

ON BUILDING A FAULT-TOLERANT QUANTUM COMPUTER

A Dissertation

Presented to the Faculty of the Graduate School

of Cornell University

in Partial Fulfillment of the Requirements for the Degree of

Doctor of Philosophy

by

Kristina Renee Colladay

December 2021

© 2021 Kristina Renee Colladay
ALL RIGHTS RESERVED

ON BUILDING A FAULT-TOLERANT QUANTUM COMPUTER

Kristina Renee Colladay, Ph.D.

Cornell University 2021

This thesis describes a variety of topics and techniques for building a scalable quantum computer. Some topics discussed are within the field of active error correction: magic state distillation, stabilizer code conversion, and bounds on thresholds and efficiency for quantum decoders. In particular, we present an algorithm that produces a mapping between any two stabilizer codes, either by performing unitary operations or through projective measurements. Another piece discussed is the efficient simulation of novel quantum hardware using tensor network techniques. We present some results from our model of a family of superinductors. Lastly, we discuss our driven-dissipative approach to preparing quantum states. We give a protocol for producing an optical analog of the Laughlin state in an open quantum system using AMO techniques – a step towards creating many-body fractional quantum Hall states having non-Abelian excitations.

BIOGRAPHICAL SKETCH

Kristina Renee Colladay's journey in physics started at a young age, mostly with curiosity about the things that surrounded her. She wanted to know why things worked in nature like they did. At her small school's science fair, she won first place in the physical sciences and eventually decided that physics was the way to go instead of music. She decided to start college at a local branch of Pennsylvania State University where she was inspired by the teachings of her physics professor and decided to continue to the main campus as a physics major. As an undergraduate, she chose research experiences in AMO physics and in math. After graduating, she went to the NASA Academy and learned about high energy physics. At the Academy, some of her dreams came true when she met astronauts with the real right stuff, including Neil Armstrong! Then she took a position as a physicist at the Naval Research Laboratory in Washington, DC in microwave photonics. At that time she also attended Johns Hopkins University where she did research in quantum key distribution. After graduating, she decided that she wanted to pursue a PhD but was not sure about which field to choose. She started at Cornell University and did research in AMO. Then she went to an internship at Northrop Grumman as a GEM fellow, and afterwards decided to pursue research in quantum computing theory for her Ph.D. During her time pursuing her Ph.D. at Cornell, she took a position as a physicist at Northrop Grumman where she learned about quantum error correction and quantum hardware. She brought back knowledge to Cornell which inspired some parts of the work done in this thesis with her advisor, Erich Mueller. She currently loves quantum computing theory and AMO theory, along with gardening, taking care of her chickens, and skiing in the Colorado mountains.

This thesis is dedicated to anyone who would choose to help people in need.

Isaiah 58:6-7

ACKNOWLEDGEMENTS

First, I must express my sincere gratitude to my advisor, Erich Mueller, without whom none of this work would exist. I am also grateful for the opportunity to be among his extraordinary group of students, visiting students, and postdocs, with whom I have had many conversations, intellectual exchanges, and encouragement through tough challenges as a graduate student. I am grateful to my graduate committee members, Greg Fuchs and Marcelo Aguiar, for their instruction and guidance, for their flexibility in allowing me to pursue an unconventional yet rewarding path of learning, and for their support during challenging times. Second, I am grateful to Bryan Eastin, Jonas Anderson, Ryan Epstein, and Matthew Wiepert for sharing their valuable insights in the field of quantum error correction and quantum hardware. Their mentorship and influence was invaluable to me during my entire career as a graduate student. Third, I am grateful to David Ferguson, who worked with me tirelessly on superconducting circuit simulations. Despite the odds, we gave talk at March Meeting and we got so very close to publishing a manuscript on simulating flux-tunable superinductors! I am grateful for his mentoring during my time as a student. I must also express my gratitude to Manny Knill and his group at NIST Boulder, with whom I have had only a few yet incredibly enlightening conversations. I am also grateful to David Clarke and Mike Mullan for conversations on quantum computing theory, and to Tony Przybysz, Joel Strand, Steven Sendelbach, and Ofer Naaman for their inspiration in quantum computing when I was just starting out as an intern in their lab! Many thanks to Jon Simon and his group in Chicago for fruitful conversations and feedback. Finally, I am grateful to John and Jane Colman for their financial support at Cornell at a time when I most needed it.

I am grateful for the support and prayers of my family.

And which of you by worrying can add a day to his life's span? Therefore if you cannot do even a very little thing, why do you worry about the other things? Consider the lilies, how they grow: they neither labor nor spin; but I tell you, not even Solomon in all his glory clothed himself like one of these. Now if God so clothes the grass in the field, which is alive today and tomorrow is thrown into the furnace, how much more will He clothe you? You of little faith! Luke 12:25-28 (NASB)

TABLE OF CONTENTS

Biographical Sketch	iii
Dedication	iv
Acknowledgements	v
Table of Contents	vii
List of Tables	ix
List of Figures	x
1 Introduction	1
2 Universal Quantum Computation in the Circuit Model	4
2.1 Magic State Distillation	7
2.1.1 State Injection	15
3 Stabilizer Conversion	17
3.1 Introduction	17
3.2 Algorithm	19
3.2.1 Stabilizer Codes	20
3.2.2 Moving in the space of stabilizer codes	20
3.2.3 Constructing a sequence of stabilizer codes	22
3.2.4 Construction of the generators and complementary operators	23
3.2.5 Distance Bounds	25
3.2.6 Fault Tolerance	28
3.2.7 Implementing Constraints	28
3.3 Applications	29
3.3.1 Universal transversal computing	29
3.3.2 Moving defects in topological codes	31
3.4 Summary and Outlook	35
3.5 Appendix	36
3.5.1 Properties of U	37
3.5.2 Fault-tolerant measurement of Pauli operators	38
3.5.3 Trivial Examples	39
3.5.4 Code distance during rewiring	40
3.5.5 Connectivity matrices for mapping between Steane and Reed-Muller Codes	41
3.5.6 Common Quantum Circuit Operators	45
4 Some Bounds for Quantum Error-Correcting Code Decoders	46
4.1 Upper Bounds on Thresholds of Error Correcting Codes	47
4.1.1 A Note on Faulty Syndromes	56
4.2 Upper Bounds on the Efficiency of Error-correcting Codes	57

5	Superconducting Qubit Design	60
5.1	The Josephson Junction	63
5.1.1	First Josephson Relation	65
5.1.2	Second Josephson Relation	67
5.1.3	Consequences of the Relations	68
5.2	The Transmon and Fluxonium	69
5.3	Superinductor	72
5.4	Harmonic Analysis and Black Box Quantization	73
5.5	Simulation Techniques	77
5.6	Tensor Networks for Qubit Simulation	80
5.7	MPS-MPO-MPS Form for DMRG	82
5.7.1	Example MPO Construction for a Superconducting Network of Capacitors and Josephson Junctions	85
5.8	Understanding the MPO: Some MPO Constructions for Supercon- ducting Circuits	87
5.9	Understanding the MPS: the AKLT state	90
5.9.1	Symmetry of AKLT state	99
5.9.2	W-state	102
5.10	Using DMRG to Model the Superinductor	104
5.10.1	The DMRG Software Tool	105
5.10.2	Low-Energy Spectrum for a Family of Superinductors	109
6	Dissipation Engineering in Open Quantum Systems	114
7	Driven dissipative preparation of few-body Laughlin states of Ry- dberg polaritons in twisted cavities	116
7.1	Introduction	116
7.2	Modeling	118
7.3	Numerical Techniques	123
7.4	Validation	124
7.5	Driven-Dissipative Preparation of the 2-polariton Laughlin State	130
7.6	Measurements	136
7.7	Summary and Discussion	138
7.8	Acknowledgments	139
7.9	Appendix	139
7.9.1	Microscopic Model of Interacting Rydberg Polaritons in Nearly Degenerate Multimode Cavity	139
7.9.2	Multi-photon Drive	144
7.9.3	Lindblad Superoperator	145
7.9.4	A Note about Long-lived Rydberg States	149
7.9.5	Additional Notes on the Microscopic Model of Interacting Rydberg Polaritons	151
	Bibliography	155

LIST OF TABLES

3.1	Generators for Steane code ($g_0 - g_5$) and Reed-Muller codes ($g'_0 - g'_{13}$). Additional qubits with arbitrarily chosen stabilizer generators ($g_6 - g_{13}$) are appended to the Steane code for the conversion. . . .	30
3.2	Connectivity matrix for the Steane and Reed-Muller codes. A one in the row indicates an anti-commutation relation, and zeros are inserted where the stabilizer generators are identical.	42
3.3	Connection matrices for intermediate steps of the row reduction used to find the generators needed to map from the Steane code to the Reed-Muller code.	43
3.4	Connection matrices for intermediate steps of the column reduction used to find the generators needed to map from the Reed-Muller code to the Steane code.	44
4.1	Qudit hashing bounds for generalized bit-flip (GBF), the bit- and phase- flip (BPF), the depolarizing noise (DPN), and the GBF given DPN channels for qudit dimension (D). The hashing bound is conjectured to be an upper bound of the threshold of a quantum decoder (i.e. it represents the highest threshold a quantum decoder can have given an error model and the types of errors the decoder detects).	56
4.2	Qudit hashing bounds for qudit dimension (D) for the scaled generalized bit-flip, GBF(s) and the scaled bit- and phase- flip, BPF(s) channels. The depolarizing noise (DPN), and the GBF given DPN channels are included for comparison.	56
5.1	The four possible states resulting from the contraction and projection in Fig. 5.8. The states in the left column correspond to the four configurations of the top state in Fig. 5.8, where the middle neighboring particles are contracted into a singlet state and the endpoint particles can be combinations of spin up and spin down. The second column corresponds to the projected state, where the remaining spin-1/2 particles on the endpoints are projected into the spin-1 space with the particles on neighboring sites, bottom of Fig. 5.8. This example can be generalized to longer 1D chains of spin-1/2 particles.	92

LIST OF FIGURES

2.1	Bloch sphere and inscribed octahedron formed by the condition in Eqn. 2.3.	8
2.2	Injection circuit for a controlled-H non-Clifford gate, modified from [1]. In the injection circuit, if a Y-error occurs during state preparation of $ H\rangle$, it propagates to a Y gate on the output. When implementing the controlled-H gate if: (1) a Y error from the first $Y\left(\frac{-\pi}{4}\right)$ gate occurs, it propagates to a Z-gate on the control and a Y-gate on the target (2) a Y-error on the second $Y\left(\frac{\pi}{4}\right)$ gate occurs, it translates to a Y-gate after the gate (3) two Y-errors occur, then there is a Z-error on the control.	12
2.3	Magic state distillation circuit modified from [1]. There are two $ H\rangle$ resource states which can have only Y-error on them after twirling. The effective circuit is a parity check circuit which can detect one Y-error on either of the two resource states. Because the parity check circuit requires non-Clifford gates, these gates are faulty, but their faults are made in a predictable way using state-injection, see Fig. 2.1. To catch the errors from the non-Clifford gates, the parity check circuit is embedded into an error-correcting code, in this case, the four-qubit code. Thus the resource $ H\rangle$ states become logical states after the encoding, and the operators in the parity check measurement are logical (i.e. the circuit in the middle performs the parity check in Fig. 2.1). Pauli error patterns produced from the non-Clifford gates are pushed through the circuit, using circuit identities, to the section in yellow. Many of these error patterns (up to order 8, eight errors at once, in all possible ways) are generated and categorized into two categories (either logical error or not) to determine the error polynomial.	14
2.4	Equivalent circuit diagram to the procedure described in [2] for implementing the arbitrary phase gate $\Lambda\left(e^{\pm i\theta}\right)$ by injecting the magic state $ A_\theta\rangle$	16
3.1	Transporting an e type defect in the surface code. Qubits sit at the vertices of the square lattice, yellow (light) and red (dark) plaquettes denote stabilizers corresponding to the product of Z or X operators on the four qubits at the corners. The white squares represent the absence of a stabilizer. To convert from the code on the left to the one on the right, one first measures the X operator highlighted in the middle panel, then the highlighted stabilizer in the right panel.	32
3.2	Changing an e type defect into an m type defect. This conversion requires measuring a non-local operator corresponding to a product of Y operators extending from the defects to the edge of the sample.	33

3.3	Shortening a twist. Twist defects, as illustrated in this figure, contain stabilizers corresponding to products of four or five operators. The ends of these defects have properties analogous to Majorana fermions. One converts the code on the left to the one on the right by measuring the operator $X_2X_3X_6X_7$. Our algorithm similarly gives the procedure for other moves.	35
3.4	Circuit illustrating the cat-state method [3,4] for the fault-tolerant projection measurement of a stabilizer generator.	39
4.1	Phase boundaries estimated from Monte Carlo simulations of the respective disordered Potts gauge glasses representing quantum error correction for d -level quantum systems. The error thresholds for each dimension correspond to the points where the corresponding Nishimori line (dashed) intersects the phase boundary. The values of the hashing bound in the first column of Table 4.1 also compare well with the figure insert. Figure and caption adopted from [5]. . .	54
5.1	Harmonic oscillator LC circuit. Figure adapted from [6].	61
5.2	Cooper pair box (left) and transmon circuit (right). The transmon is a modification of the Cooper pair box, where the capacitor is large and the Josephson junction is flux-tunable. Figure adapted from [6] and [7].	70
5.3	The fluxonium circuit contains a large superinductor composed of many Josephson junctions (left). The junctions collectively act as a large inductor (right). Figure adapted from [8].	73
5.4	Foster network model of passive, reactive circuits. Reactive circuits can be characterized completely by a single port measurement – one hooks up the end leads (open terminals pictured on the left of each circuit) to a infinite-impedance current source or a zero-impedance voltage source. If the circuit is hooked up to a zero-impedance voltage source, then the model best describing the circuit is Fig. 5.4a. The voltage source shorts the first LC loop allowing for its oscillation frequency and characteristic impedance to be measured. If the circuit is hooked up to an infinite-impedance current source, then the best model describing the circuit is Fig. 5.4b, a model of LC oscillators in series. With an infinite-impedance source, a false measurement due to the formation of the additional loop around the entire circuit is avoided.	74

5.5	Example of the admittance of a parallel LC circuit. For a parallel LC circuit, the poles of the impedance correspond to the resonance frequency. Since it is often more convenient to look at the zero instead of the pole to find the resonance frequency, we have inverted the function and have plotted the admittance instead of the impedance. The slope of the admittance near resonance yields the characteristic impedance.	76
5.6	MPS-MPO-MPS tensor network. The local Hilbert space dimension is d , and the bond dimensions of the MPO and MPS are m and k , respectively.	83
5.7	Example superconducting circuit composed of capacitors and Josephson junctions. Each Josephson junction is represented by a nonlinear inductance and capacitance in parallel. The Hamiltonian of the circuit is given in Eqn. 5.23, and the corresponding MPO is given in Eqn. 5.24. This circuit is a minimal form of superinductor [9].	86
5.8	An example spin-1 AKLT state formed from four spin-1/2 particles. The state $P \psi\rangle$ is formed by contracting neighboring pairs into a singlet, represented by the solid bond. The state is then symmetrized by projecting the remaining neighboring spins into triplets, represented by the circular bonds.	91
5.9	Tensor network representation for constructing the AKLT matrix product state. This construction follows the one shown in [10]. . .	95
5.10	Binary tree for the W-state representing the structure of its MPS form.	103
5.11	Software flow diagram of our software tool developed for simulating superconducting circuits. This image is adopted from [11]. In Python, the user draws a superconducting circuit in Circuitizer. Circuitizer then interfaces with the program QuTip to produce the Hamiltonian. An MPO form of this Hamiltonian is produced using algorithms derived from the techniques described in Sections 5.7 and 5.8. Then, the MPO is stored in a json file and imported into ITensor. ITensor performs DMRG and gives the eigenstates in MPS form. The MPS is contracted with the exact Hamiltonian (before compression) to determine the convergence properties of the simulation (i.e. at each iterative step during DMRG, the variance is calculated using the exact Hamiltonian MPO and the eigenstate produced at that step). Upon reaching a stopping criterion, the loop halts. At this time, observables of interest can be calculated using the output MPS. For example, the MPO form for the charge operator, which can include offset charges on the junctions, can be produced in Python, imported into ITensor, and contracted with the MPS to find the charge dispersion effects due to offset charges.	106

5.12	Left: A four-node superinductor drawn in Circuitizer. The user specifies the junction properties and capacitances. Right: Transition energies from the ground state for the first three excited states for a family of superinducting circuits from 5-64 nodes, where the junctions are in the transmon regime. Excited states are obtained by adding a penalty term, see [12]. Image adopted from [11]. . . .	110
5.13	Top: Upper bound on the energy difference (in GHz) between the calculated and actual energies for a family of superinductors having 5-64 nodes.. The worst case errors have an upper error bound of 500Hz corresponding to the peak at 30 nodes for the second and third excited states. Bottom (left): Variance at the last sweep used in calculating the error bound in the top figure. Bottom (right): Comparison of upper error bound and the actual error between DMRG results and exact diagonalization for a five-node superinductor.	111
5.14	Error bound and the change in energy between neighboring sweeps versus sweep number for the first four states. Each step down in the in the error bound corresponds to a drop in the truncation error by one order of magnitude. The truncation error begins at 10^{-8} and ends at 10^{-12}	112
5.15	Comparison of the harmonic analysis and DMRG results for the lowest-lying energies of various superinductors.	112
5.16	Time to solution for the DMRG simulation versus superinductor size. The calculation includes the time required to calculate the variance.	113
7.1	(color online) Schematic level diagram for three-mode system ($ 3\rangle, 6\rangle, 9\rangle$) with no more than four polaritons. Vertical axis represents energy, and horizontal axis is the angular momentum relative to a state formed from atoms in the $ 6\rangle$ mode: $L - 6N$. Dashed horizontal lines are separated by ω_0 , the mode frequencies in the absence of interactions. Numbers under each stack of levels represent the angular momenta of the polaritons in the system. For example, the states labeled with 3, 9 and 6, 6 are superpositions of the states $ 3, 9\rangle$ and $ 6, 6\rangle$. These are split by an energy proportional to the interactions U . The decay rate due to inelastic two-body collisions are represented by the width of each line. There are four long lived states: the vacuum, the single polariton states, and the two polariton Laughlin state. Green arrows are representative single-photon transitions, and brown wavy arrows indicate decay channels from the four photon manifold to the Laughlin state.	122

7.2	(color online) Energy-level diagram depicting the relevant states in the experiment of Clark et al. [13], analyzed in Sec. 7.4. A resonant drive connects $ \text{vac}\rangle$, $ 6\rangle$ and $ L\rangle$. The modes $ 3\rangle$ and $ 9\rangle$ can become occupied through single-photon decay from $ L\rangle$	125
7.3	(color online) Time evolution of the probability of being in the Laughlin state for the scenario in Fig. 7.2. Here γ_1 is the single-polariton loss rate, and $\lambda = \lambda_6$ is the drives strength. Solid blue: $\lambda/\gamma_1 = 20$; Dashed-dotted black: $\lambda/\gamma_1 = 5$. Inset: Steady-state probabilities of the Laughlin state as a function of drive strengths.	126
7.4	(color online) Correlation functions $g_{ij} = c_{ij}/\Gamma_i\Gamma_j$ corresponding to the scenario in Fig. 7.2. Here $c_{ij}(t)$ is the joint probability of emitting a photon into mode i and a second into j , with a time delay of t ; Γ_i is the steady state probability of emitting into state i . The single-photon decay rate is γ_1 , and the drive strength is λ	127
7.5	(color online) Steady state probabilities of the Laughlin state as a function of the detuning. Each of the three modes are driven with equal amplitude: $\lambda_3 = \lambda_6 = \lambda_9 = \lambda$. The $m = 6$ detuning, ω_6 is displayed on the horizontal axis, and the $m = 3$ and $m = 9$ drives are chosen such that the four-body state $ A\rangle = -0.362 3339\rangle - 0.932 3366\rangle$ is on resonance. The energy level diagram corresponding to this drive condition is depicted in Fig. 7.6. For reference, the three highest peaks are labeled by different numbers of asterisks.	130
7.6	(color online) Rotating frame energy level diagrams corresponding to the three most prominent peak Fig. 7.5, marked by (*) top, (**) middle, and (***) bottom. As illustrated by horizontal arrows in the top panel, a single-photon drive sequentially couples states which differ by one polariton. The Laughlin state is populated by two-body loss from the four-body manifold (denoted by the double arrow).	133
7.7	(color online) Steady state probability of the Laughlin state for the three largest peaks in Fig. 7.5 as a function of drive amplitude $\lambda_3 = \lambda_6 = \lambda_9 = \lambda$ for $\gamma_2 = 0.001$	134
7.8	(color online) Steady state probability of being in the Laughlin state for detuning $\omega_6 = 0.1064$ (corresponding to the (*) state in Fig. 7.5 as a function of λ_3 and λ_6 drive amplitudes with $\lambda_9 = 0$. The peak probability is 83.7% at $\lambda_6 = 0.010$ and $\lambda_3 = 0.013$	136
7.9	(color online) Probability of the long-lived states $ \text{vac}\rangle$, $ 3\rangle$, $ 6\rangle$, $ 9\rangle$ and $ L\rangle$ as the system in the (*) configuration evolves according to the Lindblad master equation. The time scale for the finite evolution is in terms of the two-body loss rate. The evolution uses optimized parameters for this configuration. The system configuration at long times contains a non-zero population of the long-lived single-body states.	137

7.10	(color online) Probability of occupying the Laughlin state at time $t = 1000/\gamma_2$ for four-photon (top), two-photon (middle), and single-photon (bottom) drive. In all cases only the $m = 6$ mode is driven, and $\gamma_2 = 0.10$. The different curves correspond to various drive amplitudes λ relative to the two-body loss rate γ_2	146
7.11	Some useful tensor network identities.	147
7.12	Tensor network manipulation of the term $H(t)\rho$ to obtain the form $H(t) \otimes \mathbb{1}$	149
7.13	Manipulation of the term $\rho H(t)$ using tensor network algebra to obtain the equivalent form $\mathbb{1} \otimes H^T(t)$	150
7.14	Tensor network manipulations of the jump operator terms.	151
7.15	Energy level model of the Rydberg atom. The state without a label corresponds to the ground state of the atom. This is the state in which most of the background sea of atoms in the cavity reside. With selection rules, the atom is pumped into the lossy p -state, $ p\rangle$. A control beam, Ω , at detuning δ pumps the atom from the p -state into the highly-excited Rydberg state, $ s\rangle$. Interference between the p and s levels, induced by a cavity photon, allows for an electromagnetically induced transparency configuration, where a polariton with no p -state component exists (i.e. the ‘dark’ polariton).	152
7.16	Energies of the bright and dark polaritons versus detuning, given by Eqn. 7.35. Widths of the lines are related to the p -state component and thus the decay of the polariton states. The points where $\delta = 0$ correspond to the electromagnetically induced transparency (EIT) configuration, where both bright polaritons are off-resonant from the long-lived dark polariton.	153
7.17	Eigenenergies of the two-body Hamiltonian, Eqn. 7.53, as a function of the Rydberg dipole interaction potential.	154

CHAPTER 1

INTRODUCTION

Quantum computers are computational devices that harness quantum entanglement and the ability of quantum systems to be in a superposition of many states [14]. At the time of writing this thesis, it is predicted that quantum computers can be much faster than classical computation for solving certain problems, and experimentally proving this quantum advantage is an active research area [15, 16].

