w. \end{cases} \tag{5.30}$$

Since there are 2^n possible values for w, the best possible classical algorithm uses $\Theta(2^n)$ queries. However, Grover's algorithm requires only $\Theta(2^{n/2})$ queries, providing a (provably optimal [14]) quadratic speedup. In Grover's algorithm, the winner is specified by an *oracle* U_w with

$$U_w|z\rangle = (-1)^{h_w(z)}|z\rangle\,. \tag{5.31}$$

This oracle is treated as a black box that one can use during the computation. One call to this black box is considered to be a single query of the oracle.

In addition to Grover's original algorithm, quadratic speedup can also be achieved in a time-independent Hamiltonian formulation [39] or by adiabatic quantum computation [89, 100]. In either of these formulations, the winner is specified by an "oracle Hamiltonian"

$$H_w = 1 - |w\rangle\langle w|, \tag{5.32}$$

whose ground state is $|w\rangle$ and that treats all orthogonal states (the non-winners) equivalently. One is provided with a black box that implements H_w, where w is unknown, and is asked to find w. Instead of counting queries, the efficiency of the algorithm is quantified in terms of the total time for which one applies the oracle Hamiltonian.

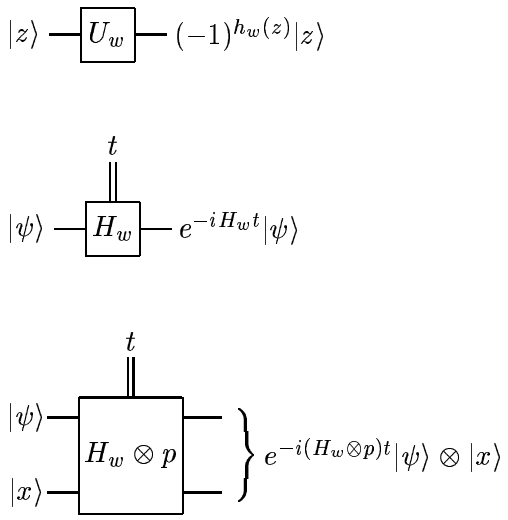

Figure 5.2: Oracles for the Grover problem. (a) Top: Grover's original oracle. (b) Center: An oracle that performs evolution according to H_w. The double line indicates a classical control parameter, the time for which the Hamiltonian is applied. (c) Bottom: An oracle that allows one to measure H_w.

Here, we show that if we are given a slightly different black box, we can achieve quadratic speedup using the measurement algorithm. We let the problem Hamiltonian be $H_P = H_w$ and we consider a one-parameter family of Hamiltonians $H(s)$ given by (5.2) for some H_B. Because we would like to *measure* this Hamiltonian, it is not sufficient to be given a black box that allows one to evolve the system according to H_w. Instead, we will use a black box that evolves the system and a pointer according to $H_w \otimes p$, where p is the momentum of the pointer. This oracle is compared to the previous two in Fig. 5.2. By repeatedly alternating between applying this black box and evolving according to $H_B \otimes p$, each for small time, we can produce an overall evolution according to the Hamiltonian $[sH_B + (1-s)H_P] \otimes p$, and thus measure the operator $H(s)$ for any s.

5.4.2 A two-measurement algorithm

Now consider the beginning Hamiltonian

$$H_B = \sum_j \frac{1 - \sigma_x^{(j)}}{2}\,, \tag{5.33}$$

where $\sigma_x^{(j)}$ is the Pauli x operator acting on the jth qubit. This beginning Hamiltonian is a sum of local terms, and has the easy-to-prepare ground state $|E_0(0)\rangle = 2^{-n/2}\sum_z |z\rangle$, the uniform superposition of all possible bit strings in the computational basis. If we consider the interpolation (5.2), then one can show [38] that the minimum gap occurs at

$$s^* = 1 - \frac{2}{n} + O(n^{-2})\,, \tag{5.34}$$

where the gap takes the value

$$g(s^*) = 2^{1-n/2}[1 + O(n^{-1})]\,. \tag{5.35}$$

Naively applying (5.27) gives a running time $T = O(2^{3n/2})$, which is even worse than the classical algorithm.

However, since we know the value of s^* independent of w, we can improve on this approach by making fewer measurements. We observe that in the limit of large n, the ground state of $H(s)$ is close to the ground state $|E_0(0)\rangle$ of H_B for $s \lesssim s^*$ and is close to the ground state $|E_0(1)\rangle = |w\rangle$ of H_P for $s \gtrsim s^*$, switching rapidly from one state to the other in the vicinity of $s = s^*$. In Section 5.7, we show that up to terms of order $1/n$, the ground state $|\psi_+\rangle$ and the first excited state $|\psi_-\rangle$ of $H(s^*)$ are the equal superpositions

$$|\psi_\pm\rangle \simeq \frac{1}{\sqrt{2}}(|E_0(0)\rangle \pm |E_0(1)\rangle) \tag{5.36}$$

of the initial and final ground states (which are nearly orthogonal for large n).

If we prepare the system in the state $|E_0(0)\rangle$ and make a perfect measurement of $H(s^*)$ followed by a perfect measurement of $H(1)$, we find the result w with probability $\frac{1}{2}$. The same effect can be achieved with an imperfect measurement, even if the pointer consists of just a single qubit. First consider the measurement of $H(s^*)$ in the state $|E_0(0)\rangle$. After the system and pointer have interacted for a time t according to (5.10) with $r = 1$, the reduced density matrix of the system in the $\{|\psi_+\rangle, |\psi_-\rangle\}$ basis is approximately

$$\frac{1}{2}\begin{pmatrix} 1 & e^{ig(s^*)t/4}\cos[g(s^*)t/4] \\ e^{-ig(s^*)t/4}\cos[g(s^*)t/4] & 1 \end{pmatrix}. \tag{5.37}$$

If we then measure $H(1)$ (i.e., measure in the computational basis), the probability of finding w is approximately

$$\frac{1}{2}\sin^2[g(s^*)t/4]\,. \tag{5.38}$$

To get an appreciable probability of finding w, we choose $t = \Theta(2^{n/2})$.

This approach is similar to the way one can achieve quadratic speedup with the adiabatic algorithm. Schrödinger time evolution governed by (5.2) does not yield quadratic speedup. However, because s^* is independent of w, we can change the Hamiltonian quickly when the gap is big and more slowly when the gap is small. Since the gap is only of size $\sim 2^{-n/2}$ for a region of width $\sim 2^{-n/2}$, the total oracle time with this modified schedule need only be $O(2^{n/2})$. This has been demonstrated explicitly by solving for the optimal schedule using a different beginning Hamiltonian H'_B that is not a sum of local terms [89, 100], but it also holds using the beginning Hamiltonian (5.33).

5.4.3 Other two-measurement algorithms

Note that measuring $H(s^*)$ is not the only way to solve the Grover problem by measurement. More generally, we can start in some w-independent state, measure

the operator

$$\tilde{H} = H_w + K \tag{5.39}$$

where K is also independent of w, and then measure in the computational basis. For example, suppose we choose

$$K = 1 - |\psi\rangle\langle\psi| \,, \tag{5.40}$$

where $|\psi\rangle$ is a w-independent state with the property $|\langle w|\psi\rangle| \sim 2^{-n/2}$ for all w. (If we are only interested in the time for which we use the black box shown in Fig. 5.2(c), i.e., if we are only interested in the oracle query complexity, then we need not restrict K to be a sum of local terms.) In (5.40), the coefficient of -1 in front of $|\psi\rangle\langle\psi|$ has been fine-tuned so that $(|\psi\rangle + |w\rangle)/\sqrt{2}$ is the ground state of $\tilde{H}$ for large n, up to terms of order $2^{-n/2}$. If the initial state has a large overlap with $|\psi\rangle$, then the measurement procedure solves the Grover problem. However, the excited state $(|\psi\rangle - |w\rangle)/\sqrt{2}$ is also an eigenstate of $\tilde{H}$ up to terms of order $2^{-n/2}$, and it is degenerate with the ground state to this order, so there is a gap in the spectrum of order $2^{-n/2}$. Thus the time to perform the measurement must be $\Omega(2^{n/2})$.

The measurement procedures described above saturate the well-known lower bound on the time required to solve the Grover problem. Using an oracle like the one shown in Fig. 5.2(a), Bennett *et al.* showed that the Grover problem cannot be solved on a quantum computer using fewer than of order $2^{n/2}$ oracle queries [14]. By a straightforward modification of their argument, the same result applies using the oracle shown in Fig. 5.2(c). Thus every possible $\tilde{H}$ as in (5.39) that can be measured to find w must have a gap between the energies of the relevant eigenstates of order $2^{-n/2}$ or smaller.

5.5 Discussion

We have described a way to solve combinatorial search problems on a quantum computer using only a sequence of measurements to keep the computer near the ground state of a smoothly varying Hamiltonian. The basic principle of this algorithm is similar to quantum computation by adiabatic evolution, and the running times of the two methods are closely related. Because of this close connection, many results on adiabatic quantum computation can be directly imported to the measurement algorithm — for example, its similarities and differences with classical simulated annealing [37]. We have also shown that the measurement algorithm can achieve quadratic speedup for the Grover problem using knowledge of the place where the gap is smallest, as in adiabatic quantum computation.

One of the advantages of adiabatic quantum computation is its inherent robustness against error [24]. In adiabatic computation, the particular path from H_B to H_P is unimportant as long as the initial and final Hamiltonians are correct, the path is smoothly varying, and the minimum gap along the path is not too small. Exactly the same considerations apply to the measurement algorithm. However, the adiabatic algorithm also enjoys robustness against thermal transitions out of the ground state: if the temperature of the environment is much smaller than the gap, then such transitions are suppressed. The measurement algorithm might not possess this kind of robustness, since the Hamiltonian of the quantum computer during the measurement procedure is not simply $H(s)$.