Building a quantum computer requires a scalable physical system with a well-characterized qubit. One needs long qubit coherence times, the ability to initialize the qubits to a starting state, and a universal set of quantum gates. One also needs to be able to carry out certain measurements. These criteria are known as the DiVincenzo criteria [17]. Several quantum computing systems possessing all of these criteria have been proposed to date: the quantum circuit model, the quantum Turing model, the adiabatic quantum model, the one-way quantum computing, and cellular automata [18–24]. Each of these proposed models for a quantum computer have their own unique hardware designs. Part of this thesis focuses on a small part of this broad field of quantum computing: the quantum circuit model with superconducting qubits and dissipative state preparation in atomic systems.

A natural way to represent gates in the quantum circuit model is the stabilizer formalism [25], see Sec. 2. The stabilizer formalism breaks down the quantum computational space into a region which is efficiently simulatable through classical computation and a region where it is not: this concept is also known as the Gottesman-Knill theorem [25].

In the region where classical computation is inefficient, one could demonstrate

a quantum advantage. Within this formalism, universality requires special gates, which arise from states outside of the efficiently-computable regime. Such states are error-prone, and one way to obtain the desired fidelity is through quantum magic state distillation, see Sec. 2.1. Magic state distillation protocols are resource intensive. This thesis elaborates on an alternative method for obtaining a protected universal computation by introducing an algorithm for mapping between any error-protecting frameworks whose complimentary gate sets form a universal set, see Sec. 3.

To date, all quantum resource states and gates are faulty to some percentile. Determining error thresholds for fault-tolerant computations is a complicated process dependent on the computing architecture and quantum error protecting procedures at hand, but generally such thresholds are low. This requires development of robust hardware that provides protection against unwanted decoherence. Several proposed protected qubit designs are promising, [9, 26–32] and require advanced techniques to model, see Sec. 5.

The method of topological quantum computation lies within the circuit model, where defects are braided to perform a universal set of gates on topologically-protected encoded qubits. Candidate defects with non-Abelian braiding statistics are found in fractional quantum Hall states [33]. Creating these states is challenging. One novel construction is to use dissipation. By design, the system would protect the state against decoherence by using decoherence to stabilize itself. We use this design and propose an AMO experiment to prepare a two-body fractional quantum Hall state, a step towards universal computation using these states, see Sec. 6.

This thesis covers a wide variety of topics. In Ch. 2, we introduce some

background information on magic distillation protocols. In Ch. 3, we present our original algorithm for converting any one stabilizer code into another. In Ch. 4, we derive some bounds on the thresholds for quantum decoders (some of which are found in the literature) to show that the addition of a type of error reduces the possible decoding thresholds despite having knowledge that the error exists. We also show how some of the thresholds relate to the estimated phase boundaries from statistical mechanical models. In Ch. 5, we discuss some background information about superconducting circuit design, and then we discuss our software tool and use it to find the lowest-lying energy spectrum of superinductor. In Ch. 6, we briefly introduce dissipation in open quantum systems. In Ch. 7, we discuss our proposal for dissipatively preparing the Laughlin state of Rydberg polaritons in a twisted cavity.

CHAPTER 2
UNIVERSAL QUANTUM COMPUTATION IN THE CIRCUIT
MODEL

Quantum information stored and manipulated in a quantum computer is sensitive to errors due to entanglement with the environment. Such entanglement can occur with stray magnetic fields, electric fields, photons from black body radiation, unstable control signals, etc. causing decoherence of the quantum information. One method to protect against decoherence is by encoding quantum information into a code, namely, stabilizer codes.

Stabilizer codes were first introduced by Peter Shor in 1995. He showed that classical error-correcting codes, which were used in computers in the years before classical hardware was reliable, had a quantum analog formed through their representation by the Pauli group [14]. Since Shor's discovery, a large number of examples of stabilizer codes have been constructed. Most of them belong to the class of additive codes, where the codewords form a common eigenspace of commuting stabilizer operators, each of which is a product of Pauli matrices acting on different qubits.

Stabilizer codes are a type of error-correcting code that encodes k logical bits of quantum information in n qubits. A stabilizer is an operator composed of a tensor product of Pauli operators $\sigma_x, \sigma_y, \sigma_z$ acting on the physical qubits. The code is generated by a collection of $n - k$ stabilizers which commute with one another, forming an Abelian group. The common eigenspace of these operators forms the codespace, in which the logical qubits reside. Allowed operations on the logical qubits are composed of operators which commute with all of the stabilizers defining the codespace, and anticommute with one another. Thus, there are only two things

for specifying a stabilizer code - a generating group of commuting stabilizers and a logical operation.

States and state evolution described in the stabilizer formalism can be tracked conveniently using the Heisenberg description of the quantum computer – this is also called the product-operator formalism [25]. A state in the stabilizer formalism is described by the set of Pauli operators of which it is an eigenstate, and quantum state evolution is tracked by elements in the Clifford group, a group of tensor products of Pauli matrices acting on the state, also see Sec. 3. States described in the stabilizer formalism and the evolution of such states can be simulated efficiently (i.e. in polynomial time) by a classical computer (see the Gottesman-Knill theorem [25]).

To see the convenience of the product-operator formalism, assume the state $|\psi\rangle$ is an eigenstate of operators N in a stabilizer code such that $N|\psi\rangle = |\psi\rangle$. One can track state evolution by a unitary operator U ,

$$UN|\psi\rangle \rightarrow UNU^\dagger U|\psi\rangle \quad (2.1)$$

so that now the operator UNU^\dagger is the new stabilizer describing the evolved state $U|\psi\rangle$. So, instead of tracking the evolution of a quantum state having d^n terms (where d is the dimension of the Hilbert space and n is the number of physical particles), one can simply keep track of the evolution operators whose product determines U .

The product operator formalism not only simplifies quantum evolution but it also simplifies the quantum noise models. A quantum system can be subject to a continuous spectrum of noise. Noise could cause qubits to be lost or erased, or noise could be modeled as any unitary acting on the physical qubits. In the stabilizer code picture, stabilizer measurements project the quantum system into

an eigenstate of all of the commuting stabilizers – a continuous spectrum of quantum noise is thus projected into the Pauli group. (For example, a qubit that is lost can be modeled by one that is lost and then prepared in the fully mixed state. A stabilizer measurement would then project the qubit into an eigenstate of an element of the Pauli group.) The simplification of the error model allows errors on qubits to be represented by an error channel composed of Pauli rotations occurring probabilistically. Such an example is the depolarizing model, $\epsilon(\rho) = (1 - p_i)\rho + p_x X\rho X + p_y Y\rho Y + p_z Z\rho Z$ where the probabilities p_x, p_y, p_z and $p_i = 1 - p_x - p_y - p_z$.

The stabilizer measurements are similar to parity checks on a classical computer – they provide information about errors on the encoded quantum information without directly measuring the state. The collection of eigenvalues obtained from measurements of the stabilizers is called a syndrome. Syndromes are accumulated throughout a computation. They are stored on a classical computer and interpreted by a decoding algorithm which decodes the errors into a correction to be applied at the end of the quantum computation.

One could argue the point of doing quantum computation in the stabilizer formalism at all, since the evolution and errors can be all be tracked efficiently using classical hardware. A quantum computation in the stabilizer formalism can be efficiently simulated on a classical computer, that is, until special gates, called non-Clifford gates are introduced, see Sec. 3. It is the addition of this type of gate which allows a quantum computer to have an advantage over a classical one – a quantum computer can calculate efficiently in a region that is inefficient for a classical computer.

One protocol for implementing these special gates is through magic state dis-

tillation, where a special state is obtained through a purification procedure within a stabilizer code and injected into the circuit to perform a non-Clifford gate, see Sec. 2.1. Distillation procedures are resource-intensive [1, 2, 34, 35], usually taking 90% of the resources in a quantum calculation (many layers of the quantum computer can be dedicated to non-Clifford gate factories [36, 37]). Another protocol for achieving universality is through stabilizer code mapping, where a universal set of gates on encoded quantum information is achieved through complementary gate sets naturally occurring in each stabilizer code. Previously, it was only known that a small subset of codes could be mapped between through a method called gauge-fixing [38–41]. The work included in this thesis shows that any stabilizer code can be mapped to any other stabilizer code, with no contingencies, and it presents an algorithm for doing so without requiring any insight, see Sec. 3.

2.1 Magic State Distillation

Clifford group operations and Pauli group measurements do not provide a universal set of quantum gates. One can verify this by performing Pauli rotations on a quantum state in the Bloch representation, i.e.,

$$\hat{\rho} = \frac{1}{2}(\mathbb{1} + \rho_x\sigma_x + \rho_y\sigma_y + \rho_z\sigma_z) \quad (2.2)$$

where the sum over the $\rho_i\sigma_i$ terms is a statistical combination of Pauli operators. As described below, statistical mixtures of Pauli rotations of Pauli operator eigenstates only allows one to reach a subspace of the Bloch sphere, namely the octahedron defined by,

$$|\rho_x| + |\rho_y| + |\rho_z| \leq 1. \quad (2.3)$$

The vertices of the octahedron correspond to the eigenstates of the Pauli matrices, and the surface and interior of the octahedron correspond to all states that are

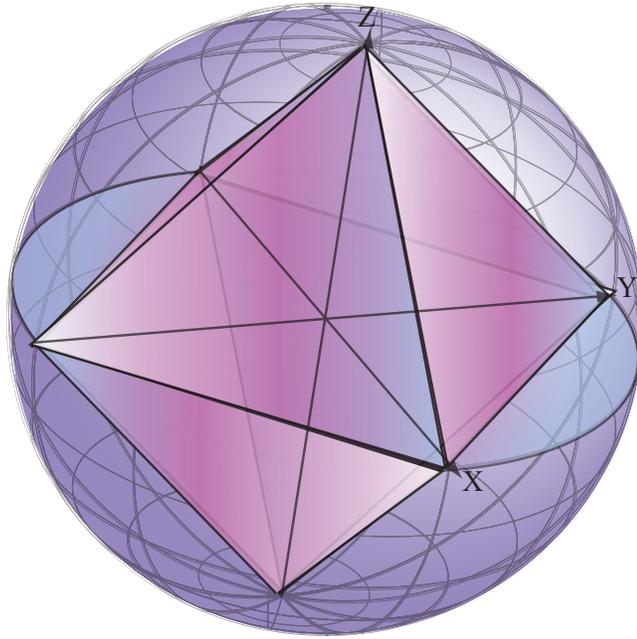


Figure 2.1: Bloch sphere and inscribed octahedron formed by the condition in Eqn. 2.3.

statistical combinations (ensembles) of those eigenstates.

For example, start with an eigenstate of the σ_x , which is a vertex on the octahedron and lies on the surface of the Bloch sphere, $\hat{\rho}_x = \frac{1}{2}(\mathbb{1} + \sigma_x)$. Rotations of this state by single-qubit Clifford operations (i.e. $\mathbb{1}, \sigma_x, \sigma_y, \sigma_z$) are symmetries of the octahedron taking the eigenstate which is a vertex on the octahedron to another vertex on the octahedron. Thus, one cannot rotate $\hat{\rho}_x$ to points lying on the Bloch sphere outside of the inscribed octahedron.

Taking statistical combinations of eigenstates of σ_x, σ_y , and σ_z produces states within or on the octahedron. For example, taking a statistical combination of the eigenstate of σ_x and σ_y say, $\hat{\rho} = \frac{1}{2}\hat{\rho}_x + \frac{1}{2}\hat{\rho}_y$, gives the state $\hat{\rho} = \frac{1}{2}(\mathbb{1} + \frac{1}{2}\sigma_x + \frac{1}{2}\sigma_y)$ - a point on the surface of the octahedron with a component of $\frac{1}{2}$ on the σ_x axis and $\frac{1}{2}$ on the σ_y axis of the Bloch sphere.

On the other hand, if we rotate the eigenstate $\hat{\rho}_x$ by a $\frac{\pi}{4}$ rotation about the σ_z axis, i.e. $R(\frac{\pi}{4}) = \begin{bmatrix} 1 & 0 \\ 0 & e^{i\frac{\pi}{4}} \end{bmatrix}$, a non-Clifford gate, we get the rotated state, $\hat{\rho}'_x = R(\frac{\pi}{4})\hat{\rho}_xR(\frac{\pi}{4})^\dagger = \frac{1}{2} \left(\mathbb{1} + \frac{1}{\sqrt{2}}(\sigma_x + \sigma_y) \right)$. The rotated state $\hat{\rho}'_x$ has a component of $\frac{1}{\sqrt{2}}$ on both the σ_x and σ_y axes of the Bloch sphere, so the rotated state lies on the Bloch sphere, does not satisfy the Eqn. 2.3, and thus is not a Clifford state.

For a series of quantum rotations to cover the entire Bloch sphere, thus enabling universal quantum computation, another gate outside of the Clifford group is needed. Such a gate could be a Toffoli gate, see Appendix 3.5.6, a $\frac{\pi}{8}$ rotation of the Bloch sphere, or the square root of the controlled-NOT gate, among other examples. The Clifford group plus an appropriate extra gate generate a set of unitary operators dense in $U(2^n)$ and therefore form a universal set of gates.

Knill's theorem implies that quantum computation is only more powerful than classical computation when it uses gates outside the Clifford group. However, networks using only Clifford group gates also have a number of important applications in the area of quantum communications. Quantum error-correcting codes are an important example. Other important communication problems, such as quantum teleportation, also use only Clifford group gates and measurements. Finally, networks consisting of Clifford group gates and measurements may provide useful subroutines to more complex quantum computations; an efficient method of analyzing and searching for such subroutines could prove very useful [25].

Magic states are non-stabilizer states, states which lie outside of the octahedron, and are used as a consumable resource to indirectly implement a non-Clifford operation. This can be accomplished using a technique called state injection, see Sec. 2.1.1. Magic states are often assumed to be faulty resource states. A technique

called magic state distillation uses a stabilizer code to convert a large number of resource states into a smaller number of magic states having higher fidelity.

Magic state distillation is often described in an abstracted manner, independent of the particular quantum computing architecture. In magic state distillation literature, Clifford operations are often considered ideal. This means that Clifford gates performed on physical qubits contribute no errors, and the only errors that need to be corrected are from the magic states to be distilled. Other operations that are considered to be ideal are preparation of the state $|0\rangle$, and measurements of a Pauli operator on any qubit. The preparation of a state, ρ , however, is faulty.

There are at least two methods for magic state distillation. The first method is to treat a collection of faulty magic states as if they were encoded and decode them, discarding the reduced state if an error has occurred [2], [34]. This allows for a higher threshold probability, although it is at the expense of computing resources. Another method is to encode faulty magic states into a code using Clifford operations, and perform a parity check measurement on the encoded magic state using non-Clifford gates created using state injection of additional copies of magic states [1]. In this method, the code would catch errors not only on the faulty magic states, but also from the errors in the non-Clifford gates used for the parity check measurement.

A great simplification for engineering a magic state distillation circuit results from the use of twirling. Twirling is the probabilistic application of a (Clifford) gate to a faulty resource state. Following the example in [1], on an initial state ρ_0 , the channel,

$$\rho = \frac{1}{2}\rho_0 + \frac{1}{2}H\rho_0H^\dagger \quad (2.4)$$

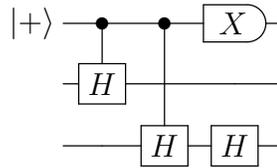
applies the Hadamard gate or the identity both with equal probability. Multiple

iterations of this gives a state that is in the eigenspace of H ,

$$\rho = \lambda_1 |H\rangle \langle H| + \lambda_2 |-H\rangle \langle -H| \quad (2.5)$$

where λ_1 and λ_2 are the eigenvalues of the twirled state ρ . Such a state is used as a faulty resource magic state which can only suffer from Y errors with some probability p , since $Y |H\rangle = -|H\rangle$.

One way to distill magic states from faulty resource states is to perform a parity check measurement. For example, consider the following circuit from [1],



which measures the logical parity operator $\bar{H}_1 \bar{H}_2$. Assuming the input states are twirled and the gates are perfect, this measurement can determine if there is one Y -error.

The gates, in fact are non-Clifford, so they are not perfect. Blindly implementing such a gate could break the fault-tolerance of the circuit. Such non-Clifford gates are rather constructed using state-injection, see Fig. 2.1.

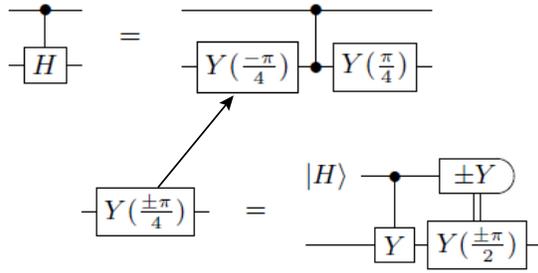


Figure 2.2: Injection circuit for a controlled-H non-Clifford gate, modified from [1]. In the injection circuit, if a Y-error occurs during state preparation of $|H\rangle$, it propagates to a Y gate on the output. When implementing the controlled-H gate if: (1) a Y error from the first $Y(-\frac{\pi}{4})$ gate occurs, it propagates to a Z-gate on the control and a Y-gate on the target (2) a Y-error on the second $Y(\frac{\pi}{4})$ gate occurs, it translates to a Y-gate after the gate (3) two Y-errors occur, then there is a Z-error on the control.

Not only do we need to check the input states for errors, but we also need to check the non-Clifford gates for errors. In order to do this, the entire parity check circuit in Eqn. 2.1, is embedded into an error-correcting code. Because the error in performing the controlled-H gate using state injection, the error-correcting code must detect and correct both Z and Y errors.

The error-correcting code choice in Meier, et al. [1] is the four-qubit code, which is just a sequence of three gates composed entirely of CNOTs. After enumerating the error patterns of Y and Z error and classifying them by whether or not they are detected and where or not they causes a logical error, one can determine the probability that an undetected error pattern will cause a logical error. In distillation routines, the errors are considered to be equiprobable and independent. This is not necessarily an experimental reality, but it greatly simplifies the counting to provide an error probability. Error probabilities are expressed as polynomials in p , the probability of a single qubit error. If an error involving n qubits can

occur C ways, it contributes Cp^n . For example, if there are nine ways two errors are not detected and cause a logical error, then its contribution to the polynomial is $9p^2$. Conversely, error patterns that have no effect reduce the probability of a logical error. For example, if there are fifty-six ways that three errors operate as stabilizers of the code, then the term in the polynomial is $-56p^3$. Error patterns which can be detected do not contribute terms.

The lowest degree of the error polynomial gives some measure of the error correcting properties of the code. In [1], the lowest degree is 2, and so the code provides quadratic suppression of errors – any single error is detected by the code. First, this means that having the code is better than not. Second, it is advantageous for protecting against higher-order error patterns that reduce to a single error. The second-order error suppression is understood by propagating errors on non-Clifford gates and reducing them to Pauli errors plus a single qubit Y-error. A Y-error projects the state out of the codespace and so it is detected by the code, see Fig. 2.1. Also, by design, a Y-error on the resource state is detected by the parity measurement.

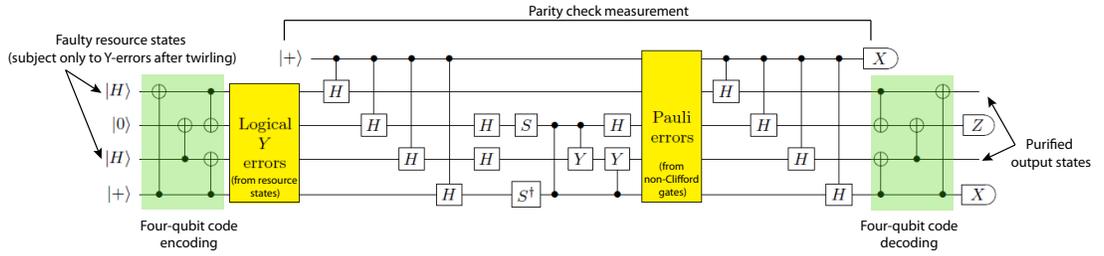


Figure 2.3: Magic state distillation circuit modified from [1]. There are two $|H\rangle$ resource states which can have only Y-error on them after twirling. The effective circuit is a parity check circuit which can detect one Y-error on either of the two resource states. Because the parity check circuit requires non-Clifford gates, these gates are faulty, but their faults are made in a predictable way using state-injection, see Fig. 2.1. To catch the errors from the non-Clifford gates, the parity check circuit is embedded into an error-correcting code, in this case, the four-qubit code. Thus the resource $|H\rangle$ states become logical states after the encoding, and the operators in the parity check measurement are logical (i.e. the circuit in the middle performs the parity check in Fig. 2.1). Pauli error patterns produced from the non-Clifford gates are pushed through the circuit, using circuit identities, to the section in yellow. Many of these error patterns (up to order 8, eight errors at once, in all possible ways) are generated and categorized into two categories (either logical error or not) to determine the error polynomial.

2.1.1 State Injection

Bravyi and Kitaev [2], introduce a protocol for consuming a magic resource state to perform an arbitrary phase gate rotation,

$$\Lambda(e^{i\theta}) = \begin{pmatrix} 1 & 0 \\ 0 & e^{i\theta} \end{pmatrix}$$

by the injection of the magic state $|A_\theta\rangle = \frac{1}{\sqrt{2}}(|0\rangle + e^{i\theta}|1\rangle)$. This process is fully described in [2], but we outline it here in a way to compare it with another injection method.