Although it does not provide a computational advantage over quantum computation by adiabatic evolution, the measurement algorithm is an alternative way to solve general combinatorial search problems on a quantum computer. The algorithm can be simply understood in terms of measurements of a set of operators, without reference to unitary time evolution. Nevertheless, we have seen that to understand the running time of the algorithm, it is important to understand the dynamical process by which these measurements are realized.

5.6 Details: The measurement process

In Section 5.3, we discussed the running time of the measurement algorithm by examining the measurement process. In this Section, we present the analysis in greater detail. First, we derive the bound on the running time by demonstrating (5.25) and (5.26). We show rigorously that these bounds are sufficient as long as the number of qubits used to represent the pointer is $r = O(\log n)$. Finally, we argue that $r = 1$ qubit should be sufficient in general.

Our goal is to find a bound on the final success probability of the measurement algorithm. We consider the effect of the measurements on the reduced density matrix of the system, which can be written as the block matrix

$$\rho = \begin{pmatrix} \mu & \nu^\dagger \\ \nu & \chi \end{pmatrix} \tag{5.41}$$

where $\mu = \rho_{00}$, $\nu_a = \rho_{a0}$ for $a \neq 0$, and $\chi_{ab} = \rho_{ab}$ for $a, b \neq 0$. Since $\operatorname{tr} \rho = 1$, $\mu = 1 - \operatorname{tr} \chi$. For ease of notation, we suppress the index of the iteration except where necessary. The unitary transformation (5.12) may also be written as a block matrix. Define $\epsilon = \Gamma\delta/g$. Using perturbation theory and the unitarity constraint, we can write

$$U = \begin{pmatrix} u & -w^\dagger V + O(\epsilon^3) \\ w & V + O(\epsilon^2) \end{pmatrix}, \tag{5.42}$$

where $|u|^2 \geq 1 - \epsilon^2 + O(\epsilon^3)$, $\|w\|^2 \leq \epsilon^2 + O(\epsilon^3)$, and V is a unitary matrix. We let $\|\cdot\|$ denote the l_2 vector or matrix norm as appropriate. Furthermore, let

$$\kappa = \begin{pmatrix} 1 & k^\dagger \\ k & J \end{pmatrix}. \tag{5.43}$$

From (5.11), the effect of a single measurement may be written

$$\rho' = (U\rho U^\dagger) \circ \kappa\,, \tag{5.44}$$

where $\circ$ denotes the element-wise (Hadamard) product. If we assume $\|\nu\| = O(\epsilon)$, we find

$$\mu' = |u|^2\mu - w^\dagger V\nu - \nu^\dagger V^\dagger w + O(\epsilon^3) \tag{5.45}$$

$$\nu' = [V\nu + \mu w - V\chi V^\dagger w + O(\epsilon^2)] \circ k\,. \tag{5.46}$$

Now we use induction to show that our assumption always remains valid. Initially, $\nu^{(0)} = 0$. Using the triangle inequality in (5.46), we find

$$\left\|\nu'\right\| \leq [\|\nu\| + \epsilon + O(\epsilon^2)]\tilde{k}\,, \tag{5.47}$$

where

$$\tilde{k} = \max_{j,a} \left|k_a^{(j)}\right|\,. \tag{5.48}$$

So long as $\tilde{k} < 1$, we can sum a geometric series, extending the limits to go from 0 to ∞, to find

$$\left\|\nu^{(j)}\right\| \leq \frac{\epsilon}{1-\tilde{k}} + O(\epsilon^2) \tag{5.49}$$

for all j. In other words, $\|\nu\| = O(\epsilon)$ so long as $\tilde{k}$ is bounded below 1 by a constant.

Finally, we put a bound on the final success probability $\mu^{(M)}$. Using the Cauchy-Schwartz inequality in (5.45) gives

$$\mu' \geq (1-\epsilon^2)\mu - \frac{2\epsilon^2}{1-\tilde{k}} + O(\epsilon^3)\,. \tag{5.50}$$

Iterating this bound M times with the initial condition $\mu^{(0)} = 1$, we find

$$\mu^{(M)} \geq 1 - \frac{\Gamma^2}{Mg^2}\left(1 + \frac{2}{1-\tilde{k}}\right) + O(M\epsilon^3)\,. \tag{5.51}$$

If $\tilde{k}$ is bounded below 1 by a constant (independent of n), we find the condition (5.25) as claimed in Section 5.3.

The requirement on $\tilde{k}$ gives the bound (5.26) on the measurement time τ, and also gives a condition on the number of pointer qubits r. To see this, we

must investigate properties of the function $|\kappa(x)|^2$ defined in (5.15) and shown in Fig. 5.1. It is straightforward to show that $|\kappa(x)|^2 \leq 1/2$ for $\pi/2 \leq x \leq \pi(2^r - 1/2)$. Thus, if we want $\tilde{k}$ to be bounded below 1 by a constant, we require

$$\pi/2 \leq [E_a(s) - E_0(s)]t/2 \leq \pi(2^r - 1/2) \tag{5.52}$$

for all s and for all $a \neq 0$. The left hand bound with $a = 1$ gives $t \geq \pi/g$, which is (5.26). Requiring the right-hand bound to hold for the largest energy difference gives the additional condition $2^r \gtrsim (E_{2^n-1} - E_0)/g$. In general, the largest possible energy difference must be bounded by a polynomial in n. If we further suppose that g is only polynomially small, this condition is satisfied by taking

$$r = O(\log n)\,, \tag{5.53}$$

as claimed at the end of Section 5.3. Thus we see that the storage requirements for the pointer are rather modest.

However, the pointer need not comprise even this many qubits. Since the goal of the measurement algorithm is to keep the system close to its ground state, it would be surprising if the energies of highly excited states were relevant. Suppose we take $r = 1$; then $|\kappa(x)|^2 = \cos^2(x/2)$. As before, (5.26) suffices to make $|\kappa_{01}|^2$ sufficiently small. However, we must also consider terms involving $|\kappa_{0a}|^2$ for $a > 1$. The algorithm will fail if the term $\mu w \circ k$ in (5.46) accumulates to be $O(1)$ over M iterations. This will only happen if, for $O(M)$ iterations, most of $\|w\|$ comes from components w_a with $(E_a - E_0)t$ close to an integer multiple of 2π. In such a special case, changing t will avoid the problem. An alternative strategy would be to choose t from a random distribution independently at each iteration.

5.7 Details: Eigenstates in the Grover problem

Here, we show that the ground state of $H(s^*)$ for the Grover problem is close to (5.36). Our analysis follows Section 4.2 of [38].

Since the Grover problem is invariant under the choice of w, we consider the case $w = 0$ without loss of generality. In this case, the problem can be analyzed in terms of the total spin operators

$$S_a = \frac{1}{2}\sum_{j=1}^{n} \sigma_a^{(j)}\,, \tag{5.54}$$

where $a = x, y, z$ and $\sigma_a^{(j)}$ is the Pauli a operator acting on the jth qubit. The Hamiltonian commutes with $\vec{S}^2 = S_x^2 + S_y^2 + S_z^2$, and the initial state has $\vec{S}^2 = \frac{n}{2}(\frac{n}{2}+1)$, so we can restrict our attention to the $(n+1)$-dimensional subspace of states with this value of $\vec{S}^2$. In this subspace, the eigenstates of the total spin operators satisfy

$$S_a|m_a = m\rangle = m|m_a = m\rangle \tag{5.55}$$

for $m = -\frac{n}{2}, -\frac{n}{2}+1, \ldots, \frac{n}{2}$. Written in terms of the total spin operators and eigenstates, the Hamiltonian is

$$\begin{aligned} H(s) &= (1-s)\left(\frac{n}{2} - S_x\right) \\ &\quad + s\left(1 - \left|m_z = \frac{n}{2}\right\rangle\left\langle m_z = \frac{n}{2}\right|\right)\,. \end{aligned} \tag{5.56}$$

The initial and final ground states are given by $|E_0(0)\rangle = |m_x = \frac{n}{2}\rangle$ and $|E_0(1)\rangle = |m_z = \frac{n}{2}\rangle$, respectively.

Projecting the equation $H(s)|\psi\rangle = E|\psi\rangle$ onto the eigenbasis of S_x, we find

$$\left\langle m_x = \frac{n}{2} - r\middle|\psi\right\rangle = \frac{s}{1-s}\frac{\sqrt{P_r}}{r-\lambda}\left\langle m_z = \frac{n}{2}\middle|\psi\right\rangle\,, \tag{5.57}$$

where we have defined $\lambda = (E-s)/(1-s)$ and $P_r = 2^{-n}\binom{n}{r}$. Now focus on the ground state $|\psi_+\rangle$ and the first excited state $|\psi_-\rangle$ of $H(s^*)$. By equation (4.39) of [38], these states have $\lambda_\pm = \mp\frac{n}{2}2^{-n/2}(1+O(1/n))$. Putting $r = 0$ in (5.57) and taking $s = s^*$ from (5.34), we find

$$\left\langle m_x = \frac{n}{2}\middle|\psi_\pm\right\rangle = \pm\left\langle m_z = \frac{n}{2}\middle|\psi_\pm\right\rangle(1+O(1/n))\,. \tag{5.58}$$

For $r \neq 0$, we have

$$\begin{aligned} \left\langle m_x = \frac{n}{2} - r \middle| \psi_\pm \right\rangle &= \frac{n}{2}\frac{\sqrt{P_r}}{r} \left\langle m_z = \frac{n}{2} \middle| \psi_\pm \right\rangle \\ &\quad \times (1 + O(1/n)) \,. \end{aligned} \tag{5.59}$$