Our goal is to perform an arbitrary phase rotation on an arbitrary state $|\psi\rangle = a|0\rangle + b|1\rangle$. First, form the product $|\psi_0\rangle = |\psi\rangle \otimes |A_\theta\rangle = \frac{1}{\sqrt{2}}(a|00\rangle + be^{i\theta}|11\rangle) + \frac{1}{\sqrt{2}}(a|01\rangle + be^{i\theta}|10\rangle) = |\psi_0^+\rangle + |\psi_0^-\rangle$, where $|\psi_0^+\rangle$ corresponds to the +1 eigenstate of $\sigma^z \otimes \sigma^z$ and $|\psi_0^-\rangle$ corresponds to the -1 eigenstate. Then, the second step in the process is to measure $S = \sigma^z \otimes \sigma^z$. This is equivalent to entangling the state with an ancilla and applying the projector $P_\pm = \frac{1}{2}(\mathbb{1} \pm \sigma^z \otimes \sigma^z)$ depending on the state of the ancilla: $\frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) \otimes |\psi_0\rangle \rightarrow \frac{1}{\sqrt{2}}(|0\rangle P_+ |\psi_0\rangle + |1\rangle P_- |\psi_0\rangle)$. If the ancilla bit $|0\rangle$ is measured, then the state of the data qubits is in the +1 eigenspace of S (i.e. $|\psi_0\rangle \rightarrow |\psi_0^+\rangle$). Conversely, if the ancilla bit $|1\rangle$ is measured, then the state of the data qubits is in the -1 eigenspace of S (i.e. $|\psi_0\rangle \rightarrow |\psi_0^-\rangle$). Finally, applying an XOR between the data qubits implements the phase gate. In the case of $|\psi_0^+\rangle$ we have: $|\psi_0^+\rangle \rightarrow \frac{1}{\sqrt{2}}(a|00\rangle + be^{i\theta}|10\rangle) \rightarrow \frac{1}{\sqrt{2}}(a|0\rangle + be^{i\theta}|1\rangle)|0\rangle$, and the procedure has applied the gate $\Lambda(e^{i\theta})$. In the case of $|\psi_0^-\rangle$ we have: $|\psi_0^-\rangle \rightarrow \frac{1}{\sqrt{2}}(ae^{i\theta}|01\rangle + b|11\rangle) \rightarrow \frac{1}{\sqrt{2}}e^{i\theta}(a|0\rangle + be^{-i\theta}|1\rangle)|1\rangle$, and the gate $\Lambda(e^{-i\theta})$ is applied. The procedure above is equivalent to the circuit in Fig. 2.4, where the eigenvalues of the Z -measurement on the first and third qubits have the relation, $\pm 1 \rightarrow \Lambda(e^{\pm i\theta})$. (i.e. if +1 eigenvalues are measured, then the gate $\Lambda(e^{\pm i\theta})$ was

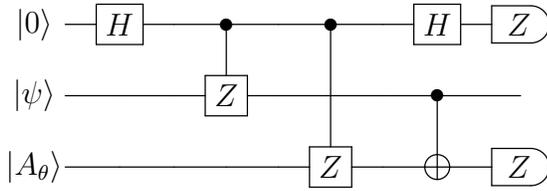
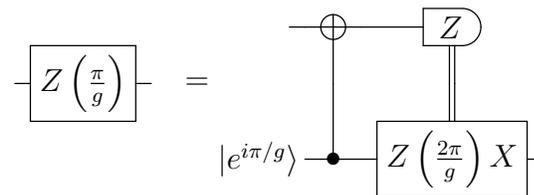


Figure 2.4: Equivalent circuit diagram to the procedure described in [2] for implementing the arbitrary phase gate $\Lambda(e^{\pm i\theta})$ by injecting the magic state $|A_\theta\rangle$.

applied. Conversely, if the -1 eigenvalues are measured, then the gate $\Lambda(e^{-i\theta})$ was applied.)

The downside of this particular procedure is the probabilistic phase shift in the gate applied. Although implementing the phase $-\theta$ instead of θ can be corrected and would probably contribute to an overall global phase in most cases, it could still in practice require some dynamic gate compensation, thus complicating the computing hardware.

Another way to inject a magic state into a stabilizer code to perform a gate is described in [42]. Equipped with a magic state, one can inject the state to perform a non-Clifford gate. Take the following circuit:



This circuit injects (and thus consumes) the resource state $|e^{i\pi/g}\rangle$ to implement a rotation of $\frac{\pi}{g}$ about the Z-axis.

CHAPTER 3

STABILIZER CONVERSION

We present an algorithm for manipulating quantum information via a sequence of projective measurements. We frame this manipulation in the language of stabilizer codes: a quantum computation approach in which errors are prevented and corrected in part by repeatedly measuring redundant degrees of freedom. We show how to construct a set of projective measurements which will map between two arbitrary stabilizer codes. We show that this process preserves all quantum information. It can be used to implement Clifford gates, braid extrinsic defects, or move between codes in which different operations are natural. This chapter was adopted from [43].

3.1 Introduction

Although there is broad agreement that quantum mechanics can provide an important resource for computation, the community continues to search for the best way to exploit that resource. Here we describe a tool which can be incorporated into a number of quantum information processing architectures, and we show how it can be applied to solve important problems such as giving access to a universal set of transversal gates and braiding non-Abelian anyons.

Our approach is framed in the language of quantum stabilizer codes. These are approaches to quantum error correction in which k logical qubits are stored in n physical qubits (quantum spins or other two level systems): the remaining degrees of freedom are restricted by requiring that the allowed wavefunctions (those in the codespace) are eigenstates of $n - k$ given stabilizer operators. These stabilizer

operators define the code.

Given two arbitrary stabilizer codes, we show how to construct a sequence of measurements which map the codespace of one code into the codespace of the other while preserving the quantum information. Under appropriate circumstances this mapping is fault-tolerant. The utility of such rewiring has been recognized by the community, and our arguments build on ideas of “code deformation” [44,45] (which involves small rewirings of topological codes) and “code conversion” [46,47]. This latter strategy has also been termed “code switching” [48,49], and typically is based on a sequence of unitary gates. Our contribution is the construction of a general algorithm for finding a sequence of projective measurements which allow arbitrarily large deformations. As with the previously studied cases, rewiring can be used to implement quantum gates: a cyclic rewiring generically acts as a rotation in the codespace. We explicitly construct the unitary matrix that corresponds to the gate.

One motivator for mapping between codes is the realization that the amount of resources needed to perform different gates depends on the code. Recently, Paetznick and Reichardt [50] noted that this disparity can be taken advantage of by mapping between codes. Specifically, a mapping can thereby produce a universal set of gates which have a particularly simple structure, described as “transversal”. Transversal operations are naturally fault-tolerant, but there is no single code that admits a universal set of transversal gates [51]. To overcome this, Anderson, et. al. [39] proposed a quantum circuit which maps between the 7-qubit Steane code and the 15-qubit Reed-Muller code. The former admits transversal Clifford gates, while the latter admits transversal T and control-control-Z gates. Subsequently, several authors proposed other circuits for this mapping [38,41]. Our algorithm provides

a systematic approach to this, and related, problems. We illustrate its utility by producing a mapping between the Steane and Reed-Muller codes. The resulting circuit is particularly simple, involving only measuring stabilizer generators of the other code, and is fault-tolerant.

Our algorithm fits in with a long tradition of using measurement to manipulate quantum information. Measurement is a key piece of any quantum circuit, and there are well known computing architectures which consist primarily of repeated measurements [22, 24, 52–59]. These algorithms are often discussed within the stabilizer formalism [59] and can be interpreted as rewiring stabilizer codes. In this context, we emphasize that our innovation is not the basic idea that one can use projective measurements to map between codes, but rather the explicit construction of the sequence of measurements.

Beyond its application to quantum computing, our algorithm can be used to enable the study of novel collective effects of interest to condensed matter physics. A physical realization of a quantum stabilizer code can be considered a Hamiltonian system where the Hamiltonian is simply the projector onto the codespace. Kitaev has argued that this mapping allows the observation of anyons (excitations which behave as particles with statistics that are neither fermionic nor bosonic [60]). Using our code rewiring algorithm, we show how to braid non-Abelian twist defects, allowing the first direct observation of non-Abelian quasi-particle statistics. Such “quantum emulation” experiments would be highly impactful.

3.2 Algorithm

3.2.1 Stabilizer Codes

A stabilizer code stores k logical qubits in n physical qubits by specifying that a physical state is an eigenstate of $n - k$ given stabilizer operators (generators) with eigenvalue 1. The space of physical states is denoted the codespace. The stabilizer generators ($g_0, g_1, \dots, g_{n-k-1} \in G$) are typically taken to belong to the Pauli group – meaning that they are products of Pauli operators (I, X, Y, Z) acting on the physical qubits. They therefore all have eigenvalues ± 1 . The generators should be independent, and the space generated by their products is denoted S . For the codespace to be non-trivial, the stabilizers (and hence the generators) must commute with one-another. The generators for a stabilizer code are not unique: the same set S is generated if $g_i g_j$ replaces g_i , for any j .

In addition to the stabilizers, one can define logical operators, which map the codespace onto itself. The space of logical operators is generated by $2k$ members of the Pauli group, which can be labeled $\bar{X}_1, \dots, \bar{X}_k, \bar{Z}_1, \dots, \bar{Z}_k$, with \bar{X}_j anti-commuting with \bar{Z}_j , but commuting with all other logical generators. The labeling of the logical operators is not unique. Any member of the Pauli group that commutes with the stabilizers, but is not itself a stabilizer, is a logical operator.

3.2.2 Moving in the space of stabilizer codes

Consider two stabilizer codes S, S' that differ by only one anti-commuting generator: S is generated by $G = \{g_0, g_1, \dots, g_{n-k-1}\}$, while S' is generated by $G' = \{g'_0, g_1, g_2, \dots, g_{n-k-1}\}$, with anti-commutator $\{g_0, g'_0\} = 0$. As is readily verified by its action on the generators, the unitary operator $U = (1 + g'_0 g_0)/\sqrt{2}$

maps the codespace of S to the codespace of S' . One can further see that when acting on a state $|\psi\rangle$ in the codespace of S ,

$$U|\psi\rangle = \sqrt{2}P_1|\psi\rangle = \sqrt{2}g_0P_{-1}|\psi\rangle, \quad (3.1)$$

where $P_{\pm 1} = (1 \pm g'_0)/2$ are projectors into the space spanned by the ± 1 eigenstates of g'_0 . This relationship suggests an algorithm. Starting from a state $|\psi\rangle$, one measures g'_0 . If the result of the measurement is -1 , one applies g_0 to the new state, otherwise one leaves it alone. Due to Eq. (3.1), this procedure is completely equivalent to the unitary transform, but is often simpler to implement.

One can chain these operations together – one-by-one changing the stabilizer generators. In the next section we show how to construct a sequence of measurements that map between any two given stabilizer codes. If the final code is the same as the initial code, then we thereby apply a gate (given by the product of the U 's in Eq. (3.1)). This feature is used in a number of algorithms, such as topological [44, 45], and teleportation [52] based computing schemes.

The operator U in Eq. (3.1) maps the Pauli group onto itself, and is therefore described as a Clifford operator. Clifford operators are insufficient for universal quantum computation, and a quantum circuit only involving Clifford operators can be efficiently simulated on a classical computer [61]. Relaxing the constraint that the generators belong to the Pauli group opens up the ability to generate arbitrary gates.

Although the arguments have largely appeared elsewhere, in 3.5.1 we give the proofs that U has all of these properties. Note, if one has a non-measurement way to apply U , such gates could also be used as part of the rewiring.

3.2.3 Constructing a sequence of stabilizer codes

Given two stabilizer codes S, S' , we wish to construct a sequence of stabilizer codes S_0, S_1, \dots, S_N , such that $S_0 = S$, $S_N = S'$, and S_j differs from S_{j-1} by only one anti-commuting generator. By the arguments in Sec. 3.2.2 one then can map between S and S' by performing N measurements.

To facilitate constructing this sequence of stabilizer codes, we first make use of the fact that the generators are not unique, and find a set of generators of S and S' such that each of the generators fall into one of three blocks, denoted G_A, G_B, G_C and G'_A, G'_B, G'_C . The first blocks are identical: $G_A = G'_A$. The second blocks, G_B and G'_B contain the same number of elements, and have the property that the generators in G_B commute with members of S' , but are not in S' . Conversely, those in G'_B commute with the members of S , but are not in S . Thus the elements of G_B are logical operators for the code defined by S' , and vice versa. The elements in the third block are in one-to-one correspondence, and for each $g_j \in G_C$, there exists a single element $g'_j \in G'_C$, such that g_j and g'_j anticommute. The element g_j however, commutes with all other elements of G' (and likewise for g'_j). We claim that one can always find generators satisfying these conditions, and in Sec. 3.2.4 we provide a constructive algorithm for finding these generators.

For each element $g_j \in G_B$ our construction also gives a complementary operator $g_j^{(c)}$ with the property that $g_j^{(c)}$ anticommutes with g_j , but commutes with all other elements of G . Furthermore $g_j^{(c)}$ commutes with all elements of G' . Similarly, for each $g'_j \in G'_B$ we find a complementary operator $g_j'^{(c)}$ with analogous properties.

Given this construction, we can then generate the sequence of stabilizer codes $S_0, S_1, S_2, \dots, S_N$. If the number of generators in blocks G_A, G_B, G_C are a, b, c , then

we require $N = 2b + c$ steps. It will require one step to replace an element of G_C with one of G'_C , and two steps to replace an element of G_B with one of G'_B . The required operation for replacing elements of G_C is simple: one just measures the elements of G'_C . In order to replace $g_j \in G_B$ with $g'_j \in G'_B$, one first measures the product $g_j^{(c)} g_j'^{(c)}$, then measures g'_j . Each of these steps is of the form detailed in Sec. 3.2.2.

Note if G_B is empty, then the two codes S and S' are different gauge fixings of a single subsystem code. In that case, our algorithm reduces to a standard gauge fixing approach. See Sec. 3.2.5.

In 3.5.3 we give some simple examples to illustrate the mechanics of this procedure. Section 3.3 further explores the utility of these mappings.

3.2.4 Construction of the generators and complementary operators

In this section we explicitly construct generators of the form described in Sec. 3.2.3. Given an arbitrary set of generators, G and G' of S and S' , we first construct a connectivity matrix M , whose (i, j) 'th element is 1 if the i 'th member of G anti-commutes with the j 'th member of G' . Otherwise that element is zero. For codes with n physical qubits and k logical qubits, M will be a $(n - k) \times (n - k)$ matrix. As already emphasized, the generators are not unique, and the same set of stabilizers are formed if one replaces any one generator by its product with another. Such a replacement in G corresponds to adding the j -th row of M to the i -th row mod(2). A similar replacement in G' corresponds to adding columns mod(2). By using the same techniques used for row reduction, one can thereby find a set of generators

for which M is diagonal (with zeros and ones along the diagonal). The generators g_i and g'_i with $M_{ii} = 1$ correspond to block C , as defined in Sec. 3.2.2.

Consider any $g_j \notin G_C$. Because of the structure of M , it commutes with all of the generators of the stabilizer group of S' . Thus it is either a member of S' or logical operator of S' . If it belongs to S' , then it equals a product of the members of G' – and one can always construct a set of generators for S' such that $g_j \in G'$, giving us blocks G_A and G'_A . This rearrangement of G' does not require replacing any of the elements of block C . Finally, all of the remaining generators are logical operators for the other code, and hence are in blocks G_B and G'_B .

We further transform the generators, based upon the operators in blocks G_B and G'_B . Each $g_j \in G_B$ is a logical operator for S' , and hence there is a complementary logical operator $g_j^{(c)}$, that anti-commutes with g_j . By construction, $g_j^{(c)}$ commutes with all elements of G' (and hence the elements of G_A). The complementary operators can always be chosen to commute with each other: If $\{g_j^{(c)}, g_k^{(c)}\} = 0$ for some $g_k \in G_B$, one simply takes $g_j^{(c)} \rightarrow g_j^{(c)} g_k$. After this manipulation, one can further transform the generators so that $g_j^{(c)}$ commutes with all if $g_k \in G_B$ or G_C , for $k \neq j$: If $g_k \in G_C$ anti-commutes with $g_j^{(c)}$, we take $g_k \rightarrow g_k g_j$. This transformation does not change any of the other commutation relations. Note, as illustrated in Sec. 3.3, one can map between codes with different numbers of qubits by simply appending auxiliary qubits to the shorter code with trivial local stabilizers.

3.2.5 Distance Bounds

In commonly-studied noise models, noise acts independently on physical qubits. Consequently, one measure of the robustness of the code is its *distance* – the lowest weight operator that performs a logical X- or Z- operation. Here *weight* is the number of qubits which are acted on by the operator. A code of distance d can correct errors of weight t or less, where $d = 2t + 1$.

Our basic algorithm can produce intermediate codes whose distance is smaller than that of S and S' . 3.5.4 gives an explicit example. In Secs. 3.2.5 and 3.2.5 we quantify this issue by deriving lower bounds on the distances of the intervening codes generated by our algorithm when mapping between S and S' . As recently shown by Huang et al. [62], these lower bounds are worst case scenarios, and after adding appropriate ancilla qubits, there always exists a path in which the intermediate codes have distance no smaller than S or S' . Moreover, Huang et al. find the remarkable result that a random path yields high distance codes with high probability (and this probability can be made arbitrarily large by adding more ancillary qubits).

Subsystem Codes

We first consider the case where G_B is empty, and one only needs to measure stabilizers belonging to the code to which we wish to map. In this case our algorithm may be interpreted as a gauge fixing procedure on a subsystem code [63–66]. Below we describe subsystem codes and outline this procedure. In particular we show that if G_B is empty, then all codes in the path between S and S' are gauge fixings of a single subsystem code. One consequence is that the distances of each

of these codes are bounded below by the distance of that subsystem code.

A subsystem code stores k logical qubits in n physical qubits. States in the codespace are eigenstates of s independent stabilizer operators (forming a group S and having generators G), where $s < n - k$. This leaves $r = n - k - s$ degrees of freedom which are not used to encode any information. These r degrees of freedom are called gauge qubits, and they can be freely manipulated. The gauge group T is composed of Pauli operators that commute with both the logical operators and the stabilizers of the subsystem code. This is the set of operators which cannot disturb the encoded information. The gauge group of a subsystem code is generated by $s + 2r$ elements – the generators of the stabilizer group and the generators of the “logical operators” that act on the gauge qubits. Gauge fixing amounts to creating a stabilizer code whose generators consist of r independent commuting elements of T/S along with the elements of G . Choosing those r elements in different ways can produce distinct stabilizer codes. Given a state stored in a subsystem code, its gauge-fixed variant is produced by simply measuring the relevant gauge operators.

Consider two stabilizer codes with stabilizer generators G and G' , decomposed into blocks A , B , and C , as in Sec. 3.2.3. Recall that the elements of the two A blocks are identical: $G_A = G'_A$, and the elements of the two C blocks are in one-to-one correspondence with $g_j \in G_C$ anti-commuting with one $g'_j \in G'_C$ and commuting with all other elements of G'_C . If the B block is empty, we can construct a subsystem code by taking its stabilizer group to be generated by $G_A = G'_A$. We use G_C and G'_C as the generators of the gauge group (along with G_A). The two stabilizer codes are created from this subsystem code via gauge fixing. Our algorithm then reduces to the standard gauge-fixing process, and each code along the path represents a different gauge fixing. A consequence is that if the

subsystem code has distance d , then we are guaranteed that each code visited during the conversion will have at least distance d . This distance, for example, can be calculated using the approach in [67].

When G_B and G'_B contain $b > 0$ elements, the most naive generalization of this procedure fails. Namely, consider a subsystem code with stabilizer group G_A and gauge group containing the union of G_B, G'_B, G_C , and G'_C . If b is the number of elements in G_B and c is the number of elements in G_C , this gauge group contains at least $2b + c$ linearly independent commuting elements – namely the set $G_B \cup G'_B \cup G'_C$. Thus it encodes at most $n - (a + 2b + c)$ logical qubits, which is smaller than the $n - (a + b + c)$ qubits encoded by the two original stabilizer codes.

Generic Case

Here we generalize the argument of Sec. 3.2.5 to produce a general bound on the distances of our codes. In particular, if the B-blocks of S and S' contain b elements each, then we construct 2^b subsystem codes. We argue that any stabilizer code generated by our procedure must be a gauge fixing of one of these. In the special case where $b = 0$, this procedure generates the subsystem code in Sec. 3.2.5. The distance of any code generated by our procedure is then bounded below by the smaller of the distances of all these subsystem codes. For modest b the task of enumerating these subsystem codes and finding their distances is reasonable.

For this construction we take the B-blocks to contain $G_B = \{Z'_1, \dots, Z'_b\}$ and $G'_B = \{Z_1, \dots, Z_b\}$ where Z'_j and Z_j are logical operators for S and S' . Note, this notation does not imply that these necessarily correspond to logical Z . The complementary logical operators are $\{X'_1, \dots, X'_b\}$ and $\{X_1, \dots, X_b\}$. We further define the set $\overline{G_B} = \{X_1 X'_1, X_2 X'_2, \dots, X_b X'_b\}$ and the 2^b sets G_B^s made by choosing

h_s elements from G_B and $b - h_s$ from G'_B . Here s labels which of these choices are made. We then construct the subsystem code with stabilizer group G_A , and gauge group generated by $G_A, G_B^s, \overline{G_B}, G_C, G'_C$. Any code traversed in our procedure will have its stabilizer generators in the gauge group of one of these codes, and hence is a gauge fixing of that code.

3.2.6 Fault Tolerance

Intuitively, a measurement of the form of those in Sec. 3.2.2, is fault-tolerant if errors that occur during the measurement do not degrade the error-protective properties of the code. For a code of distance 3 or higher (ie. one in which at least a single qubit error can be corrected), this intuitive requirement can be satisfied by the property that no single error during the measurement can propagate to more than one qubit on the data [61]. The cat state method, as introduced by Shor [3] and nicely discussed by Aliferis, Gottesman, and Preskill [4], provides one approach to meeting this requirement. The measurement is described in detail in 3.5.2.

3.2.7 Implementing Constraints

In practical applications, the physical apparatus may introduce constraints on the operators measured. For example, one may only be able to carry out single qubit measurements, or measurements on sets of qubits that are connected by hard-coded wires. In the presence of such constraints, it is no longer possible to map between any two arbitrary stabilizer codes. The mapping may also be asymmetric: the constraints may be satisfied in mapping from A to B, but not in mapping from

B to A. This latter setting is the domain of “one-way” quantum computing [52].

One defines the constraints through the set W of all measurable operators. Given stabilizers codes S and S' generated by G and G' , a necessary, but not sufficient, condition for the existence of a rewiring path is that G' is a subset of the group generated by $W \cup G$. For small systems, constrained paths can be found via an exhaustive search.

Note in many cases, measuring high-weight stabilizers is problematic. If one constrains the weights of the generators to be measured along the path from one code to another, such a path is not guaranteed to be generated by the algorithm.

3.3 Applications

In this section we illustrate the utility of our algorithm by using it in physically relevant cases: code conversion for universal transversal computing and braiding defects and twists in topological codes.