Requiring that $|\psi_\pm\rangle$ be normalized, we find

$$1 = \sum_{r=0}^{n} \left| \left\langle m_x = \frac{n}{2} - r \middle| \psi_\pm \right\rangle \right|^2 \tag{5.60}$$

$$= \left| \left\langle m_z = \frac{n}{2} \middle| \psi_\pm \right\rangle \right|^2 \left(1 + \frac{n^2}{4} \sum_{r=1}^{n} \frac{P_r}{r^2} \right) \times (1 + O(1/n)) \tag{5.61}$$

$$= \left| \left\langle m_z = \frac{n}{2} \middle| \psi_\pm \right\rangle \right|^2 (2 + O(1/n)) \,, \tag{5.62}$$

which implies $|\langle m_z = \frac{n}{2}|\psi_\pm\rangle|^2 = \frac{1}{2} + O(1/n)$. From (5.58), we also have $|\langle m_x = \frac{n}{2}|\psi_\pm\rangle|^2 = \frac{1}{2} + O(1/n)$. Thus we find

$$|\psi_\pm\rangle \simeq \frac{1}{\sqrt{2}} \left(\left| m_x = \frac{n}{2} \right\rangle \pm \left| m_z = \frac{n}{2} \right\rangle \right) \tag{5.63}$$

up to terms of order $1/n$, which is (5.36).

Bibliography

[1] D. S. Abrams and S. Lloyd. Quantum algorithm providing exponential speed increase for finding eigenvalues and eigenvectors. *Phys. Rev. Lett.*, 83:5162, 1999.

[2] D. Aharonov and M. Ben-Or. Fault tolerant computation with constant error. In *Proceedings of the Twenty-Ninth Annual ACM Symposium on the Theory of Computing*, pages 176–188, New York, NY, 1997. ACM. quant-ph/9611025; D. Aharonov and M. Ben-Or, "Fault-tolerant quantum computation with constant error rate," quant-ph/9906129 (1999).

[3] D. Aharonov, M. Ben-Or, R. Impagliazzo, and N. Nisan. Limitations of noisy reversible computation, 1996. quant-ph/9611028.

[4] Y. Aharonov and M. Vardi. Meaning of an individual "Feynman path". *Phys. Rev. D*, 321:2235, 1980.

[5] C. Ahn, A. C. Doherty, and A. J. Landahl. Continuous quantum error correction via quantum feedback control. *Phys. Rev. A*, 65:042301, 2002. quant-ph/0110111.

[6] M. G. Alford, K.-M. Lee, J. March-Russell, and J. Preskill. Quantum field theory of nonabelian strings and vortices. *Nucl. Phys. B*, 384:251–317, 1992. hep-th/9112038.

[7] A. Aspect, J. Dalibard, and G. Roger. Experimental test of Bell's inequalities using time-varying analyzers. *Phys. Rev. Lett.*, 49:1804–1807, 1982.

[8] F. Barahona, R. Maynard, R. Rammal, and J. P. Uhry. Morphology of ground states of a two-dimensional frustration model. *J. Phys. A.*, 15:673–699, 1982.

[9] A. Barenco, T. A. Brun, R. Schack, and T. Spiller. Effects of noise on quantum error correction algorithms. *Phys. Rev. A*, 56:1177–1188, 1997. quant-ph/9612047.

[10] J. P. Barnes and W. S. Warren. Automatic quantum error correction. *Phys. Rev. Lett.*, 85(4):856–859, 2000. quant-ph/9912104.

[11] D. Beckman, D. Gottesman, A. Kitaev, and J. Preskill. Measurability of Wilson loop operators. *Phys. Rev. D*, 65:065022, 2001. quant-ph/0110205.

[12] D. Beckman, D. Gottesman, M. Nielsen, and J. Preskill. Causal and localizable quantum operations. *Phys. Rev. A*, 64:052309, 2001. quant-ph/0102043.

[13] J. S. Bell. On the Einstein-Podolsky-Rosen paradox. *Physics*, 1:195–200, 1964. Reprinted in J. S. Bell, *Speakable and unspeakable in quantum mechanics*, Cambridge University Press, Cambridge, 1987.

[14] C. H. Bennett, E. Bernstein, G. Brassard, and U. Vazirani. Strengths and weaknesses of quantum computing. *SIAM J. Comput.*, 26:1510, 1997.

[15] C. H. Bennett and G. Brassard. Quantum cryptography: Public-key distribution and coin tossing. In *Proceedings of IEEE International Conference on Computers, Systems and Signal Processing*, pages 175–179, Bangalore, India, 1984. IEEE Press. C. H. Bennett and G. Brassard, "Quantum public key distribution," IBM Technical Disclosure Bulletin **28**, 3153–3163 (1985).

[16] C. H. Bennett and S. J. Wiesner. Communication via one- and two-particle operators on Einstein-Podolsky-Rosen states. *Phys. Rev. Lett.*, 69(20):2881–2884, November 1992.

[17] S. B. Bravyi and A. Yu. Kitaev. Quantum codes on a lattice with boundary, 1998. quant-ph/9810052.

[18] A. R. Calderbank, E. M. Rains, P. W. Shor, and N. J. A. Sloane. Quantum error correction and orthogonal geometry. *Phys. Rev. Lett.*, 78:405, 1997. quant-ph/9605005.

[19] A. R. Calderbank and P. W. Shor. Good quantum error-correcting codes exist. *Phys. Rev. A*, 54:1098, 1996. quant-ph/9512032.

[20] H. J. Carmichael. *An Open Systems Approach to Quantum Optics*. Springer-Verlag, Berlin, 1993.

[21] C. Caves. Resource material for promoting the bayesian view of everything, 2001. http://info.phys.unm.edu/~caves/.

[22] A. M. Childs, E. Farhi, E. Deotto, J. Goldstone, S. Gutmann, and A. Landahl. Quantum search by measurement, 2002. quant-ph/0204013.

[23] A. M. Childs, E. Farhi, J. Goldstone, and S. Gutmann. Finding cliques by quantum adiabatic evolution, 2000. quant-ph/0012104.

[24] A. M. Childs, E. Farhi, and J. Preskill. Robustness of adiabatic quantum computation. *Phys. Rev. A*, 65:012322, 2002.

[25] I. L. Chuang and Y. Yamamoto. The persistent qubit, 1996. quant-ph/9604030.

[26] R. Cleve, A. Ekert, C. Macchiavello, and M. Mosca. Quantum algorithms revisited. *Proc. Roy. Soc. London A*, 454:339, 1998. quant-ph/9905027.

[27] P.-G. de Gennes. *Scaling Concepts in Polymer Physics*. Cornell University Press, Ithaca, NY, 1970.

[28] E. Dennis. Fault-tolerant computation without concatenation. *Phys. Rev. A*, 63:052314, 2001. quant-ph/9905027.

[29] E. Dennis, A. Kitaev, A. Landahl, and J. Preskill. Topological quantum memory, 2001. quant-ph/0110143.

[30] A. C. Doherty, S. Habib, K. Jacobs, H. Mabuchi, and S. M. Tan. Quantum feedback control and classical control theory. *Phys. Rev. A*, 62:012105, 2000. quant-ph/9912107.

[31] A. C. Doherty and K. Jacobs. Feedback-control of quantum systems using continuous state-estimation. *Phys. Rev. A*, 60:2700, 1999. quant-ph/9812004.

[32] P. Le Doussal and A. B. Harris. Location of the Ising spin-glass multicritical point on Nishimori's line. *Phys. Rev. Lett.*, 61:625–628, 1988.

[33] J. Edmonds. Paths, trees, and flowers. *Canada J. Math*, 17:449–467, 1965.

[34] T. Einarsson. Fractional statistics on a torus. *Phys. Rev. Lett.*, 64:1995, 1990.

[35] A. Einstein. Physics and reality. *Franklin Institute Journal*, 221(3):349–382, 1936. This is the popular paraphrasing of the original quote, "The eternal mystery of the world is its comprehensibility. ... The fact that it is comprehensible is a miracle," reprinted in *Ideas and Opinions, p. 292.*

[36] A. Einstein, B. Podolsky, and N. Rosen. Can the quantum-mechanical description of physical reality be considered complete? *Phys. Rev.*, 46:777–780, 1935.

[37] E. Farhi, J. Goldstone, and S. Gutmann. Quantum adiabatic evolution algorithms versus simulated annealing, 2002. quant-ph/0201031.

[38] E. Farhi, J. Goldstone, S. Gutmann, and M. Sipser. Quantum computation by adiabatic evolution, 2000. quant-ph/0001106.

[39] E. Farhi and S. Gutmann. Analog analogue of a digital quantum computation. *Phys. Rev. A*, 57:2403, 1998.

[40] S. A. Fenner and Y. Zhang. Universal quantum computation with two- and three-qubit projective measurements, 2001. quant-ph/0111077.

[41] M. H. Freedman, A. Kitaev, M. J. Larsen, and Z. Wang. Topological quantum computation, 2001. quant-ph/0101025.

[42] M. H. Freedman and D. A. Meyer. Projective plane and planar quantum codes, 1998. quant-ph/9810055.

[43] C. A. Fuchs. Information gain vs. state disturbance in quantum theory, 1996. quant-ph/9611010.

[44] P. Gaćs. Reliable computation with cellular automata. *J. Comp. Sys. Sci.*, 32:15, 1986.

[45] C. W. Gardiner. *Handbook of Stochastic Methods*. Springer, Berlin, 1985.

[46] P. Goetsch, P. Tombesi, and D. Vitali. Effect of feedback on the decoherence of a Schrödinger cat state: A quantum trajectory description. *Phys. Rev. A*, 54(5):4519–4527, November 1996.

[47] D. Gottesman. A class of quantum error-correcting codes saturating the quantum Hamming bound. *Phys. Rev. A*, 54:1862, 1996. quant-ph/9604038.

[48] D. Gottesman. *Stabilizer codes and quantum error correction*. Ph.D. thesis, Caltech, 1997. quant-ph/9705052.

[49] D. Gottesman. The Heisenberg representation of quantum computers, 1998. quant-ph/9807006.

[50] D. Gottesman. A theory of fault-tolerant quantum computation. *Phys. Rev. A*, 57:127, 1998. quant-ph/9702029.

[51] D. Gottesman. Fault-tolerant quantum computation with local gates, 1999. quant-ph/9903099.