3.3.1 Universal transversal computing

The $[[7,1,1]]$ Steane code is a 7-qubit code that supports transversal Clifford gates, and the 15-qubit $[[15,1,3]]$ quantum Reed-Muller code supports several transversal non-Clifford gates which complement those of the Steane code. Thus [38, 39, 41] suggested sequentially mapping between these codes to produce a universal set of transversal gates. The generators for these codes are given in Table 3.1, where we have appended extra bits to the Steane code.

Table 3.1: Generators for Steane code ($g_0 - g_5$) and Reed-Muller codes ($g'_0 - g'_{13}$). Additional qubits with arbitrarily chosen stabilizer generators ($g_6 - g_{13}$) are appended to the Steane code for the conversion.

	Steane	Reed-Muller
j	g_j	g'_j
0	$X_1 X_3 X_5 X_7$	$X_1 X_3 X_5 X_7 X_9 X_{11} X_{13} X_{15}$
1	$X_2 X_3 X_6 X_7$	$X_2 X_3 X_6 X_7 X_{10} X_{11} X_{14} X_{15}$
2	$X_4 X_5 X_6 X_7$	$X_4 X_5 X_6 X_7 X_{12} X_{13} X_{14} X_{15}$
3	$Z_1 Z_3 Z_5 Z_7$	$X_8 X_9 X_{10} X_{11} X_{12} X_{13} X_{14} X_{15}$
4	$Z_2 Z_3 Z_6 Z_7$	$Z_1 Z_3 Z_5 Z_7 Z_9 Z_{11} Z_{13} Z_{15}$
5	$Z_4 Z_5 Z_6 Z_7$	$Z_2 Z_3 Z_6 Z_7 Z_{10} Z_{11} Z_{14} Z_{15}$
6	Z_8	$Z_4 Z_5 Z_6 Z_7 Z_{12} Z_{13} Z_{14} Z_{15}$
7	Z_9	$Z_8 Z_9 Z_{10} Z_{11} Z_{12} Z_{13} Z_{14} Z_{15}$
8	Z_{10}	$Z_1 Z_3 Z_9 Z_{11}$
9	Z_{11}	$Z_2 Z_3 Z_{10} Z_{11}$
10	Z_{12}	$Z_3 Z_7 Z_{11} Z_{15}$
11	Z_{13}	$Z_1 Z_3 Z_5 Z_7$
12	Z_{14}	$Z_2 Z_3 Z_6 Z_7$
13	Z_{15}	$Z_4 Z_5 Z_6 Z_7$

Without any reference to the structure of the codes, our algorithm gives a mapping between them (see 3.5.5 for details). For example, we find that 7 operators need to be measured to map from the Steane to the Reed-Muller code: $g'_0, g'_1, g'_2, g'_3, g'_8, g'_9, g'_{10}$. Remarkably, these are all stabilizer generators of the Reed-Muller code, and hence no additional hardware is required, beyond what one already needs for implementing error correction. These operators can be measured in any order desired, or even simultaneously. Similarly one can map back to the Steane code by only measuring stabilizers: $g_0, g_1, g_2, g_6, g_8, g_9, g_{10}$. The resulting round-trip acts as the identity operator. The distance of the code never falls below 3 so, as discussed in Sec. 3.2.6, the process is fault-tolerant.

Due to its importance, the literature contains several other approaches to map between the Steane and Reed-Muller codes. Hwang et al. produced a circuit based on the particular structure of these codes and the fact that they can both

be transformed into a common canonical form [38]. Anderson, Duclos-Cianci, and Poulin made the insightful observation that these two codes could be considered to be identical subsystem codes – with additional stabilizers added to fix the gauge [39]. They were able to use this structure to generate a map between the codes. Bombin also made similar observations [40]. Quan et al. later extended this idea [41]. Our approach is in the same category as these gauge-fixing methods, but has the added benefit that it automatically finds the minimal set of operators which need to be measured. It is purely algorithmic and does not require any insight.

As a related example, Huang and Newman used our algorithm to create a mapping between the 5-qubit code and the Steane code [62].

3.3.2 Moving defects in topological codes

Topological codes are stabilizer codes where the spins are arranged on a lattice, the stabilizer generators are local (meaning they only involve spins which are near one-another), but the logical operators are non-local. These codes are particularly robust against local noise sources. They are also of great intellectual interest, as they have connections to gauge theories, spin liquids, and topological order. These codes are typically translationally invariant, but it can be advantageous to introduce “extrinsic defects,” which locally disrupt the wiring. One can implement gates by deforming the code so that these defects move around one-another [68]. Here we apply our algorithm to the problem of moving these defects – reproducing known protocols for moving “ e ” and “ m ” type defects in surface codes, and finding new protocols for converting between these defects, and moving “twist defects”. The latter is valuable beyond quantum information processing, as moving twist

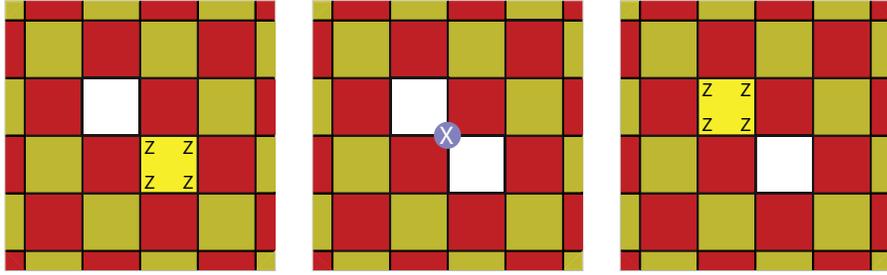


Figure 3.1: Transporting an e type defect in the surface code. Qubits sit at the vertices of the square lattice, yellow (light) and red (dark) plaquettes denote stabilizers corresponding to the product of Z or X operators on the four qubits at the corners. The white squares represent the absence of a stabilizer. To convert from the code on the left to the one on the right, one first measures the X operator highlighted in the middle panel, then the highlighted stabilizer in the right panel.

defects around one-another is equivalent to braiding Majorana fermions, which would allow the direct observation of excitations with non-Abelian statistics.

We will explicitly consider the toric/surface code. There are several conventions for defining this code. We follow [69], and place the qubits on the vertices of a checkerboard lattice with alternate plaquettes colored yellow and red (Fig. 3.1). For each yellow plaquette there is a stabilizer corresponding to the product of Z 's on the four qubits at the vertices. For each red plaquette there is another stabilizer corresponding to the product of X 's. The number of logical qubits stored by the code depends on the boundary conditions and the topology of the surface tiled by the qubits.

The simplest defect involves “removing” one of the stabilizers – meaning that one removes that generator from G , reducing the constraints on the codespace. In the presence of appropriate boundaries, this extrinsic defect yields one additional logical qubit. This defect is referred to as an Z or X defect (or equivalently e and m) depending if the removed stabilizer is a product of Z 's or X 's. Bombin [44,45] uses the phrase “code deformation” to describes the operations required to move

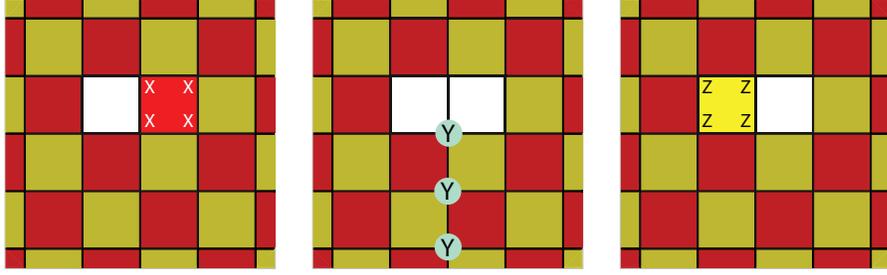


Figure 3.2: Changing an e type defect into an m type defect. This conversion requires measuring a non-local operator corresponding to a product of Y operators extending from the defects to the edge of the sample.

these defects, and the algorithm is explained in detail by Fowler [68]. Our general code rewiring algorithm can be used to find these operations.

Consider for instance the setup in Fig. 3.1 where one wishes to move a Z defect (shown as a white square in the leftmost panel) to a diagonally adjacent site (rightmost panel). The two codes differ by only one generator: $g_0 = Z_1 Z_2 Z_3 Z_4$, $g'_0 = Z_4 Z_5 Z_6 Z_7$, where Z_4 acts on the shared qubit. These are logical operators for the other code (ie. lie in the B blocks). The complementary logical operators involves a product of X operators extending from the each of the defects to the boundary. The product of these logical operators is simply X_4 . Thus, following the prescription in Sec. 3.2.2, one rewires the code by first measuring X_4 then measuring g'_0 . This protocol coincides with the one in [68].

As a second example, consider Fig. 3.2 where one wishes to convert a Z defect into an X defect on a neighboring plaquette. Again the codes differ in only one generator, and the generators are logical operators for the other code. The complementary operators this time are a string of X 's and a string of Z 's extending to a boundary. Their product is a string of Y 's. Thus rewiring this code requires measuring a non-local quantity. In many topological quantum computer architectures this is impractical, as one wishes to avoid measuring non-local operators. In such

a case, one could first move the defect near a boundary, where the string becomes short, at the cost of reducing the distance of the code.

Finally we consider a “twist” defect, illustrated in Fig. 3. Two of the square plaquettes are replaced by pentagons, and the plaquettes between them become rhombuses. As illustrated in the figure, the stabilizer associated with a pentagon involves measuring a product of three X operators, a Y and a Z or the product of three Z operators, a Y and a X . The stabilizer associated with a rhombus involves the product of two Z 's and two X 's. Twist defects are important for several reasons. For example, an X defect can be converted into a Z defect by moving it through the rhombus cells – a procedure that only requires local measurements. Given appropriate boundary conditions, adding a twist defect yields one additional logical qubit.

We wish to consider how to deform the code in order to move one of the pentagons. For example, in Fig. 3 we illustrate a move in which the defect is made shorter. In this case, the code changes by two generators: On the left, $g_0 = Z_1 Z_2 X_4 X_5$ and $g_1 = X_2 X_3 Z_5 Y_6 X_7$. On the right, $g'_0 = Z_1 Z_2 X_4 Y_5 Z_6$ and $g'_1 = X_2 X_3 X_6 X_7$. Following the algorithm in Sec. 3.2, we find a new set of generators $G = \{g_0, g_1 g_0\}$, $G' = \{g'_0, g'_1 g'_0\}$, where $g_0 g_1 = g'_0 g'_1$, and g_0 anti-commutes with g'_0 . Thus to convert between the codes one need only measure g'_0 .

These manipulations allow for operations which can be interpreted as braiding quasi-particles with unusual quantum statistics. As discussed by Kitaev [60], the codespace can be identified with the ground-space manifold of a Hamiltonian which is the sum of the stabilizer generators. Turning off one of the generators, say Z_0 , enlarges the space. States that are eigenstates of Z_0 with eigenvalue -1 are identified with excited states of the Hamiltonian and are said to contain a quasi-

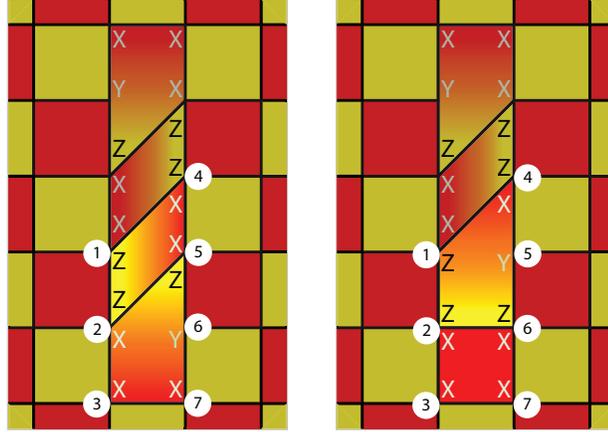


Figure 3.3: Shortening a twist. Twist defects, as illustrated in this figure, contain stabilizers corresponding to products of four or five operators. The ends of these defects have properties analogous to Majorana fermions. One converts the code on the left to the one on the right by measuring the operator $X_2X_3X_6X_7$. Our algorithm similarly gives the procedure for other moves.

particle at that location. Moving a Z defect around a X defect produces a control-not gate [68], which in this context can be interpreted as a phase that is contingent on the presence of the quasi-particles. Hence, there is a phase generated by moving one quasi-particle around another, which is the defining property of an anyon. The twist defects are even more interesting, as in this mapping the pentagons play the role of particles with non-Abelian statistics (described either as Ising anyons or Majorana fermions) [69].

3.4 Summary and Outlook

Quantum codes, first introduced for error correction, become more powerful with operations that map between them. The way information is encoded influences what operations are most accessible; moreover, gates can be applied through the act of mapping between codes. This code-conversion paradigm is at the heart of one-way quantum computing and topological quantum computing. In this paper, we

introduce a general-purpose algorithm for mapping between two arbitrary stabilizer codes, often in a fault-tolerant manner. We illustrate two applications of the algorithm: mapping between the Steane and Reed-Muller codes and moving defects in topological codes.

Our algorithm provides a means for creating a mapping between arbitrary stabilizer codes. Such mappings are not unique, and depending on which path is taken, a different logical gate can be produced. It would be exciting to extend our approach and gain the ability to choose which logical operation is performed: i.e. given a desired logical operator, how does one construct a sequence of measurements that performs the desired gate? Similarly, it would be useful to develop approaches that allow the incorporation of constraints, such as locality or code distance [62].

Acknowledgements

We thank Bryan Eastin for critical comments, including providing the example in 3.5.4. This material is based upon work supported by the National Science Foundation Grant No PHY-1508300, and the ARO-MURI Non-equilibrium Many-body Dynamics grant (W911NF-14-1-0003).

3.5 Appendix

3.5.1 Properties of U

Here we establish that $U = (1 + g'_0 g_0)/\sqrt{2}$ in Eq. (3.1) is a unitary Clifford rotation, and we verify its relationship to the projectors $P_{\pm 1} = (1 \pm g'_0)/2$. We rely on the fact that g'_0 and g_0 are Pauli operators that anti-commute, $\{g'_0, g_0\} = 0$. The fact that U is unitary follows from writing

$$UU^\dagger = (1 + g'g)(1 + gg')/2 = (1 + g'ggg' + \{g', g\})/2, \quad (3.2)$$

where we have neglected the subscript. We then note that for any Pauli operator, $g^2 = (g')^2 = 1$, and hence $UU^\dagger = 1$. Similar arithmetic gives $UgU^\dagger = g'$.

The relationships with the projectors come from noting that for any state $|\psi\rangle$ in the S -codespace, $g|\psi\rangle = |\psi\rangle$. Thus

$$U|\psi\rangle = \frac{1 + g'g}{\sqrt{2}}|\psi\rangle = \frac{1 + g'}{\sqrt{2}}|\psi\rangle = \sqrt{2}P_1|\psi\rangle \quad (3.3)$$

$$U|\psi\rangle = \frac{1 + g'g}{\sqrt{2}}|\psi\rangle = \frac{g - gg'}{\sqrt{2}}|\psi\rangle = \sqrt{2}gP_{-1}|\psi\rangle. \quad (3.4)$$

To show that U is a Clifford rotation, we first recall that a Clifford rotation is defined by the property that for any σ in the Pauli group on n qubits, P_n , $U\sigma U^\dagger = \sigma' \in P_n$. We take an arbitrary $\sigma \in P_n$, and separately consider the two cases where σ commutes or anti-commutes with gg' . (One of these conditions is always satisfied by any two arbitrary Pauli operators.) If $[\sigma, gg'] = 0$, then $U\sigma U^\dagger = \sigma \in P_n$. If $\{\sigma, gg'\} = 0$, then $U\sigma U^\dagger = (\sigma + 2\sigma gg' - \sigma)/2 = \sigma gg'$. Products of Pauli operators are Pauli operators (P_n is closed under multiplication), so $\sigma gg' \in P_n$.

3.5.2 Fault-tolerant measurement of Pauli operators

A stabilizer measurement requires measuring operators of the form $g = \sigma_1 \sigma_2 \cdots \sigma_m$, which is a product of m Pauli matrices, each acting on a different qubit, labeled by $j = 1, \dots, m$. Here we reproduce the argument from Shor [3] (see also, [4]), showing that this measurement can be done in a fault-tolerant manner.

One first prepares m ancilla qubits in a m -qubit cat-state, which is an equal superposition of all ancilla qubits in the $|0\rangle$ state and all ancilla qubits in the $|1\rangle$ state. As shown in Fig. 3.4, one then entangles the j 'th ancilla qubit with the system by applying a control- σ_j operator. A Hadamard gate is applied to each ancilla qubit, then the ancilla are measured in the standard basis. A measurement having even parity will have projected the encoded state into the $+1$ eigenbasis of g . A measurement having odd parity will have projected the encoded state into the -1 eigenbasis of g . Neglecting normalization, the quantum state of the composite system evolves as $(|0\rangle^{\otimes m} + |1\rangle^{\otimes m}) \otimes |\psi\rangle \rightarrow |0\rangle^{\otimes m} \otimes |\psi\rangle + |1\rangle^{\otimes m} \otimes g|\psi\rangle \rightarrow |e\rangle \otimes (1 + g)|\psi\rangle + |o\rangle \otimes (1 - g)|\psi\rangle$, where $|e\rangle$ and $|o\rangle$ are equal weight superpositions of all of the ancilla states with even and odd parity. Each ancilla qubit interacts with a single, unique physical qubit.

While this scheme prevents errors from propagating, an error on one of the ancilla qubits could be mistaken for an error on one of the system qubits (or could mask such an error). The standard procedure for addressing this problem is to repeatedly measure the stabilizer. By comparing subsequent measurements, one can bound the probability of an undetected or misdiagnosed error, yielding a fault-tolerant algorithm.

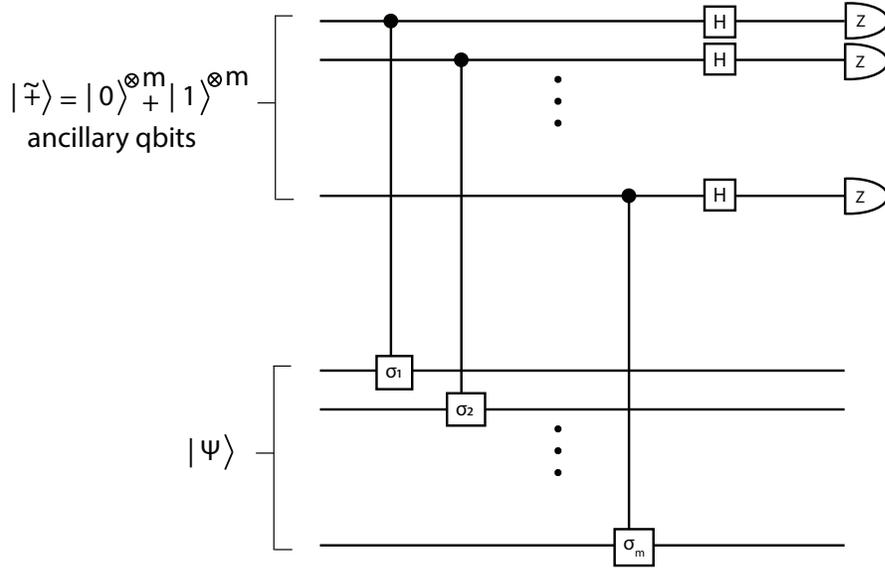


Figure 3.4: Circuit illustrating the cat-state method [3, 4] for the fault-tolerant projection measurement of a stabilizer generator.

3.5.3 Trivial Examples

To illustrate the mechanics of our algorithm, we carefully apply it to two simple examples. These are not physically relevant, but they provide a platform for understanding the arithmetic.

First consider the case where we have $n = 2$ physical qubits, storing $k = 1$ logical qubits. Suppose S is generated by $g_0 = Z_1$ and S' is generated by $g'_0 = Z_2$. Following our algorithm, we construct the 1×1 connectivity matrix M , which in this case is equal to zero. This tells us that our generators are either equal to one-another (which would put them in blocks G_A and G'_A), or they are logical operators of the other code (which puts them in blocks G_B and G'_B). Clearly the latter is the case. The complementary operators are $g_0^{(c)} = X_1$ and $(g'_0)^{(c)} = X_2$. The sequence of stabilizer codes is then generated by $G_0 = \{Z_1\}, G_1 = \{X_1 X_2\}, G_2 = \{Z_2\}$. Each subsequent set of generators differs by exactly one non-commuting element,

allowing us to map between them via the the procedure in Sec. 3.2.2.

A somewhat more sophisticated, but equally artificial, example involves $n = 3$ physical qubits and $k = 1$ logical qubits. Suppose S is generated by $g_0 = Z_1Z_2, g_1 = Z_3$ while S' is generated by $g'_0 = Z_1, g'_1 = X_2X_3$, The connectivity matrix is then

$$M = \begin{pmatrix} 0 & 1 \\ 0 & 1 \end{pmatrix}. \quad (3.5)$$

This is made diagonal via the transformation $g_0 \rightarrow \bar{g}_0 = g_0g_1 = Z_1Z_2Z_3$. By construction, S is generated by \bar{g}_0 and g_1 . The operators \bar{g}_0 and g'_0 are logical operators for the other code (so are in block G_B and G'_B), and the complementary operators can be taken to be $\bar{g}_0^{(c)} = X_3$ and $g'_0{}^{(c)} = X_1X_2$. Following Sec. 3.2.4, we then check the anti-commutator of these complementary operators with the elements of S and S' . We find that $\bar{g}_0^{(c)}$ anti-commutes with g_1 , necessitating the replacement $g_1 \rightarrow \bar{g}_1 = g_1\bar{g}_0$. The sequence of stabilizer generators is then: $\{Z_1Z_2Z_3, Z_1Z_2\} \rightarrow \{X_1X_2X_3, Z_1Z_2\} \rightarrow \{Z_1, Z_1Z_2\} \rightarrow \{Z_1, X_2X_3\}$.

3.5.4 Code distance during rewiring

The distance of a code d is the minimum number of single qubit errors required to give a non-zero overlap between two basis states in the code-space. Equivalently, d is the minimum number of single qubit Pauli operators which are needed to construct a logical operator. In a distance d code one can correct $t = (d - 1)/2$ errors. As described in Sec. 3.2.5, an important question is how the distance of the code evolves along the path in Sec. 3.2.3. In that section we established a lower bound on the distance of these codes. Nonetheless, in many practical examples (Secs. 3.3.1, 3.3.2) the distance of the code is maintained throughout the path.

Here we explicitly construct a mapping between two distance 3 codes for which our algorithm yields an intermediate code of lower distance.