[52] D. Gottesman and J. Preskill. unpublished.

[53] L. K. Grover. Quantum mechanics helps in searching for a needle in a haystack. *Phys. Rev. Lett.*, 79:325, 1997.

[54] I. A. Gruzberg, N. Read, and A. W. W. Ludwig. Random-bond Ising model in two dimensions, the Nishimori line, and supersymmetry. *Phys. Rev. B*, 63:104422, 2001. cond-mat/0007254.

[55] A. Honecker, M. Picco, and P. Pujol. Nishimori point in the 2D $\pm j$ random-bond Ising model, 2000. cond-mat/00010143.

[56] O. L. R. Jacobs. *Introduction to Control Theory*. Oxford, New York, NY, 1993.

[57] N. Kawashima and T. Aoki. Zero-temperature critical phenomena in two-dimensional spin glasses, 1999. cond-mat/9911120.

[58] A. Yu. Kitaev. unpublished.

[59] A. Yu. Kitaev. Quantum measurements and the abelian stabilizer problem, 1995. quant-ph/9511026.

[60] A. Yu. Kitaev. Quantum error correction with imperfect gates. In O. Hirota, A. S. Holevo, and C. M. Caves, editors, *Proceedings of the Third International Conference on Quantum Communication, Computing and Measurement*, New York, NY, 1996. Plenum Press.

[61] A. Yu. Kitaev. Fault-tolerant quantum computation by anyons, 1997. quant-ph/9707021.

[62] A. Yu. Kitaev. Quantum computations: algorithms and error correction. *Russian Math. Surveys*, 52:1191–1249, 1997.

[63] A. Yu. Kitaev, A. H. Shen, and M. N. Vyalyi. *Classical and Quantum Computation*, volume 47 of *Graduate Studies in Mathematics*. American Mathematical Society, Providence, RI, 2002.

[64] H. Kitatani. The verticality of the ferromagnetic-spin glass phase boundary of the $\pm j$ Ising model in the p–t plane. *J. Phys. Soc. Japan*, 61:4049–4055, 1992.

[65] P. L. Kloeden, E. Platen, and H. Schurz. *Numerical solution of SDE through computer experiments*. Springer-Verlag, Berlin, 1994.

[66] E. Knill and R. Laflamme. A theory of quantum error-correcting codes. *Phys. Rev. A*, 55:900–911, 1997. quant-ph/9604034.

[67] E. Knill, R. Laflamme, and W. H. Zurek. Resilient quantum computation: error models and thresholds. *Proc. Roy. Soc. London A*, 454:365–384, 1998. quant-ph/9702058.

[68] A. N. Korotkov. Selective evolution of a qubit state due to continuous measurement. *Phys. Rev. B*, 63:115403, 2001. cond-mat/0008461.

[69] K. Kraus. *States, Effects, and Operations: Fundamental Notions of Quantum Theory*, volume 190 of *Lecture Notes in Physics*. Springer-Verlag, Berlin, 1983.

[70] D. W. Leung. Two-qubit projective measurements are universal for quantum computation, 2001. quant-ph/0111122.

[71] S. Lloyd. Universal quantum simulators. *Science*, 273:1073, 1996.

[72] S. Lloyd and J.-J. E. Slotine. Quantum feedback with weak measurements. *Phys. Rev. A*, 62:012307, 2000. quant-ph/9905064.

[73] H. Mabuchi and P. Zoller. Inversion of quantum jumps in quantum optical systems under continuous observation. *Phys. Rev. Lett.*, 76(17):3108–3111, April 1996.

[74] N. Madras and G. Slade. *The Self-Avoiding Walk*. Birkhäuser, Boston, 1996.

[75] J. C. Maxwell. *Matter and Motion*. Dover, Mineola, NY, 1991.

[76] F. Merz and J. T. Chalker. The two-dimensional random-bond Ising model, free fermions and the network model, 2001. cond-mat/0106023.

[77] A. Messiah. *Quantum Mechanics*. Dover, Mineola, NY, 1999.

[78] M. A. Nielsen. Computable functions, quantum measurements, and quantum dynamics. *Phys. Rev. Lett.*, 79:2915–2918, 1997. quant-ph/9706006.

[79] M. A. Nielsen. Universal quantum computation using only projective measurement, quantum memory, and preparation of the $|0\rangle$ state, 2001. quant-ph/0108020.

[80] M. A. Nielsen and I. L. Chuang. *Quantum Computation and Quantum Information*. Cambridge University Press, Cambridge, 2000.

[81] H. Nishimori. Internal energy, specific-heat and correlation-function of the bond-random Ising-model. *Prog. Theor. Phys.*, 66:1169, 1981.

[82] H. Nishimori. Geometry-induced phase transition in the $\pm j$ Ising model. *J. Phys. Soc. Japan*, 55:3305–3307, 1986.

[83] W. Ogburn and J. Preskill. Topological quantum computation. *Lecture Notes in Computer Science*, 1509:341–356, 1999.

[84] J. P. Paz and W. Zurek. Continuous error correction. *Proc. Trans. R. Soc. Lond. A*, 454:355–364, 1998. quant-ph/9707049.

[85] J. Preskill. Fault-tolerant quantum computation. In H. K. Lo, S. Popescu, and T. Spiller, editors, *Introduction to Quantum Computation and Information*, chapter 8, page 213. World Scientific, New Jersey, 1998. quant-ph/9712048.

[86] J. Preskill. Lecture notes for Caltech course Ph 219: Quantum Information and Computation, 1998. http://www.theory.caltech.edu/~preskill/ph219/.

[87] J. Preskill. Reliable quantum computers. *Proc. Roy. Soc. London A*, 454:385–410, 1998. quant-ph/9705031.

[88] R. Raussendorf and H. J. Briegel. Quantum computing via measurements only, 2000. quant-ph/0010033.

[89] J. Roland and N. Cerf. Quantum search by local adiabatic evolution, 2001. quant-ph/0107015.

[90] L. S. Schulman, A. Ranfagni, and D. Mugnai. Characteristic scales for dominated time evolution. *Physica Scripta*, 49:536, 1993.

[91] P. W. Shor. Algorithms for quantum computation: discrete logarithms and factoring. In S. Goldwasser, editor, *Proceedings, 35th Annual Symposium on Foundations of Computer Science*, page 124, Los Alamitos, CA, 1994. IEEE Press.

[92] P. W. Shor. Scheme for reducing decoherence in quantum computer memory. *Phys. Rev. A*, 52:2493, 1995.

[93] P. W. Shor. Fault-tolerant quantum computation. In *Proceedings, 37th Annual Symposium on Foundations of Computer Science*, pages 56–65, Los Alamitos, CA, 1996. IEEE Press. quant-ph/9605011.

[94] A. Steane. Error-correcting codes in quantum theory. *Phys. Rev. Lett.*, 77:793, 1996.

[95] A. Steane. Multiple particle interference and quantum error correction. *Proc. Roy. Soc. London A*, 452:2551, 1996. quant-ph/9601029.

[96] A. Steane. Active stabilization, quantum computation, and quantum state synthesis. *Phys. Rev. Lett.*, 78:2552, 1997. quant-ph/9611027.

[97] P. Tombesi and D. Vitali. Macroscopic coherence via quantum feedback. *Phys. Rev. A*, 51(6):4913–4917, June 1995.

[98] A. L. Toom. Stable and attractive trajectories in multicomponent systems. In R. L. Dobrushin, editor, *Advances in Probability*, volume 6, pages 549–575. Dekke, New York, NY, 1980.

[99] B. C. Travaglione, G. J. Milburn, and T. C. Ralph. Phase estimation as a quantum nondemolition measurement, 2002. quant-ph/0203130.

[100] W. van Dam, M. Mosca, and U. Vazirani. How powerful is adiabatic quantum computation? to appear in FOCS 2001.

[101] C. Vanderzande. *Lattice Models of Polymers*. Cambridge University Press, Cambridge, U. K., 1998.

[102] L. Viola and S. Lloyd. Dynamical suppression of decoherence in two-state quantum systems. *Phys. Rev. A*, 58:2733, 1998. quant-ph/9803057.

[103] J. von Neumann. *Mathematical Foundations of Quantum Mechanics*. Princeton University Press, Princeton, NJ, 1955.

[104] C. Wang, 2002. Private communication.

[105] C. Wang and J. Preskill. Confinement Higgs transition in a disordered gauge theory, 2002. In preparation.

[106] J. Wang and H. M. Wiseman. Feedback-stabilization of an arbitrary pure state of a two-level atom: a quantum trajectory treatment, 2000. quant-ph/0008003.

[107] P. Warszawski, H. M. Wiseman, and H. Mabuchi, 2002. Preprint.

[108] X. .G. Wen and Q. Niu. Ground-state degeneracy of the fractional quantum Hall states in the presence of a random potential and on high-genus Riemann surfaces. *Phys. Rev. B*, 41:9377–9396, 1990.

[109] S. Wiesner. Simulations of many-body quantum systems by a quantum computer, 1996. quant-ph/9603028.

[110] D. J. Wineland, C. Monroe, W. M. Itano, D. Leibfried, B. E. King, and D. M. Meekhof. Experimental issues in coherent quantum-state manipulation of trapped atomic ions. *Journal of Research of the National Institute of Standards and Technology*, 103:259–328, 1998.

[111] H. M. Wiseman. Quantum theory of continuous feedback. *Phys. Rev. A*, 49:2133–2150, 1994. Errata in Phys. Rev. A **49**, 5159 (1994), Phys. Rev. A **50**, 4428 (1994).

[112] H. M. Wiseman. Quantum trajectories and quantum measurement theory. *Quantum Semiclass. Opt.*, 8:205, 1996.

[113] H. M. Wiseman and G. J. Milburn. Quantum theory of field-quadrature measurements. *Phys. Rev. A*, 47:642, 1993.

[114] C. Zalka. Efficient simulation of quantum systems by quantum computers. *Proc. Roy. Soc. London A*, 454:313, 1998.

Index