In particular, consider the case where $S_0 = S$ is the Steane code defined by the first six stabilizers in the left-hand column of Table 3.1, and $S_2 = S'$ is a modified version of the same code where the third and fourth qubit are exchanged – that is in each stabilizer one makes the substitution $X_3 \rightarrow X_4, Z_3 \rightarrow Z_4, X_4 \rightarrow X_3, Z_4 \rightarrow Z_3$. Both of these are distance 3 codes. Our algorithm gives an intermediate code S_1 which is generated by $\{Z_1Z_4Z_5Z_7, X_1X_2X_5X_6, X_1X_3X_4X_6, Z_1Z_3Z_5Z_7, Z_1Z_2Z_5Z_6, Z_1Z_3Z_4Z_6\}$. This code is only distance 1 as the operator Z_7 commutes with all of the generators, but is not itself a stabilizer – and is therefore a logical operator. Thus if a “phase” error occurs on qubit 7 in this intermediate state, the error could neither be detected, nor corrected.

One approach to avoiding this issue is given in [62].

3.5.5 Connectivity matrices for mapping between Steane and Reed-Muller Codes

Here we explicitly show how our algorithm is applied to finding a mapping between the Steane and Reed-Muller codes. As described in Sec. 3.3.1, we first append arbitrary stabilizers to the Steane code so that there are an equal number of stabilizers in both codes (see Table 3.1). Then, we form the connectivity matrix and find its diagonal form by mod(2) row reduction.

The row reduction procedure yields a new set of generators that are used to

Table 3.2: Connectivity matrix for the Steane and Reed-Muller codes. A one in the row indicates an anti-commutation relation, and zeros are inserted where the stabilizer generators are identical.

	g'_0	g'_1	g'_2	g'_3	g'_4	g'_5	g'_6	g'_7	g'_8	g'_9	g'_{10}	g'_{11}	g'_{12}	g'_{13}
g_0										1				
g_1								1						
g_2										1				
g_3												0		
g_4													0	
g_5														0
g_6					1									
g_7	1				1									
g_8		1			1									
g_9	1	1			1									
g_{10}				1	1									
g_{11}	1			1	1									
g_{12}		1	1	1										
g_{13}	1	1	1	1										

construct the sequence of measurements described in Sec. 3.3.1. We additionally use column reduction in order to produce the simplest sequence of measurements for mapping in the opposite direction. We start by creating the connectivity matrix, Table 3.2, corresponding to the operators in Table 3.1. We show ones where the stabilizers anticommute, and zeros are implied at all other locations. We place explicit zeros in the locations where the stabilizers are identical.

We first note that the first six rows and the last six columns decouple from the others, and we can diagonalize them by simply reordering the rows and columns. Three of these generators are identical (and hence belong in block A), $(g_3, g_4, g_5) = (g'_{11}, g'_{12}, g'_{13})$. The other elements form anti-commuting pairs (and hence belong in block C), $\{g_0, g'_9\} = \{g_1, g'_8\} = \{g_2, g'_{10}\} = 0$.

We diagonalize the last 8 rows using row reduction mod(2). First, we rearrange the rows so that the rows with ones in the first column are at the top, rows with ones in the second column are next, etc. This yields an ordering $g_{13}, g_{11}, g_9, g_7, g_{12}, g_8, g_{10}, g_6$. We add the first row (mod 2) to the second,

Table 3.3: Connection matrices for intermediate steps of the row reduction used to find the generators needed to map from the Steane code to the Reed-Muller code.

	g'_0	g'_1	g'_2	g'_3		g'_0	g'_1	g'_2	g'_3
g_{13}	1	1	1	1	$g_{13}g_{12}$	1			
$g_{11}g_{13}$		1			$g_{11}g_{13}$		1		
g_9g_{13}			1		g_9g_{13}			1	
g_6				1	g_6				1
g_{12}		1	1	1	$g_{12}g_6g_9g_{11}$				
g_8		1		1	$g_8g_6g_{11}g_{13}$				
g_7g_{13}		1	1		$g_7g_{13}g_{11}g_9$				
g_{10}			1	1	$g_{10}g_6g_9g_{13}$				

third, and fourth, to eliminate the ones in the first column, yielding generators $g_{13}, g_{11}g_{13}, g_9g_{13}, g_7g_{13}, g_{12}, g_8, g_{10}, g_6$. Table 3.3 shows the resulting connection matrices after slight reordering. We then pivot, adding the fifth row g_{12} to the first g_{13} , making the first four rows diagonal. The remaining 1's in the last four rows are then readily eliminated by adding rows from this diagonal block (see Table 3.3).

Thus we have four new elements for block C: $\{g_{13}g_{12}, g'_0\} = \{g_{11}g_{13}, g'_1\} = \{g_9g_{13}, g'_2\} = \{g_6, g'_3\} = 0$. The remaining generators either belong to block A (meaning they are equal to stabilizers of the other code) or block B (in which case they are logical operators for the other code). Given that the two codes share a logical operator $Z = \prod_j Z_j$, block B must be empty, and the rest of the generators are in block A.

To find the most convenient sequence to convert backwards from the Reed-Muller code to the Steane code, we follow the same procedure but use column reduction. We reorder the rows as shown in Table 3.4, and add columns together (mod 2) in order to reduce the lower four rows to diagonal form. To simplify the expressions we also reorder the rows. Finally, we use row reduction to eliminate all ones in the upper four rows. This establishes a simple mapping backwards.

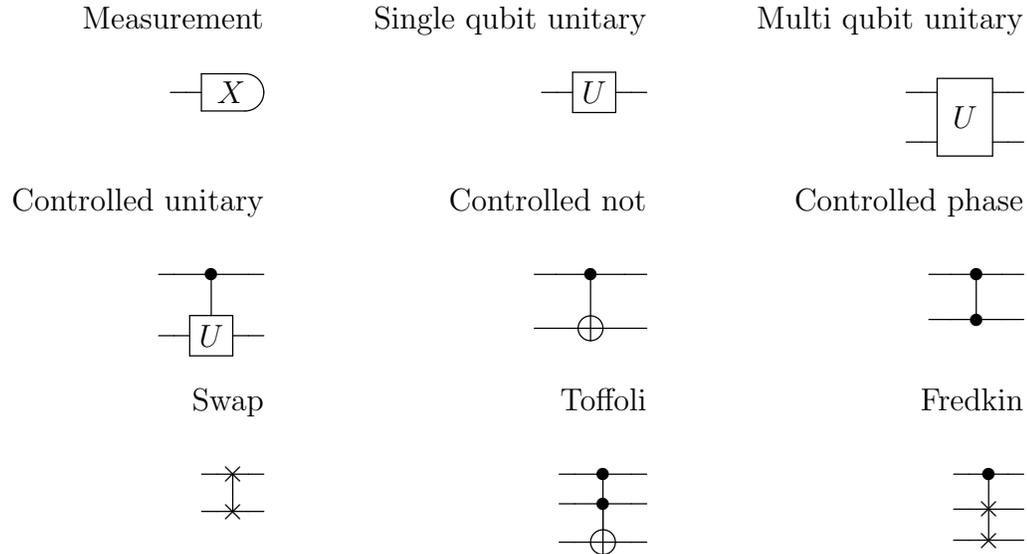
To map a state from the codespace of the Reed-Muller code to that of the Steane code, we measure: $g_0, g_1, g_2, g_6, g_8, g_9, g_{10}$.

Table 3.4: Connection matrices for intermediate steps of the column reduction used to find the generators needed to map from the Reed-Muller code to the Steane code.

	g'_0	g'_1	g'_2	g'_3		g'_0	$g'_1 g'_0$	g'_2	$g'_3 g'_2 g'_1$
g_{13}	1	1	1	1	g_{13}	1		1	1
g_{11}	1		1	1	g_{11}	1	1	1	
g_9	1	1		1	g_{12}		1	1	1
g_7	1			1	g_7	1	1		1
g_{12}		1	1	1	g_9	1			
g_8		1		1	g_8		1		
g_{10}			1	1	g_{10}			1	
g_6				1	g_6				1

	g'_0	$g'_1 g'_0$	g'_2	$g'_3 g'_2 g'_1$
$g_{13} g_9 g_{10} g_6$				
$g_{11} g_9 g_8 g_{10}$				
$g_{12} g_8 g_{10} g_6$				
$g_7 g_9 g_8 g_6$				
g_9		1		
g_8			1	
g_{10}				1
g_6				1

3.5.6 Common Quantum Circuit Operators



The Pauli group can be generalized to higher dimensions, d . Instead of generating the Pauli group from the two-dimensional X and Z , we generate it from tensor products of X_d and Z_d where $X_d |j\rangle = |j \oplus 1\rangle$ and $Z_d |j\rangle = \omega^j |j\rangle$, where ω is the primitive d -th root of unity (i.e. $\omega = e^{2\pi i/d}$), and the addition \oplus is performed modulo d [70]. Another equivalent representation is [71],

$$X_d = \sum_{j \in \mathbb{Z}_d} |j \oplus 1\rangle \langle j| \quad Z_d = \sum_{j \in \mathbb{Z}_d} \omega^j |j\rangle \langle j|. \quad (3.6)$$

The operators satisfy the commutation relation $X_d Z_d = \omega^{-1} Z_d X_d$. (This is easy to see by first multiplying the operators and then using orthogonality.)

CHAPTER 4
**SOME BOUNDS FOR QUANTUM ERROR-CORRECTING CODE
DECODERS**

Quantum computing hardware is susceptible to noise which causes the destruction of the quantum information in the computing system. Noise is a limiting factor of the scalability of quantum system – it limits the number of qubits (width) and the number of calculation time steps (depth). Fortunately, quantum error-correcting codes can mitigate certain errors in the system, allowing for scalability under certain conditions.

Quantum error correction involves a process called syndrome decoding. Syndromes are measured eigenvalues of the parity check operators of the quantum code. Eigenvalues that do not correspond to the desired eigenspace of the code are interpreted as errors in the quantum state. Syndrome decoding algorithms take in the errors detected from the measurements and output a best guess at a correction to those errors. Decoding calculations are performed on a classical computer, and the corrections are stored and applied to the corrupted quantum information that is measured at the end of the quantum computation.

The threshold for error is a measure of how well the decoder performs. The threshold is the maximum physical error probability below which a quantum calculation can be corrected to arbitrarily low logical error probability. When physical error rates are above the threshold value, running the decoding algorithm produces logical error rates that are worse than if the calculation was done without any error correction.

Thresholds of decoding algorithms can be computed for a single time step.

Neglecting the measurement errors, which form error patterns in time, gives the threshold for a quantum memory rather than for a quantum computation. Although this implies an unrealistic situation where the system can be measured quickly and performed instantaneously, one can generalize the decoder to a fully fault-tolerant setting [72]. Also see Sec. 4.1.1.

An optimal decoder is one which gives the highest threshold for the error model (a description of the errors that occur on the system) and the measurements from the error correcting code at hand. In the following sections, we calculate an upper limit to the thresholds of decoders for several error models and the information provided by the syndrome measurements, and we compare our bounds to the thresholds calculated numerically in the literature.

4.1 Upper Bounds on Thresholds of Error Correcting Codes

Takeda and Nishimori conjectured that optimal decoding thresholds should correspond to the hashing bound by showing that the hashing bound thresholds apply to more general models in statistical mechanics [73, 74]. Their conjecture remains an open question, but numerical results support it [5, 71, 72].

In this section, we use the hashing bound and entropy arguments to find an upper bound for the thresholds of error correcting codes for a variety of models – each model is based on the types errors the quantum code can detect and the types of errors present in the quantum system. We compare our calculations to thresholds found in the literature.

One way to protect a noisy quantum system is to encode the qubits using a stabilizer code and to perform a combination of protective stabilizer measurements on the system. These stabilizer measurements reveal information about the errors present on the system, and they give partial information about what errors occurred on the physical qubits. The amount of information gained about the independent errors on the physical qubits from these measurements can be quantified by the information entropy.

The information entropy can be interpreted as the average amount of information we need about the qubit to determine which error occurred on the qubit (i.e. It is the average amount of information we need to answer the question, “Did the qubit undergo a bit-flip, or a phase-flip, etc.?”). To find the form of the entropy for a noisy quantum system, we must consider the types of errors that can occur on the physical qubits and what the measurements on the system can tell us about those errors – the entropy is dependent on the error model of the system and the measurements made on the system. If an independent error occurs on a physical qubit (or qudit with dimension D) with a probability p_i , and the information is measured in dits having dimension D , then the information entropy gained by a measurement of the qubit is,

$$H_D = - \sum_i^n p_{i=1} \log_D(p_i). \quad (4.1)$$

The number of terms in the sum correspond to the number of operations that can occur on the system.

Let us look at an example where we have a single error: a bit-flip error model where a bit-flip error occurs with probability p_x . We have the error channel $\rho \rightarrow \epsilon(\rho)$, where

$$\epsilon_2^{bf}(\rho) = (1 - p_x)I\rho I + p_x X\rho X. \quad (4.2)$$

Say we measure the qubit, and our measurement determines if a the bit has flipped or not – the measurement determines in which X-eigenstate the qubit resides. The entropy we gain is,

$$H_2^{bf}(p_x) = -(1 - p_x) \log_2(1 - p_x) - p_x \log_2 p_x. \quad (4.3)$$

Here we are viewing the qubit as a two-sided coin where it remains heads with probability $1 - p_x$ (the identity operation occurs) or it is tails with probability p_x (a bit-flip operation occurs). A measurement on the coin answers the question of whether or not the coin is heads or tails, and so it provides information about which operation occurred.

The amount of information we learn from a measurement event is dependent on the uncertainty we have of the outcome. Continuing with this example, let us consider the case where we are entirely uncertain if a qubit was subjected to a bit-flip error, where $p_x = 1/2$. In this case, the entropy gained by measuring the qubit is,

$$H_2^{bf}\left(\frac{1}{2}\right) = -\log_2\left(\frac{1}{2}\right) = \log_2(2) = 1. \quad (4.4)$$

The information we gain is measured in bits, so the base of the log is 2. Thus, we learn one bit of information from the measurement. If we have some prior knowledge of the outcome, say we know a bit-flip error occurs with $p_x \neq 1/2$, then $H_2^{bf}(p_x)$ is always less than 1 – we learn less than one bit of information from the measurement.

If our physical system has more than two degrees of freedom, then we can extend the model to account for the extra degrees of freedom. In D dimensions we have the error channel,

$$\epsilon_D^{bf}(\rho) = (1 - p)I\rho I + \frac{p}{D - 1} \sum_{i=1}^{D-1} X_D^i \rho (X_D^i)^\dagger, \quad (4.5)$$

where X_D^i is the D -dimensional generalized Pauli-X operator, see Sec. 3.5.6.

If our measurement determines in which X-eigenstate the qudit resides, the entropy we would gain from that measurement is,

$$H_D^{bf}(p) = -(1-p)\log_D(1-p) - p\log_D\frac{p}{D-1}. \quad (4.6)$$

Similarly, we can impose a phase-flip error model on the quantum system and measure in which Z-eigenstate the qudit resides. In this case, we find an expression for the entropy as given in Eqn. 4.6 where p is instead represents the probability of a phase flip. When the noise channel is symmetric (i.e. the probability of a bit flip is the same as the probability of a phase flip), we have $H_D^{pf}(p) = H_D^{bf}(p)$.

Now let us consider the case where three types of errors can occur on a qubit with equal probability. This corresponds to a depolarizing error model – the probabilities of Pauli-X, -Y, and -Z errors are $p/3$, where p is the total probability of having any error. The qubit depolarizing error model is,

$$\epsilon_2^{dp}(\rho) = (1-p)I\rho I + p/3(X\rho X + Y\rho Y + Z\rho Z), \quad (4.7)$$

and the entropy gained from measuring the qubit is,

$$H_2^{dp}(p) = -(1-p)\log_2(1-p) - p\log_2(p/3). \quad (4.8)$$

We can generalize further to a qudit depolarizing error model where there are $D-1$ Pauli-X errors, $D-1$ Pauli-Z errors, and $(D-1)^2$ Pauli-Y errors giving,

$$\epsilon_D^{dp}(\rho) = (1-p)I\rho I + \frac{p}{D^2-1} \left(\sum_{k=1}^{D-1} X_D^k \rho (X_D^k)^\dagger + \sum_{k=1}^{D-1} Z_D^k \rho (Z_D^k)^\dagger + \sum_{l=1}^{D-1} \sum_{m=1}^{D-1} (iX_D^l Z_D^m) \rho (iX_D^l Z_D^m)^\dagger \right). \quad (4.9)$$

The entropy gained by a measurement of the D -level qudit is,

$$H_D^{dp}(p) = -(1-p)\log_D(1-p) - p\log_D\frac{p}{D^2-1}. \quad (4.10)$$

Now say we have a qudit depolarizing noise model, but the measurement only provides bit-flip information. In this case, we cannot distinguish between I or Pauli-Z errors and Pauli-X or Pauli-Y errors. Since there are $D - 1$ Pauli-Z errors that cannot be distinguished from the identity, the probability of detecting the identity is $1 - p + \frac{p(D-1)}{D^2-1} = 1 - p\frac{D}{D+1}$. Similarly, since our measurement cannot tell us the difference between Pauli-Y errors and Pauli-X errors, the probability of detecting one bit-flip error is $\frac{p}{D^2-1} (1 + D - 1) = \frac{pD}{D^2-1}$. The expression for the entropy gained is,

$$H_D^{bf|dp}(p) = - \left(1 - \frac{D}{D+1}p\right) \log_D \left(1 - \frac{D}{D+1}p\right) - \frac{D}{D+1}p \log_D \frac{D}{D^2-1}p. \quad (4.11)$$

There are $D - 1$ Pauli-X errors occurring with probability $\frac{pD}{D^2-1}$, so there is a factor of $D - 1$ on the second term corresponding to the number of Pauli-X error terms in the sum.

We can calculate the entropy in a similar way for the case when we have a system subject to depolarizing noise and our measurements on the system provide only phase-flip information, as opposed to bit-flip information. In this case we cannot distinguish between I or Pauli-X and Pauli-Z or -Y errors, and we get an expression for the entropy which is the same as Eqn. 4.11.

Now that we have shown how to derive the entropies for several different models of the system (i.e. bit-flip error model with the measurement detecting Pauli-X errors, a depolarizing error model with the measurement detecting Pauli-X, -Y, and -Z errors, and a depolarizing error model with the measurement detecting only Pauli-X errors), we can use the quantum hashing bound to make a statement about the amount of information gained about the errors in the quantum system. The hashing bound is defined by [75, 76],

$$H_D^T(\bar{p}) \leq 1, \quad (4.12)$$

where H_D^T is the total entropy measured in units of D (i.e. dits) and \bar{p} is the error probability vector containing the possible error probabilities (in our case these probabilities are $p_x = p_y = p_z = p$, where the probabilities of each Pauli error are equal). The bound originates from the rate achievable by using a random coding protocol, given by $R = 1 - H_D^T(\bar{p})$. Values of \bar{p} where the rate, R , is zero correspond to the hashing bound threshold.

A quantum decoder interprets the measurements on the system and attempts to provide to us as much information as possible about the errors on the qubits. A perfect decoder would provide the maximum information about the errors on the system. A decoder that attains the equality in Eqn. 4.12 is said to be a perfect decoder [71].

We would like to calculate the maximum probability of error that a perfect decoder could handle, i.e. its error correction threshold. Below the threshold error probability, the information we have about the errors is sufficient for error correction and we can correct errors to arbitrarily high accuracy; above the threshold we cannot.

The error correction threshold depends on a given type of error model and the errors that are detected. We are considering symmetric error models, so the probability at which the quantum hashing bound is a maximum can be represented in terms of the single Pauli error probability, p .

Table 4.1 gives the probabilities, p , that maximize the hashing bound for qudit dimensions 2-6 for the three different models we discussed. We see that the bounds obey the predicted general trend of increasing with increasing qudit dimension [5, 71, 77].

Data in the first column lists the probabilities in the case where the system undergoes bit- and phase- flips and our decoder does not take into account Pauli-X and Pauli-Z error correlations. The error model corresponds to Eqn. 4.5 for both the bit flips and phase flips. The total entropy is the sum of the individual entropies gained from the independent information attained about the bit-flips and the phase-flips, $H_D^T(p)^{bf+pf} = H_D^{bf}(p) + H_D^{pf}(p) = 2H_D^{bf}(p)$ where $H_D^{bf}(p)$ is given by Eqn. 4.6. This model is equivalent to the error channel,

$$\epsilon_D^{bf/pf}(\rho) = (1-p)I\rho I + \frac{p}{2(D-1)} \sum_{k=1}^{D-1} X_D^k \rho (X_D^k)^\dagger + Z_D^k \rho (Z_D^k)^\dagger \quad (4.13)$$

where the probability p corresponds to the probability of both a Pauli-X or Pauli-Z error occurring. The form for the bit-flip entropy is $H_D^{bf_0}(p) = -(1 - \frac{p}{2}) \log_D(1 - \frac{p}{2}) - \frac{p}{2} \log_D \frac{p}{2(D-1)}$. The entropies for the independent bit-flip and phase-flip measurements are equal in the symmetric error model, $H_D^{bf_0}(p) = H_D^{pf_0}(p)$, so the total entropy is given by $H_D^T(p)^{bf/pf} = H_D^{bf_0}(p) + H_D^{pf_0}(p) = 2H_D^{bf_0}(p)$.

We can compare the thresholds calculated in first column to the values of the optimal error correction thresholds predicted using phase boundaries for disordered Potts gauge glasses, see Fig. 4.1 adopted from Fig. 3 in [5]. (The Potts gauge glass is a generalization of the random bond Ising model to qudits, see Appendix B in [71].) Furthermore, the bound calculated for $D = 2$ is the optimal threshold value predicted using the random bond Ising model by Dennis, et. al. [72].

The second column corresponds to the error model in Eqn. 4.13, except now both the Pauli-X and Pauli-Z errors can be measured simultaneously. The total entropy in this case is given by, $H_D^T(p)^{bpf} = -(1-p) \log_D(1-p) - p \log_D \frac{p}{2(D-1)}$. Notice the slight increase in the threshold from the first column. This increase is likely from the increased entropy gained from the simultaneous measurement of Pauli-X and Pauli-Z, as opposed to two independent measurements.

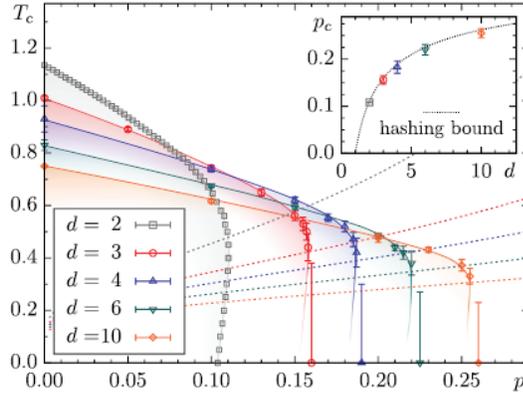


Figure 4.1: Phase boundaries estimated from Monte Carlo simulations of the respective disordered Potts gauge glasses representing quantum error correction for d -level quantum systems. The error thresholds for each dimension correspond to the points where the corresponding Nishimori line (dashed) intersects the phase boundary. The values of the hashing bound in the first column of Table 4.1 also compare well with the figure insert. Figure and caption adopted from [5].

The third data column lists the probabilities for a depolarizing error model where the decoder takes into account Pauli-X and Pauli-Z error correlations (which is the Pauli-Y error). The bound for $D = 2$ corresponds for the threshold of 18.9(3)% calculated in Bombin, et. al. [77] using Monte Carlo simulations that take into account these correlations. The error model is given by Eqn. 4.9 and the total entropy by Eqn. 4.10.

The fourth column gives the probabilities where the error model is depolarizing (we have Pauli X-, -Y, and -Z errors), but the measurement events only separately provide bit-flip or phase-flip information and do not take into account the Pauli-X and Pauli-Z correlations (the Pauli-Y errors). The total entropy is $2H_D^{bf|dp}(p)$ where $H_D^{bf|dp}(p)$ is given by Eqn. 4.11.

The highest thresholds for the independent bit- and phase-flip error model are in the second column, where the decoder can detect error correlations but the system not subject to them. The highest thresholds for the depolarizing error model are

calculated in the third column, where the decoder can detect error correlations. This motivates using decoders like the maximum likelihood decoders [78], which take into account Pauli error correlations.

If we want to compare the bit- and phase-flip models to the depolarizing models (i.e. comparing columns 1 and 2 with columns 3 and 4 in Table 4.1), then we need the thresholds to correspond to the probability of any error occurring (as opposed to the probability of just a Pauli-X or Pauli-Z error occurring). Since the thresholds in the first and second columns correspond to the probability of a single Pauli-X or Pauli-Z error occurring, and we have assumed that those probabilities are equal, we can multiply the thresholds in those columns by a factor of two. For example, in the first row of the first column, the threshold probability is $p_x = 11.00\%$. Since that probability corresponds to a single X (or Z) error, the total probability of both an X or Z error occurring is $p_T = p_x + p_z = 2p_x = 22.01\%$ (the 1 is from rounding the exact calculation to the fourth decimal place). We list the scaled values in Table 4.2.

Using this scaling for comparison of the two error models, from Table 4.2, we see an overall reduction in the threshold, for all dimensions, when the code is subject to correlated Pauli-X and Pauli-Z errors (the thresholds in the first and second columns of Table 4.2 are larger than the thresholds in the third and fourth columns). This reduction is from the inclusion of an additional flavor of error (the Pauli-Y error) in the depolarizing model.

The values of the hashing bound for the depolarizing model for dimension D are equal to the values of the hashing bound for the generalized bit flip error model for dimension D^2 . This is easily seen analytically by equating Eqns. 4.6 and 4.10.

D	GBF	BPF	DPN	GBF given DPN
2	0.1100	0.1135	0.1893	0.1650
3	0.1595	0.1667	0.2552	0.2126
4	0.1893	0.1992	0.2897	0.2366
5	0.2099	0.2218	0.3113	0.2518
6	0.2252	0.2388	0.3263	0.2627

Table 4.1: Qudit hashing bounds for generalized bit-flip (GBF), the bit- and phase-flip (BPF), the depolarizing noise (DPN), and the GBF given DPN channels for qudit dimension (D). The hashing bound is conjectured to be an upper bound of the threshold of a quantum decoder (i.e. it represents the highest threshold a quantum decoder can have given an error model and the types of errors the decoder detects).

D	GBF(s)	BPF(s)	DPN	GBF given DPN
2	0.2201	0.2271	0.1893	0.1650
3	0.3189	0.3333	0.2552	0.2126
4	0.3786	0.3984	0.2897	0.2366
5	0.4197	0.4437	0.3113	0.2518
6	0.4504	0.4776	0.3263	0.2627

Table 4.2: Qudit hashing bounds for qudit dimension (D) for the scaled generalized bit-flip, GBF(s) and the scaled bit- and phase- flip, BPF(s) channels. The depolarizing noise (DPN), and the GBF given DPN channels are included for comparison.

4.1.1 A Note on Faulty Syndromes

Each time stabilizers are measured, one obtains a pattern of syndromes corresponding to the errors which are detected. It is possible, however, for the syndromes to be erroneous. For example, errors can occur during the process of the projection measurement involved in measuring a stabilizer. For the computation to be fault-tolerant, the process of extracting a syndrome must also be fault-tolerant.

During a quantum calculation, syndromes are measured many times, and erroneous syndromes measurements can form error patterns in the time domain. Typically syndrome measurement errors are detected and corrected using a major-

ity vote - if the syndrome gives a result the majority of the time, then we assume the syndrome to be accurate. This assumes that the errors occurring on the code change more slowly than the time it takes for the all of the syndromes on the code to be measured multiple times to get a sufficient result. When syndrome errors are included in a model, it is called a gate model. A gate model of a stabilizer code takes into account the errors occurring on the code, which are seen from stabilizer measurements, and multiple measurements of the code are shown in time.

4.2 Upper Bounds on the Efficiency of Error-correcting Codes

The efficiency of an error-correcting code, either classical or quantum, is given by the ratio of the number of correctable words to possible words. In this section we use the Hamming bound to find an upper bound on the efficiency of error-correcting codes, and we examine the difference between the classical Hamming bound and the quantum Hamming bound using a sphere packing argument.

Consider a classical code that corrects up to t errors and has codeword vectors of length n with elements in \mathbb{Z}_q . The maximum number of correctable errors can be represented by a ball with a codeword at the origin and having a size corresponding to the number of correctable errors. The ball has size,

$$E_{max} = \sum_{k=0}^t \binom{n}{k} (q-1)^k. \quad (4.14)$$

The binomial coefficient counts the number of possible correctable words having k errors, and the factor $(q-1)^k$ counts the number of different errors. For example, let $q = 2$ (so we have bits), $n = 5$ (so the codewords are made of five bits), and

$t = 2$ (so we can correct up to two errors on each codeword). This can be an example of majority vote decoding, where if three or four bits out of five are the same (a codeword 10000 or 11000), then we say the word was the closest codeword having the majority of identical bits (we say it is supposed to be 00000). The size of the ball is $\sum_{k=0}^2 \binom{5}{k} = 16$.

Say we stick with the parameters in the previous example but instead had a dit with dimension $q = 3$. The codewords would have elements in $\{0, 1, 2\}$, and there would be a factor of 2^k to count for the possible errors on each dit. For example, we can choose a codeword 00000 and correct 2^1 codewords with an error on the first dit: 10000 or 20000. Since there are $\binom{5}{1}$ ways to have one error (it can be on the first dit, the second dit, etc.), we can correct a total of 10 codewords having one error.

Each codeword has a ball of size E_{max} surrounding it, and no two balls can intersect (i.e. if two balls corresponding to two different codewords intersected, then words inside of the intersection could belong to either codeword, and thus be uncorrectable). If the number of codewords is equal to A , (i.e. the code has rank A), then the union of all the balls is $A \sum_{k=0}^t \binom{n}{k} (q-1)^k$. The number of possible words is q^n so,

$$A \sum_{k=0}^t \binom{n}{k} (q-1)^k \leq q^n. \quad (4.15)$$

The equality is achieved when the code can correct all of the possible words. In this case each word lies in a single ball belonging to a unique codeword (i.e. one can partition the space of all words into correctable sets, where each set can be represented by a unique codeword). Rearranging this expression gives a bound on the total number of codewords, called the Hamming bound,

$$A \leq \frac{q^n}{\sum_{k=0}^t \binom{n}{k} (q-1)^k}. \quad (4.16)$$

A code that achieves equality is called a perfect code.

Now let us look at the form of quantum Hamming bound. For an $[[n, A, d]]$ code in \mathbb{Z}_q it is [79],

$$A \leq \frac{q^n}{\sum_{k=0}^t \binom{n}{k} (q^2 - 1)^k}. \quad (4.17)$$

Notice the difference between this expression and the expression for classical codes – the size of the ball is proportional to $(q^2 - 1)^k$ instead of $(q - 1)^k$. This difference arises from counting the number of single errors that can occur on one codeword. In a quantum code, a single error arises from one of three Pauli operators. There are $q^2 - 1$ Pauli errors which can occur (i.e. $q - 1$ Pauli-X errors, $q - 1$ Pauli-Z errors, and $(q - 1)^2$ Pauli-Y errors), so the ball size is proportional to $(q^2 - 1)^k$. In a classical code, only bit flips are present, so there are only $q - 1$ errors and the ball size is proportional to $(q - 1)^k$.

For example, say we have a codeword made from qubits, and we have one error on the first qubit. The operator corresponding to one error could be one three Pauli errors, X, Y, or Z (i.e. $(2^2 - 1)^1 = 3$). If we have a codeword made from bits, then there is no phase and the operator corresponding to one error is a bit flip (i.e. X, and $(2 - 1)^1 = 1$). See [80, 81].

Deriving the quantum Hamming bound for only Pauli-X and Pauli-Z errors follows similarly, with the ball size proportional to $(2(q-1))^k$: the quantum Hamming bound for independent X and Z error models is,

$$A \leq \frac{q^n}{\sum_{k=0}^t \binom{n}{k} 2^k (q - 1)^k}. \quad (4.18)$$

CHAPTER 5

SUPERCONDUCTING QUBIT DESIGN

Superconducting circuits are composed of wires carrying a superconducting current. These wires can be configured in different ways to form circuit components, like capacitors, inductors, and Josephson junctions, see Sec. 5.1. Since superconducting circuits are complicated networks different superconducting configurations, they can be difficult to model from first principles. Luckily there are two main ideas (see [6]) that allow us to simplify the description of quantum circuits: first, we can treat the quantum circuit as a network of lumped components, and second, we can quantize the circuit using canonically conjugate variables by replacing variables by operators. Quantization of a quantum circuit is analogous to quantization in wave mechanics where one can choose a basis in which the coordinates retain their meaning as numbers but the canonically conjugate variables, the momenta, become differential operators [82]. In superconducting circuits, these canonical variables are charge and phase operators.

An advantageous consequence of these simplifications is that a harmonic quantum circuit can be quantized by a measurement of its frequency response [7]. This observation follows from a reactance theorem developed by Foster in 1924 [83], which states that any classical reactive circuit (inductors and capacitors only, no resistors or power sources) can be fully described by the zeros and poles of the impedance. Applying the theorem to a harmonic superconducting circuit allows one to characterize the properties of the circuit using a single measurement of the impedance (or reactance, since there is no resistance) as a function of frequency.

As an example, let us quantize a simple quantum harmonic oscillator (quantum LC circuit, see page 12 in [6] or Fig. 5.1). We first write the charge and flux

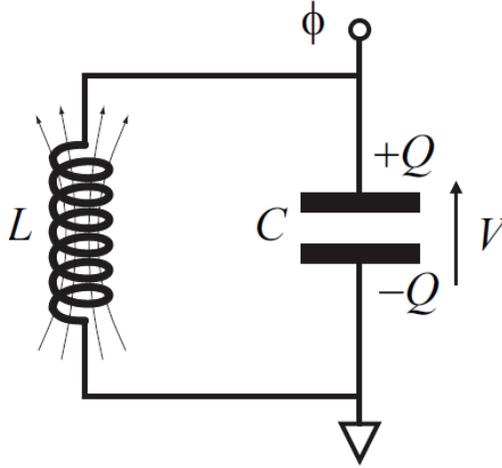


Figure 5.1: Harmonic oscillator LC circuit. Figure adapted from [6].

operators as $\hat{Q} = -iQ_{ZPF}(\hat{a} - \hat{a}^\dagger)$ and $\hat{\phi} = \Phi_{ZPF}(\hat{a} + \hat{a}^\dagger)$ where $Q_{ZPF} = \sqrt{\frac{\hbar}{2Z}}$ and $\Phi_{ZPF} = \sqrt{\frac{\hbar Z}{2}}$, and $Z = \sqrt{\frac{L}{C}}$ is the characteristic impedance of the oscillator. Here we already notice two fundamental scales of quantum circuits emerging: the resistance quantum, R_Q , and the superconducting flux quantum, Φ_0 . We can define a dimensionless characteristic impedance $z = Z/R_Q$ where $R_Q = \frac{\hbar}{(2e)^2}$ and $\Phi_0 = \frac{\hbar}{2e}$ (note: $2eR_Q = \Phi_0$). We can also see how the zero-point fluctuations in charge and phase are dependent on the characteristic impedance: $Q_{ZPF} = 2e\sqrt{\frac{1}{4\pi z}}$ and $\Phi_{ZPF} = \Phi_0\sqrt{\frac{z}{4\pi}}$.

Modern superconducting circuit designs incorporate many degrees of freedom. Having so many degrees of freedom, these circuits can be difficult to simulate. One way to find the majority of the circuit properties is to approximate each element in the circuit as a linear element and solve the resulting generalized eigenvalue problem for coupled harmonic oscillators. In this case, the inverse inductance matrix and capacitance matrices are used, and their sizes are equal to the number of isolated nodes (or islands) by the number of LC oscillators in the circuit [84]. Solving the generalized eigenvalue problem yields a collection of eigenvalue and

eigenvector solutions. Each eigenvalue corresponds to the frequency of a mode and each eigenvector describes the phase behavior across the nodes of the circuit. Taking the derivative of the eigenvectors with respect to the nodes describes the current through the circuit.

We can also quantize the circuit. One can use the collection of eigenvectors found from solving the generalized eigenvalue problem to write down the phase and charge operators. The phase variable matrix is formed using the eigenvectors and yields a matrix describing the zero point fluctuations in the circuit, $\Phi_{ZPF}^{\nu,n}$ where ν is the mode index and n is the node index. Using this matrix, one can write down the flux operator for a particular node by summing over all of the various modes in the multi-mode circuit, $\hat{\Phi}^n = \sum_{\nu} \Phi_{ZPF}^{\nu,n} (\hat{a}_{\nu} + \hat{a}_{\nu}^{\dagger})$. The conjugate variable (charge) is constructed similarly using the inverse transpose of the phase variable matrix.

We have just seen that we can find the mode structure as well as the phase behavior across the junctions and the superconducting current. We can also quantize the circuit using techniques from classical harmonic analysis.

The approximation of the superconducting elements as linear elements for harmonic mode analysis is called “black box quantization” [85, 86]. The term “black box” is used because the linear part of the circuit can be viewed as a black box, and one can characterize the main effects of the circuit by measuring the impedance (which can be accomplished by single-port measurements).

Black box quantization works well if the circuit is well within the transmon regime, meaning the ratio of the energy for Cooper pairs to mobilize across the junctions to the charging energy of the circuit is set so the effects of offset charges are low. In this case, the non-linearity of the Josephson junctions can be included

by using perturbation methods.

Modern software programs typically use the black box quantization approach to simulate superconducting circuits. The linear circuit is solved and then the junction nonlinearity is included to various orders using first-order perturbation theory (such as in QuCAT [87]).

This perturbative method also uses a second assumption in addition to low offset charge effects – the approximation is good at certain flux values where the effects of the junction non-linearity are low. Close to critical values of the flux, higher-order perturbative terms or other simulation methods are necessary.

Recently, several other methods have been proposed for finding effects due to offset charges and calculating the energy spectrum near critical values of the flux in highly multi-mode circuits [8, 11, 88–91]. It was found that the eigenspectrum and the dispersive effects of offset charges can be found when modeling the full Hamiltonian, even for circuits having a large number of junctions [8, 11]. Such circuits are introduced in Secs. 5.2, 5.3 and techniques for modeling them are discussed in Secs. 5.5, and 5.6.

5.1 The Josephson Junction

In this section we find the properties of a Josephson junction. This is discussed in [6], and here we include some additional details.

A Josephson junction can be viewed as a small gap between two relatively large superconducting metals across which Cooper pairs can tunnel. The superconducting metals lie on the left and right sides of the gap and are relatively large in size.

The total number of Cooper pairs on the metals is large: less than Avogadro's number but still large. Let us denote the number of Cooper pairs on the left by N_L and the number of Cooper pairs on the right by N_R . The total number of Cooper pairs on the two metals is the sum of the numbers on the left and the right, $N = N_L + N_R$

The Josephson junction can be modeled as a double well potential, where we label the left well c and the right d . The tunneling of a Cooper pair from the left side to the right is denoted by the annihilation of a pair in the left well and the creation of a pair in the right well, i.e. cd^\dagger . If one Cooper pair has a tunneling energy t , then the Hamiltonian is $H = t(d^\dagger c + c^\dagger d)$.

We can use the number of Cooper pairs on either side of the junction gap to represent the degenerate states. We label the degenerate states by the number of Cooper pairs that have tunneled from the left side of the junction to the right. If m Cooper pairs have tunneled from the left to the right, we have $N_L - m$ Cooper pairs on the left and $N_R + m$ Cooper pairs on the right, where this ground state is labeled $|m\rangle = |N_L - m, N_R + m\rangle$. For example, if one pair has hopped from the left well into the right, we have the state $|1\rangle = cd^\dagger |0\rangle \propto |N_L - 1, N_R + 1\rangle$.

In the m -basis, the Hamiltonian becomes,

$$H_T = -\frac{1}{2}E_J \sum_m |m\rangle \langle m+1| + |m+1\rangle \langle m| \quad (5.1)$$

where we have replaced the tunneling energy t with $-\frac{1}{2}E_J$, where E_J is the Josephson energy. The Josephson energy is derived from the Ambegaokar and Baratoff relation, which gives the junction energy from first principles of the superconducting material: $E_J = \frac{1}{2} \frac{\hbar}{(2e)^2} G_N \Delta$ where G_N is the conductance and Δ is the gap energy [92].

We have a model analogous to a 1D nearest neighbor hopping model in a crystal, and a trial general solution to this Hamiltonian could be the unnormalized wavefunction $|\phi\rangle = \sum_{m=-\infty}^{\infty} e^{im\phi} |m\rangle$, where $\hbar\phi$ is a momentum and $\phi = ka$ where the lattice constant $a = 1$. In this basis, we find $H_T |\phi\rangle = -E_J/2 \sum_{mj} (|m\rangle \langle m+1| + |m+1\rangle \langle m|) e^{ij\phi} |j\rangle = -\frac{E_J}{2} (e^{i\phi} |\phi\rangle + e^{-i\phi} |\phi\rangle) = -E_J \cos \phi |\phi\rangle$.

This is interesting. We have derived both the discrete hopping form of the Hamiltonian as well as the cosine form. First we viewed the Josephson junction as a double well potential, and by counting the number of Cooper pairs crossing the junction gap we have derived the discrete form of the Hamiltonian. Then by relating it to a nearest-neighbor 1D hopping model in a crystal, we derived the cosine form of the Hamiltonian.

5.1.1 First Josephson Relation

Our basis thus far labels degenerate states by the number of Cooper pairs that have crossed the junction gap. A quantity of interest we might like to measure is the number of Cooper pairs that have hopped across at some instant in time. In this basis we can define the number operator $\hat{n} = \sum_m |m\rangle m \langle m|$. We can also

define a rate for the hopping and define the current operator, $\hat{I} = 2e \frac{d\hat{n}}{dt}$.

$$\hat{I} = 2e \frac{d\hat{n}}{dt} \quad (5.2)$$

$$= 2e \frac{i}{\hbar} [H_T, \hat{n}] \quad (5.3)$$

$$= -2e \frac{i}{2\hbar} E_J [|m\rangle \langle m+1| + |m+1\rangle \langle m|, |j\rangle j \langle j|] \quad (5.4)$$

$$\begin{aligned} &\propto \sum_{mj} [|m\rangle \langle m+1|, |j\rangle j \langle j|] + [|m+1\rangle \langle m|, |j\rangle j \langle j|] \\ &\propto \sum_{mj} |m\rangle \delta_{j,m+1} j \langle j| - |j\rangle j \delta_{jm} \langle m+1| + |m+1\rangle \delta_{jm} j \langle j| - |j\rangle j \delta_{j,m+1} \langle m| \\ &\propto \sum_m |m\rangle (m+1) \langle m+1| - |m\rangle m \langle m+1| + |m+1\rangle m \langle m| - |m+1\rangle (m+1) \langle m| \\ &= \frac{e}{i\hbar} E_J \sum_m |m\rangle \langle m+1| - |m+1\rangle \langle m| \end{aligned} \quad (5.5)$$

$$= \frac{e}{i\hbar} E_J (\hat{h} - \hat{h}^\dagger) \quad (5.6)$$

where H_T is the Hamiltonian for the Josephson junction, Eqn. 5.1 and $\hat{h} = \sum_m |m\rangle \langle m+1|$ is the hopping operator. The first Josephson relation defines a relationship between the supercurrent and the phase across the junction. Applying our current operator to the state $|\phi\rangle$ yields the first Josephson relation:

$$\hat{I} |\phi\rangle = \frac{e}{i\hbar} E_J \left(\sum_m |m\rangle \langle m+1| - |m+1\rangle \langle m| \right) \left(\sum_{j=-\infty}^{\infty} e^{ij\phi} |j\rangle \right) \quad (5.7)$$

$$= \frac{e}{i\hbar} E_J \sum_{mj} |m\rangle \langle m+1| e^{ij\phi} |j\rangle - |m+1\rangle \langle m| e^{ij\phi} |j\rangle \quad (5.8)$$

$$= \frac{e}{i\hbar} E_J (e^{i\phi} |\phi\rangle - e^{-i\phi} |\phi\rangle) \quad (5.9)$$

$$= \frac{2e}{\hbar} E_J \sin \phi |\phi\rangle \quad (5.10)$$

$$= I_c \sin \phi |\phi\rangle \quad (5.11)$$

where we used the relation $\sum_m |m+1\rangle e^{im\phi} = e^{-i\phi} \sum_m |m+1\rangle e^{i(m+1)\phi} = e^{-i\phi} |\phi\rangle$.

Thus the maximum value of the current is reached where the sine is unity, and it is referred to as the critical current of the Josephson junction $I_c = \frac{2e}{\hbar} E_J$.

We can also recover the first Josephson relation by using the differential form for the number operator \hat{n} in Eqn. 5.3. We find the differential form for \hat{n} in a similar manner as in the previous section, where we found the differential form for the Hamiltonian of the Josephson junction, H_T by applying the discrete form of the operator to the unnormalized trial wavefunction $|\phi\rangle$. Here we have, $\hat{n}|\phi\rangle = \sum_{m,j} |m\rangle m \langle m| e^{ij\phi} |j\rangle = \sum_m m e^{im\phi} |m\rangle = -i\frac{\partial}{\partial\phi} |\phi\rangle$. Being careful to extract the form for \hat{n} we use: $\psi(\phi) = \langle\phi|\psi\rangle$ and $\langle\phi|\hat{n}|\psi\rangle = i\frac{\partial}{\partial\phi} \langle\phi|\psi\rangle$ so $\hat{n} = i\frac{\partial}{\partial\phi}$. Now we can recover the first Josephson relation by using the differential form for \hat{n} :

$$\hat{I} = 2e \frac{d\hat{n}}{dt} \quad (5.12)$$

$$= \frac{2e}{-i\hbar} [H_T, \hat{n}] \quad (5.13)$$

$$= \frac{2e}{-i\hbar} \left[-E_J \cos \phi, i\frac{\partial}{\partial\phi} \right] \quad (5.14)$$

$$= \frac{2eE_J}{\hbar} \left[\cos \phi, \frac{\partial}{\partial\phi} \right] \quad (5.15)$$

$$= \frac{2e}{\hbar} E_J \sin \phi. \quad (5.16)$$

The differential form for the number operator is useful – can apply Hamilton's equations of motion directly to arrive at the first and second Josephson relations.

5.1.2 Second Josephson Relation

Let us add a potential term to the Hamiltonian by producing a fixed voltage drop across the junction by applying an external electric field, $H_T = -E_J \cos \phi - 2eV\hat{n}$. By Hamilton's equations of motion where we treat \hat{n} as the position variable and $\hbar\phi$ as the momentum variable, we have, $\frac{d\hat{n}}{dt} = \frac{1}{\hbar} \frac{\partial H}{\partial\phi}$ and $\frac{d\phi}{dt} = -\frac{1}{\hbar} \frac{\partial H}{\partial\hat{n}}$. (In our picture of the junction discussed above, the momentum $\hbar k$ is $\hbar\phi$, where the lattice constant $a = 1$.) Then, Hamilton's equations of motion yield: $2eV = -\frac{\partial H}{\partial\hat{n}} = \hbar \frac{d\phi}{dt}$.

Solving for V gives,

$$V = \frac{\hbar}{2e} \frac{d\phi}{dt} \quad (5.17)$$

which is the second Josephson relation.

Note, using the equations of motion we can immediately recover the first Josephson relation: The current operator is $\hat{I} = 2e \frac{d\hat{n}}{dt} = \frac{2e}{\hbar} \frac{\partial H}{\partial \phi} = \frac{2e}{\hbar} E_J \sin \phi$.

5.1.3 Consequences of the Relations

Here we combine the Josephson relations and find some interesting effects. First, we will see how the junction acts as a DC to AC converter, then we will see how the junction acts as a nonlinear inductor.

The first and second Josephson relations give us equations of motion in voltage and current: $I(t) = I_c \sin \phi(t)$ and $V(t) = \frac{\hbar}{2e} \frac{d\phi(t)}{dt}$. If we apply a constant DC voltage, V_0 , across the junction, integrate the second Josephson relation and plug it into the first relation, then we get an AC current, $I(t) = I_c \sin(\frac{2e}{\hbar} V_0 t)$. By applying a DC voltage across the junction we can generate an AC current.

Now, we can see how the Josephson junction acts as a nonlinear inductor. Differentiation of the first Josephson relation gives $\dot{I} = I_c \dot{\phi} \cos \phi$. Observing that $V = L\dot{I}$ and plugging V into the first Josephson relation yields a nonlinear inductance, $L(\phi) = (I_c \frac{2e}{\hbar} \cos \phi)^{-1} = \left(E_J \left(\frac{2e}{\hbar} \right)^2 \cos \phi \right)^{-1}$. As $\phi \rightarrow \frac{\pi}{2}$, the nonlinear Josephson inductance reaches a maximum, and as $\phi \rightarrow 0$, we find $L_J = \left(\frac{2e}{\hbar} I_c \right)^{-1} = \left(\left(\frac{2e}{\hbar} \right)^2 E_J \right)^{-1}$ which is the inductance of the Josephson junction. (Also note the flux quantum which we introduced in Sec. 5 is formed from the square root of the product of the Josephson energy and the Josephson inductance

that we derived here: $\Phi_0 = 2\pi\sqrt{E_J L_J}$.

5.2 The Transmon and Fluxonium

Different superconducting circuits are designed to mitigate the effects of different types of environmental noise. For example, fluctuations in charge cause fluctuations of the qubit transition energies. We know from the analysis of the quantum harmonic oscillator in Sec. 5 that decreasing the impedance of circuit spreads out the wavefunction in the charge variable – a low impedance causes the circuit to be less sensitive to the effects of charge fluctuations. In this section, we briefly introduce two superconducting circuits which have small and large impedances, respectively: the transmon and fluxonium.

The transmon is a Cooper pair box (i.e. the first superconducting version of an artificial atom) [93] shunted by a large capacitor, see Fig, 5.2. This large shunt capacitor decreases the charging energy of the circuit, E_C . For a derivation of the charging energy, see [6, 7]. The ratio of the Josephson energy to the charging energy (E_J/E_C) is set to a sweet spot which suppresses the circuit's sensitivity to charge noise and increases the circuit's coherence time [7].

Relating this back to the LC oscillator, it is easy to see how increasing the capacitance reduces sensitivity to offset charge noise. The charging energy of the circuit, E_C , is inversely proportional to the circuit capacitance. By increasing the capacitance, the characteristic impedance of the circuit is reduced causing the zero point fluctuations in phase to become small and the fluctuations in charge to become large. The wavefunction spreads out in charge, so the circuit is less sensitive to fluctuations in charge.

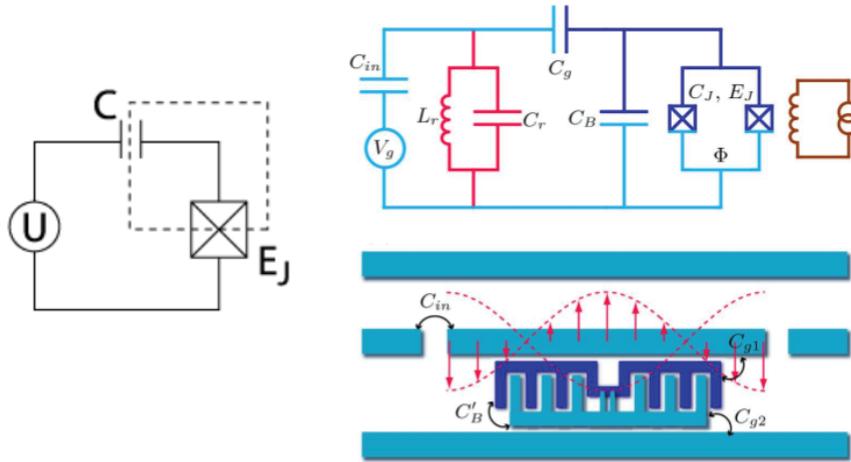


Figure 5.2: Cooper pair box (left) and transmon circuit (right). The transmon is a modification of the Cooper pair box, where the capacitor is large and the Josephson junction is flux-tunable. Figure adapted from [6] and [7].

The flux variable, on the other hand, is well-defined. The transmon is sensitive to flux noise, and it a prominent noise source when variable magnetic field sources are used to control the qubit. Recent efforts to improve the transmon include replacing flux control with voltage control [94–96].

There is a tradeoff between reducing the charge sensitivity of the circuit and the anharmonicity of the energy level transitions. Sufficient anharmonicity is required for accurately controlling transitions between the lowest two energy levels. The large capacitance of the transmon reduces the anharmonicity of the energy levels compared to the Cooper pair box. Fortunately, this reduction of anharmonicity is linear in the charging energy while the dispersive effects due to charge offsets decrease exponentially.

There is a sweet spot of this tradeoff between the charge sensitivity of the circuit and the anharmonicity, and it is called the “transmon regime.” In the transmon regime, the ratio of the Josephson energy to charging energy E_J/E_C is about 50. Here, effects of offset charges are minimal and the anharmonicity is large enough

to avoid unwanted transitions between higher energy levels.

The transmon has an additional feature to the Cooper pair box – it is flux-tunable. The junction of the Cooper pair box is replaced with a flux-tunable junction. The flux-tunable junction is created by two parallel junctions that together form a loop through which an externally applied flux can penetrate. The large capacitance is achieved by a inter-digital capacitor on either side of the flux-tunable junction. Strong qubit-cavity coupling is achieved by coupling the junction to a resonator where the electric field oscillating inside is the greatest, see Fig. 5.2.

Fluxonium is the opposite of the transmon in the sense that the characteristic impedance of a Cooper pair box is increased using a large inductor (instead of being decreased by using a large capacitance, as in the transmon case). Fluxonium is a Cooper pair box shunted by a large inductor, and the idea is to suppress sensitivity to flux noise. Suppression of flux noise can be understood intuitively from the LC oscillator picture. By increasing the circuit inductance, and thus the circuit's characteristic impedance, fluctuations in charge become suppressed causing the wavefunction to spread out in phase.

A large inductance is hard to create. Several approaches for creating a large inductance have been to increase the kinetic inductance by using a string of Josephson junctions [9] or a small nanowire [30, 31], and to increase the geometrical inductance by winding a large wire [29].

5.3 Superinductor

A large inductance is required in several superconducting circuit designs such as the $0 - \pi$ qubit [26] and fluxonium [9, 27, 28].

There are several ways to make such a large inductance [9, 29, 31]. One way is to string together a chain of Josephson junctions. This design uses the idea of kinetic inductance, where the kinetic inductance is found from equating the kinetic energy of a Cooper pair to the energy stored in the inductor. Each Josephson junction in the array is constructed so that it is in the transmon regime (for a discussion see 5.2). This is the design currently employed in [9], see Fig. 5.3. This string of Josephson junctions can be called a superinductor.

There are several advantages to creating a large inductance this way. This construction allows for a large impedance while simultaneously avoiding impedance limitations from accumulated parasitic capacitance (such as those accumulated by using a large wire) [6], [9]. This construction also allows for superinductors to have large impedance while remaining insensitive to offset charge fluctuations. This may sound counter-intuitive – we know that in an LC oscillator, a large characteristic impedance reduces zero-point fluctuations in charge and increases the circuit’s sensitivity to charge fluctuations. But, by choosing Josephson junction parameters in the transmon regime (see Sec. 5.2) and increasing the number of array junctions, superinductors can have large impedance while remaining insensitive to offset charge fluctuations.

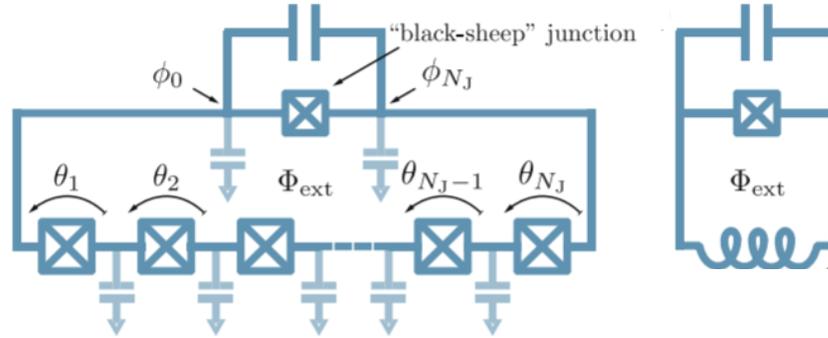


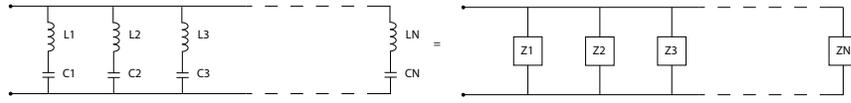
Figure 5.3: The fluxonium circuit contains a large superinductor composed of many Josephson junctions (left). The junctions collectively act as a large inductor (right). Figure adapted from [8].

5.4 Harmonic Analysis and Black Box Quantization

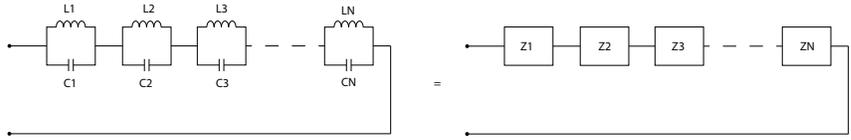
Some novel qubit designs have a large number of degrees of freedom [9, 32]. Modelling such circuits requires approximations. Many large circuits can be approximated by a harmonic circuit where the junction nonlinearity is added as a perturbation to the harmonic Hamiltonian. Alternatively, numerical techniques can solve the full nonlinear problem. In this section, we introduce a method which solves the harmonic part of the circuit exactly and includes the nonlinear part perturbatively. This method is also known as “black box quantization.”

First let us consider a circuit consisting only of capacitors and inductors with no power sources. Since there is no resistance, the impedance is completely imaginary. Such circuits are called reactive.

Foster’s theorem [83] states that for a reactive circuit, the impedance increases strictly monotonically with frequency. This means that the impedance as a function of frequency is a series of alternating zeros and poles. Such alternation suggests that any passive, reactive circuit is equivalent to another if it has the identical



(a) A Foster equivalent circuit modelled as series LC circuits in parallel.



(b) Dual Foster equivalent circuit to (a) modelled as parallel LC circuits in series.

Figure 5.4: Foster network model of passive, reactive circuits. Reactive circuits can be characterized completely by a single port measurement – one hooks up the end leads (open terminals pictured on the left of each circuit) to a infinite-impedance current source or a zero-impedance voltage source. If the circuit is hooked up to a zero-impedance voltage source, then the model best describing the circuit is Fig. 5.4a. The voltage source shorts the first LC loop allowing for its oscillation frequency and characteristic impedance to be measured. If the circuit is hooked up to an infinite-impedance current source, then the best model describing the circuit is Fig. 5.4b, a model of LC oscillators in series. With an infinite-impedance source, a false measurement due to the formation of the additional loop around the entire circuit is avoided.

impedance diagram. This stems from the condition that such circuits can be modeled as coupled LC oscillators, each having its mode frequencies and characteristic impedances given by the corresponding zeros and poles. In other words, these circuits can be modeled by coupled LC oscillators in series or series LC oscillators in parallel, where each LC section of the circuit is described by its characteristic frequency and characteristic impedance, see Fig. 5.4

In Fig. 5.4a, the passive network model consists of parallel (ladder) configuration of inductors and capacitors in series. This is the best model of an admittance, where the total admittance of the circuit is the sum of admittance of the individual rungs (i.e. $Y_T = Y_1 + Y_2 + \dots + Y_N$). In this model, the poles of the admittance correspond to the resonance frequencies.

In Fig. 5.4b, the passive network model consists of a series combination of parallel inductors and capacitors. This is the best model of an impedance, where the total impedance of the circuit is the sum of the impedance of the individual parallel circuits (i.e. $Z_T = Z_1 + Z_2 + \dots + Z_N$). In this model, the poles of the impedance correspond to the resonance frequencies.

The model that represents the circuit, either Fig. 5.4a or Fig. 5.4b, is dependent on the drive source. This is described in detail in Fig. 5.4.

Relating this model back to the quantum LC oscillator, we see that we can obtain the harmonic excitations of any superconducting circuit simply by finding its zeros and poles. To quantize the circuit, we also need the characteristic impedance since the zero-point fluctuations are written in terms of fundamental constants and the characteristic impedance.

We can find the characteristic impedance using the slope of the frequency response near resonance. For this, we take the inverse of the function and look at the zeros (i.e. for an admittance diagram, where the resonance frequencies correspond to the poles, we invert the diagram and instead look at the zeros which correspond to the resonance frequencies.) Taking for example, a model of parallel LC circuits in series (where we look at the poles of the impedance), we get the following admittance,

$$\frac{1}{Z} = \frac{1}{i\omega L} + i\omega C \quad (5.18)$$

$$= \frac{-i}{\omega} \frac{1}{Z_0 \sqrt{LC} + i\omega \frac{\sqrt{LC}}{Z_0}} \quad (5.19)$$

$$= \frac{i}{Z_0} \left(\frac{\omega}{\omega_0} - \frac{\omega_0}{\omega} \right) \quad (5.20)$$

where we see that the zero corresponds to the resonance frequency, also see Fig. 5.5. Taking the derivative of the admittance with respect to frequency and eval-

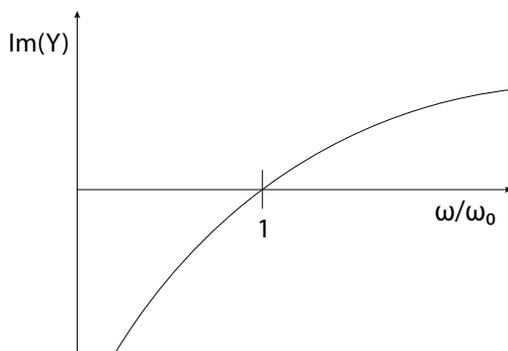


Figure 5.5: Example of the admittance of a parallel LC circuit. For a parallel LC circuit, the poles of the impedance correspond to the resonance frequency. Since it is often more convenient to look at the zero instead of the pole to find the resonance frequency, we have inverted the function and have plotted the admittance instead of the impedance. The slope of the admittance near resonance yields the characteristic impedance.

uating it at the resonance frequency gives, $\frac{\partial Y}{\partial \omega} \Big|_{\omega=\omega_0} = \frac{2i}{\omega_0 Z_0}$. So we have that the characteristic impedance is,

$$Z_0 = \frac{2i}{\omega_0 \frac{\partial Y}{\partial \omega} \Big|_{\omega=\omega_0}} \quad (5.21)$$

We can fully quantize any reactive quantum circuit using the characteristic impedances of its harmonic modes – each harmonic mode corresponds to a unique LC harmonic oscillator having its own characteristic impedance. The Hamiltonian is written as a sum over all of the harmonic modes in the circuit, $H_0 = \sum_m \Omega_m \hat{a}_m^\dagger \hat{a}_m$. The phase operator is given by the sum of the phase operators over the modes, $\hat{\Phi} = \sum_m \Phi_{ZPF}^{(m)}(\hat{a}_m + \hat{a}_m^\dagger)$ where $\Phi_{ZPF}^{(m)} = \sqrt{\frac{\hbar Z_m}{2}}$. The canonically conjugate variable, the charge operator is $\hat{Q} = -i \sum_m Q_{ZPF}^{(m)}(\hat{a}_m - \hat{a}_m^\dagger)$ where $Q_{ZPF}^{(m)} = \sqrt{\frac{\hbar}{2Z_m}}$.

5.5 Simulation Techniques

Circuits having a large number of degrees of freedom have a large Hilbert space and can be difficult to simulate. Simulations of large superconducting circuits often require approximations to the Hamiltonian, an implementation of numerical techniques, or both.

One method to simulate a large circuit is to use “black box quantization”, discussed in Sec. 5.4. Here one approximates the Josephson junctions in the circuit as linear inductances. With this approximation, the Hamiltonian becomes harmonic. One then solves the generalized eigenvalue problem for a system of coupled harmonic oscillators. The conjugate phase and charge variables found in the same way discussed in Sec. 5.4. The junction non-linearity is later added as a perturbation to obtain a first-order correction to the energy.

Let us discuss this method a little more in detail. We first recall from Sec. 5.4 that any passive LC circuit can be fully quantized by its equivalent circuit – the conjugate variables are found by the summing over the circuit modes, where each mode is represented by its harmonic frequency and characteristic impedance. This is a good motivation for solving a harmonic approximation to the circuit. If the circuit is weakly anharmonic, where the Josephson junctions are weakly nonlinear (for example, the circuit could be in the transmon regime, see Sec. 5.2), a harmonic approximation is a good first approximation to the Hamiltonian. We accomplish this by approximating the nonlinear inductance of each Josephson junction by a linear inductance, $(\frac{\hbar}{2e})^2 \frac{1}{E_J}$, see Sec. 5.1.3. Since the junction non-linearity is assumed to be small, we can later add the nonlinear inductive terms as a perturbative correction. The result of approximating the Josephson junctions as linear

inductors is a harmonic circuit composed only of inductors and capacitors.

The number of LC oscillators in the equivalent circuit is equal to the number of modes of the circuit (see Sec, 5.4), so we can solve a generalized eigenvalue problem for the coupled harmonic oscillators and obtain the resonances and characteristic impedances of the circuit.

The generalized eigenvalue problem is formed using the inverse inductance matrix and the capacitance matrix. Both the capacitance and inverse inductance matrices are adjacency matrices where the nodes of the circuit (islands) correspond to the locations. The circuit ground is included as a node. The eigenvalues of the generalized eigenvalue problem are the square of the energies of the modes. The matrix of eigenvectors corresponds to the phase matrix, where each eigenvector gives the superconducting phase of each node for a given circuit mode.

One can easily obtain the current by taking the derivative of the phase eigenvector for a particular mode with respect to the nodes. The current flow across the circuit is interesting. It shows the character of the mode, and it reveals places where the current builds up and might be most sensitive to charge offsets.

If we would like to see how the modes behave at different values of flux, we can replace the non-linear inductance of the Josephson junctions with its inductance at a certain flux value and solve the corresponding generalized eigenvalue problem. Repeating this at each specified value of flux gives a composite energy spectrum which shows the mode structure as a function of applied flux.

The harmonic spectrum we obtain this way is in a form which can often be compared directly with experimental values reported in the literature. For weakly nonlinear circuits, the harmonic spectrum should resemble the measured spectrum,

and upon the inclusion of a perturbative term to account for the junction non-linearity, it should compare well with the measured spectrum.

We are now in a position to include the junction non-linearity in the Hamiltonian. A perturbative term that includes higher order terms in the junction non-linearity can be easily added to the Hamiltonian. The cosine of the phase operator is expanded to the desired order. The unperturbed part of the Hamiltonian is a harmonic oscillator written in terms of the squares of the conjugate variables weighed by the Josephson and charging energy matrices. The higher order terms of the phase operator are used in the first order correction to the energy of the harmonic oscillator. The higher order terms can be expanded out in terms of its creation and annihilation operators, (i.e $\hat{\Phi} = \sum_m \Phi_{ZPF}^{(m)}(\hat{a}_m + \hat{a}_m^\dagger)$ for mode m), which reveals the self- and cross-Kerr shift terms. The self-Kerr shift of the first mode is often referred to as the “anharmonicity” of the circuit [7].

This type of harmonic analysis is done in open-source software [87], where the cosine of the phase operator is expanded to the order specified by the user. For a weakly nonlinear circuit, the contribution of the junction non-linearity shifts the energy spectrum slightly from the harmonic approximation.

This method is acceptable for analyzing many circuits. The energy spectrum found from using this method can resemble the measured energy spectrum particularly well when the Josephson junctions are operated in the transmon regime. However, near half-integer values of the flux quantum (see see analysis of the LC oscillator in Sec. 5), this approximation can break down. Also, the effects of offset charges were assumed to be small. To calculate the spectrum accurately near certain critical values of flux, and to see the effects of offset charges, other numerical methods need to be employed.

5.6 Tensor Networks for Qubit Simulation

Modeling the exact Hamiltonian for quantum circuits having many degrees of freedom can be difficult. It could be to our advantage to find methods in the literature for modeling many-body systems and apply them to our problem of modeling large superconducting circuits. In this section, we describe a tensor network model of superconducting circuits which is fairly useful for finding the low-energy spectrum of quantum circuits having a large number of degrees of freedom.

First, let us start by finding the similarities between quantum circuit Hamiltonians and ones modeled in condensed-matter – a mature field employing many advanced numerical techniques.

Many superconducting circuits are composed of capacitors and Josephson junctions. We would like to find a Hamiltonian that models a large collection of these quantum circuits. To find a basis, we can make an assumption that the circuit is designed so that the phase variable is well-confined: this occurs when the circuit is in the “transmon regime,” see Sec. 5.2. If the phase variable is well-confined, the phase itself becomes the quantized variable instead of the cosine of the phase, for a great discussion see [6]. In the transmon regime, the wavefunction is no longer periodic and the charge is well-defined on each node. A good choice for a basis in this case could be the charge number basis, which counts the number of Cooper pairs on a node.

We can make this even better by observing that most circuits are floating. In this case, the total charge of the circuit is conserved. We can place the total charge on the first node and set it to zero, removing the first degree of freedom. On the

remaining degrees of freedom, instead of using the charge number basis, we can use the junction basis, where we count the difference in number of Cooper pairs between each node, i.e. $|m\rangle$ where m is the difference in Cooper pairs across the nodes, see Sec. 5.1. A discussion of this coordinate transformation can be found in the first appendix of [8].

Floating circuit composed of capacitors and Josephson junctions have a common form for the Hamiltonian,

$$H = \sum_{i<j}^{N-1} E_C^{(ij)} \hat{n}_i \hat{n}_j + \sum_{i=1}^{N-1} E_J^{(i)} - \frac{1}{2} \sum_{i=1}^{N-1} E_J^{(i)} (\hat{h}_i^\dagger + \hat{h}_i), \quad (5.22)$$

where there are N nodes in the circuit, the operator \hat{n} is the number operator, see Sec. 5.1.1, and \hat{h} is the hopping operator, see Sec. 5.1.1, E_C is the charging energy, and E_J is the Josephson junction energy. The first term in the Hamiltonian is the sum of the charging energies over each pair of nodes of the circuit, which are similar to superconducting “islands,” see [8]. The charging energy term is the source of long-range interactions in the circuit, which could be problematic in the simulation. The second term is the sum of all of the Josephson junction energies, and it is an offset term and gives positive energy eigenvalues. The third term in the Hamiltonian is the energy it takes for Cooper pairs to hop across the junctions.

The form of the Hamiltonian and choice of basis is suggestive – we may be able solve the Hamiltonian using standard techniques from condensed matter physics. This Hamiltonian looks like some popular models that have been solved using a density matrix renormalization group method, DMRG [10, 12, 97, 98].

DMRG is a tensor network method that takes in an input matrix product operator (MPO) form of the Hamiltonian and outputs a matrix product state (MPS) form of the wavefunction that minimizes the energy. For a great review,

see [10]. Additional, yet instructive descriptions of the DMRG algorithm can be found in [12, 99].

The bond dimension of the tensor network represents the product of tensors in the network. The larger the bond dimension, the larger the contraction between tensors. Since the charging energy can cause long-range interactions between nodes, larger MPS bond dimensions might be necessary. Two site DMRG method is a viable candidate for large superconducting circuits [10], since the method allows the bond dimension to grow during the algorithm.

It would be exciting if we could find the lowest-lying energies of superconducting circuits using DMRG or other tensor network methods. To date, there are several examples in the literature employing DMRG techniques for quantum circuits: current mirror [91], the superinductor [11], and fluxonium [8]. Typically the low-energy spectrum is desired, and to obtain energies using DMRG beyond the ground state, a penalty term is added [12]. In [8], an extra index was added onto the MPS wavefunction, so that the optimized solution contained a collection of the low-energy eigenvalues. The DMRG technique for circuits accurately produces the low energy spectrum at critical flux values, it allows for the calculation of energy dispersion as a function of offset charge, and even the decoherence effects of coherent quantum phase slips [8].

5.7 MPS-MPO-MPS Form for DMRG

In this section, we explain how set up the tensor network form of a superconducting circuit to model using DMRG.

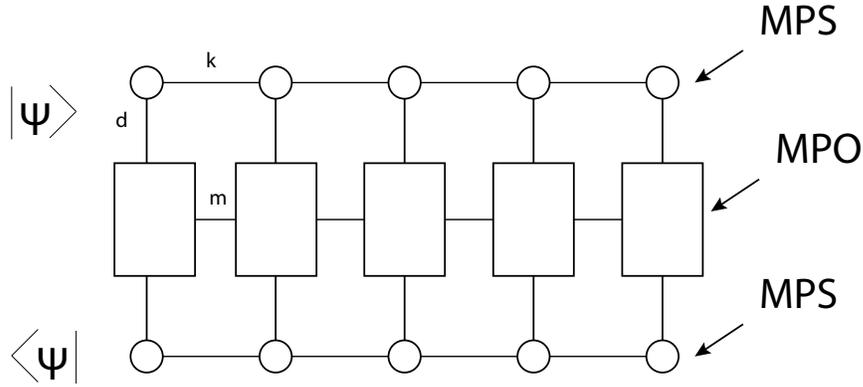


Figure 5.6: MPS-MPO-MPS tensor network. The local Hilbert space dimension is d , and the bond dimensions of the MPO and MPS are m and k , respectively.

DMRG uses a tensor network like the one shown in Fig. 5.6. The DMRG algorithm uses the MPO to find the MPS that minimizes the contraction value of the tensor network. If we let the MPO in Fig. 5.6 be the MPO form of the Hamiltonian of our quantum circuit, then the MPS output from DMRG will be the ground state wavefunction and the contraction value of the network will be the ground state energy of the quantum circuit.

To start, let us see if we can recast the Hamiltonian into MPO form. Every tensor in the MPO has physical indices and virtual indices. The physical indices correspond to operators that act locally on one node. The virtual indices represent contractions of the MPO.

To create an MPO from the Hamiltonian, we want to break down the Hamiltonian into a tensor product of operators, where each tensor corresponds to one node. The MPO tensor should be set up such that after contraction of the MPO (taking the tensor product of the operators), we arrive at the original algebraic form of the Hamiltonian.

In the Hamiltonian, Eqn. 5.22, we have the operators \hat{h} and \hat{n} , which are $d \times d$

sized matrices where d is the dimension of the local Hilbert space. Since these operators are matrices that act locally on the physical space, they each have two indices with length d that correspond to physical indices.

The local Hilbert space dimension is cut off by the minimum and maximum number of Cooper pairs we allow to hop along the nodes of the circuit, i.e. if we allow 10 Cooper pairs to hop across the junctions from one node to another, our local Hilbert space is 21 dimensional (-10 to 10), and we say that the cutoff for that node is 10.

We choose the cutoff value. It takes energy for Cooper pairs to hop along the circuit, so we would like to choose a cutoff that is large enough to capture the dynamics of our problem but small enough that the Hilbert space does not bog down the simulation. As the cutoff is increased, the simulation becomes more accurate, but it takes longer. A good choice of a cutoff value can be found by running multiple simulations at increasingly larger cutoffs and looking for a change in energy which is below our desired error tolerance.

Each i th tensor in the MPO has virtual index m_i , called the bond dimension. The length of the index is dependent on the Hamiltonian. In Sec. 5.7.1, we will find a specific MPO and its bond dimension by example.

Each i th tensor in the MPS has physical indices with dimension d of the local Hilbert space and virtual indices with dimension k_i . The physical indices of the MPS contract with the physical indices of the MPO, and so they have the same dimension, d . The dimension of the virtual indices of the MPS is typically called the bond dimension of the MPS. The wavefunction on the bottom side of the MPO in Fig. 5.6 is the complex conjugate of the MPS on top.

The contraction value of the MPS-MPO-MPS tensor network in Fig. 5.6 (the scalar resulting from the tensor product taken by a sum over the virtual and physical indices) corresponds to an energy – it is the expectation value of the Hamiltonian (represented by the MPO) in some quantum state (represented by the MPS). DMRG finds the MPS which minimizes the contraction value of the MPS-MPO-MPS tensor network, so the output MPS from DMRG is the ground state wavefunction and the contraction value of the tensor network is the ground state energy.

5.7.1 Example MPO Construction for a Superconducting Network of Capacitors and Josephson Junctions

In this section, we show by example how to construct an MPO for a superconducting circuit which is composed of capacitors and Josephson junctions. This section is adopted from [11].

Consider the circuit in Fig. 5.7. The circuit has three Josephson junctions two large end capacitances to ground, and two small parasitic capacitances to ground, i.e. this is a minimal superinductor circuit having open boundaries, see [9]. There are four nodes in the circuit. The circuit is floating, so the total charge of the circuit is conserved. Here each Josephson junction is represented by a nonlinear inductance (represented by an X) and a capacitance in parallel. This is one way to represent the Josephson junctions. Each junction has a capacitance and a nonlinear inductance. The capacitance of each junction can be represented separately from its inductive element by a capacitor in parallel. The parallel capacitors contribute to the charging energy, and the nonlinear inductances contribute to the Cooper

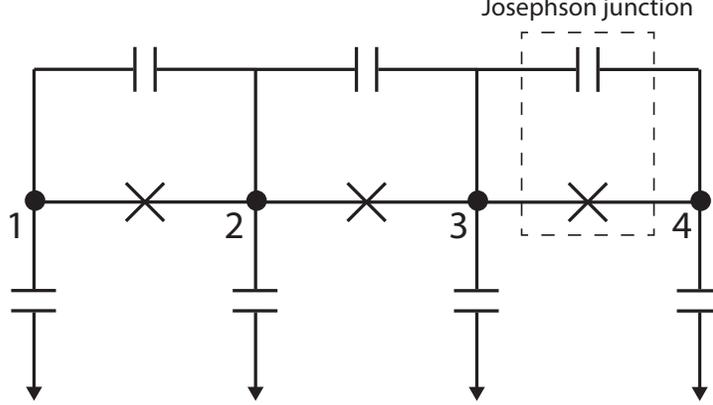


Figure 5.7: Example superconducting circuit composed of capacitors and Josephson junctions. Each Josephson junction is represented by a nonlinear inductance and capacitance in parallel. The Hamiltonian of the circuit is given in Eqn. 5.23, and the corresponding MPO is given in Eqn. 5.24. This circuit is a minimal form of superinductor [9].

pair hopping energy.

The Hamiltonian is a sum of the kinetic and potential energies, $H = T + U$. The potential energy is often chosen to be the energy stored in the capacitors $U = \sum_{i < j}^{N-1} E_C^{(ij)} \hat{n}_i \hat{n}_j$ where \hat{n}_i is the Cooper pair number operator on the i th node in the network. The term $E_C^{(ij)}$ is the charging energy matrix that is determined by the configuration of the capacitors in the network – it is the inverse capacitance matrix. This matrix is often the source of long-range interactions in the circuit. (If these are exponentially-decaying interactions, the DMRG simulation should converge well, see [12,98].) The kinetic energy is the energy it takes for the Cooper pairs to hop across the junctions, $T = -\sum_i E_J^{(i)} \cos \theta_i$. The $E_J^{(i)}$ term is the Josephson junction energy for the i th Josephson junction. By expanding $\cos \theta_i$, shown in Sec. 5.1.1, we find the Hamiltonian given by Eqn. 5.22. The Josephson junction energy can be different for each Josephson junction due to differences in fabrication. Although different E_J are easily accounted for in the MPO (see Sec.

5.8), in this example we make the simplification that similar junctions have the same E_J .

Including these simplifications, the Hamiltonian of the circuit in Fig. 5.7 is,

$$H = E_C^{(11)}\hat{n}_1\hat{n}_1 + E_C^{(12)}\hat{n}_1\hat{n}_2 + E_C^{(13)}\hat{n}_1\hat{n}_3 + E_C^{(22)}\hat{n}_2\hat{n}_2 + E_C^{(23)}\hat{n}_2\hat{n}_3 + E_C^{(33)}\hat{n}_3\hat{n}_3 + (N-1)E_J + E_J(\hat{h}_1^\dagger + \hat{h}_2^\dagger + \hat{h}_3^\dagger + \hat{h}_1 + \hat{h}_2 + \hat{h}_3). \quad (5.23)$$

We can find a form for the MPO of the Hamiltonian,

$$H = \left(E_C^{(11)}\hat{n}_1^2 - \frac{1}{2}E_J(\hat{h}_1^\dagger + \hat{h}_1) \quad \hat{n}_1 \quad 1 \right) \otimes \begin{pmatrix} 1 & 0 & 0 \\ E_C^{(12)}\hat{n}_2 & E_C^{(13)} & 0 \\ E_C^{(22)}\hat{n}_2^2 - \frac{1}{2}E_J(\hat{h}_2^\dagger + \hat{h}_2) & E_C^{(23)}\hat{n}_2 & 1 \end{pmatrix} \otimes \begin{pmatrix} 1 \\ \hat{n}_3 \\ E_C^{(33)}\hat{n}_3^2 - \frac{1}{2}E_J(\hat{h}_3^\dagger + \hat{h}_3) + 3E_J \end{pmatrix}. \quad (5.24)$$

We have included subscripts on the operators \hat{n} and \hat{h} for clarity. We do not need to do this since subscripts are implied by the tensor product (i.e. $1 \otimes \hat{n} = \hat{n}_2$).

5.8 Understanding the MPO: Some MPO Constructions for Superconducting Circuits

There are an infinite number of ways to write the MPO form of a Hamiltonian. In this section, we describe a few of these ways to construct the MPO for various superconducting circuit Hamiltonians.

We use the triangular form since it is straightforward. This method does not always provide the most compact representation for the MPO. There are compres-

sion algorithms that can reduce the bond dimension after a representation is found using the triangular form. One excellent example of a compression algorithm is in [12]. This algorithm can reduce the bond dimension by finding another form of the MPO, or it can reduce the bond dimension by finding an approximate MPO to an accuracy specified by the user.

Superconducting circuits often have a term in the Hamiltonian which consists of a sum of operators, each acting on a single site, i.e.

$$H = \sum_1^L X_i, \quad (5.25)$$

where each X operator acts on a site i where there are L sites. We want to represent H as a tensor product $H = M^{(1)} \otimes \dots M^{(i)} \otimes \dots M^{(L)}$, where each operator $M^{(i)}$ contains elements that only act on site i . We can write one operator in the form,

$$M^{(i)} = \begin{pmatrix} I & 0 \\ X_i & I \end{pmatrix} \quad (5.26)$$

where X_i operates on the i th site. The tensor product of the operator which operates on site i with the operator that operates on site $M(i+1)$ gives,

$$M^{(i)} \otimes M^{(i+1)} = \begin{pmatrix} I & 0 \\ X & I \end{pmatrix} \otimes \begin{pmatrix} I & 0 \\ X & I \end{pmatrix} = \begin{pmatrix} I \otimes I & 0 \\ X \otimes I + I \otimes X & I \otimes I \end{pmatrix},$$

where we have dropped the subscript i since the site on which the operator acts is implied by the tensor product.

The Hamiltonian in Eqn. 5.25 has open boundary conditions, so the first and last operators in the tensor product are vector-like: $M^{(1)} = \begin{pmatrix} X & I \end{pmatrix}$ on the left, and $M^{(L)} = \begin{pmatrix} I \\ X \end{pmatrix}$ on the right.

We have a similar pattern for a Hamiltonian of the form $H = \sum_i X_i Y_{i+1}$, where the $M^{(i)}$ operators have the form,

$$M^{(i)} = \begin{pmatrix} I & 0 & 0 \\ Y & 0 & 0 \\ 0 & X & I \end{pmatrix}. \quad (5.27)$$

The tensor product of nearest-neighbor operators is,

$$M^{(i)} \otimes M^{(i+1)} = \begin{pmatrix} I & 0 & 0 \\ Y & 0 & 0 \\ 0 & X & I \end{pmatrix} \otimes \begin{pmatrix} I & 0 & 0 \\ Y & 0 & 0 \\ 0 & X & I \end{pmatrix} = \begin{pmatrix} I \otimes I & 0 & 0 \\ Y \otimes I & 0 & 0 \\ X \otimes Y & I \otimes X & I \otimes I \end{pmatrix}.$$

Again, one chooses the form of the boundary operators to match the specified boundary conditions.

Below is a MPO form for a Hamiltonian having unique interaction energies, $H = J_1 \sum_i Z_i Z_{i+1} + J_2 \sum_i Z_i Z_{i+2}$. The zeros are converted to blanks for clarity and the explicit denotation of the tensor product (\otimes) is dropped for compactness.

$$M^{(i)} = \begin{pmatrix} I & & & \\ Z & & & \\ & I & & \\ & J_1 Z & J_2 Z & I \end{pmatrix} \quad (5.28)$$

$$\begin{aligned}
M^{(i)} \otimes M^{(i+1)} \otimes M^{(i+2)} &= \begin{pmatrix} I & & & \\ Z & & & \\ & I & & \\ & J_1 Z & J_2 Z & I \end{pmatrix} \begin{pmatrix} I & & & \\ Z & & & \\ & I & & \\ & J_1 Z & J_2 Z & I \end{pmatrix} \begin{pmatrix} I & & & \\ Z & & & \\ & I & & \\ & J_1 Z & J_2 Z & I \end{pmatrix} \\
&= \begin{pmatrix} II & & & \\ ZI & & & \\ IZ & & & \\ J_1 ZZ & J_2 ZI + J_1 IZ & J_2 IZ & II \end{pmatrix} \begin{pmatrix} I & & & \\ Z & & & \\ & I & & \\ & J_1 Z & J_2 Z & I \end{pmatrix} \\
&= \begin{pmatrix} III & & & \\ ZII & & & \\ IZI & & & \\ J_1 ZZI + J_2 ZIZ + J_1 IZZ & J_2 IZI + J_1 IIZ & J_2 IIZ & III \end{pmatrix}
\end{aligned}$$

5.9 Understanding the MPS: the AKLT state

Now that we understand how to create an MPO, we would like to have a better understanding of the MPS. In this section, we explain matrix product states through example. Here we use the AKLT state, which is the exact solution of the Heisenberg spin model by Affleck, Kennedy, Lieb, and Tasaki [100]. This state has interesting properties and can be used as a resource state for quantum computation [101–103].

The AKLT state is a very good example of a MPS. A MPS is the representation of a wavefunction as a product of matrices. When one hears the term product state, the product of vectors may come to mind. The MPS is a product of matrices rather than a product of vectors. The extra dimension of a matrix allows for entanglement

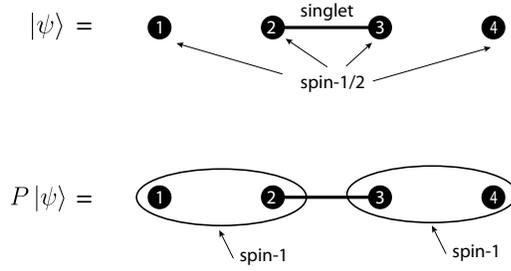


Figure 5.8: An example spin-1 AKLT state formed from four spin-1/2 particles. The state $P|\psi\rangle$ is formed by contracting neighboring pairs into a singlet, represented by the solid bond. The state is then symmetrized by projecting the remaining neighboring spins into triplets, represented by the circular bonds.

in the wavefunction while maintaining a product state notation.

The spin-1 AKLT state is formed by contracting neighboring pairs of spin-1/2 particles into singlets and projecting the particles between each pair into the spin-1 space. See Fig. 5.8.

The spin-1 AKLT state has the property that no two neighboring sites are in a spin-2 state. We see this from the state $|\psi\rangle$ in Fig. 5.8: the two spin-1/2 particles in the middle of the chain are contracted to form a singlet, and the remaining spin-1/2 particles could form a triplet or a singlet (spin-2 is not possible). This picture also show us that there is a 4-fold degeneracy, and up to this 4-fold degeneracy, the AKLT state is unique.

Let us explicitly construct the AKLT state shown in Fig. 5.8. We use the convention where spin up is $|0\rangle$ and spin down is $|1\rangle$ so the singlet state is, $|s\rangle = \frac{1}{\sqrt{2}} (|01\rangle - |10\rangle)$. The projector onto the spin-1 configuration is,

$$P = |00\rangle\langle 00| + \frac{1}{\sqrt{2}} (|01\rangle + |10\rangle) \frac{1}{\sqrt{2}} (\langle 01| - \langle 10|) + |11\rangle\langle 11|. \quad (5.29)$$

We denote the spin-1 states compactly,

$$|\tilde{+}\rangle = |00\rangle, \quad |\tilde{0}\rangle = \frac{1}{\sqrt{2}}(|10\rangle + |01\rangle), \quad |\tilde{-}\rangle = |11\rangle$$

where tildes are used so the triplet spin-1 state $|\tilde{0}\rangle$ is not confused with the spin-1/2 particles on the ends in state $|0\rangle$.

Let us start in the configuration where we have projected the center neighboring pair into the singlet state, i.e. $|\psi^{ab}\rangle = |a\rangle |s\rangle |b\rangle$. This corresponds to the configuration where the endpoint spins (denoted by $|a\rangle$ and $|b\rangle$) can be either spin up or spin down. If the two endpoint spins are both spin up, we have $|\psi^{00}\rangle = |0\rangle |s\rangle |0\rangle$. Applying the projection operator, Eqn. 5.29, to spins 1 and 2, and spins 3 and 4 gives $P_{1,2}P_{3,4}|\psi^{00}\rangle = -\frac{1}{\sqrt{2}}(|\tilde{+}\tilde{0}\rangle + |\tilde{0}\tilde{+}\rangle)$.

We can repeat this procedure for all four configurations of the endpoint spins and obtain the following table:

Contracted state	Projected state
$ \psi^{00}\rangle = 0\rangle s\rangle 0\rangle$	$P_{1,2}P_{3,4} \psi^{00}\rangle = -\frac{1}{\sqrt{2}}(\tilde{+}\tilde{0}\rangle + \tilde{0}\tilde{+}\rangle)$
$ \psi^{01}\rangle = 0\rangle s\rangle 1\rangle$	$P_{1,2}P_{3,4} \psi^{01}\rangle = \frac{1}{\sqrt{2}}(\tilde{+}\tilde{-}\rangle - \tilde{0}\tilde{0}\rangle)$
$ \psi^{10}\rangle = 1\rangle s\rangle 0\rangle$	$P_{1,2}P_{3,4} \psi^{10}\rangle = \frac{1}{\sqrt{2}}(\tilde{0}\tilde{0}\rangle - \tilde{-}\tilde{+}\rangle)$
$ \psi^{11}\rangle = 1\rangle s\rangle 1\rangle$	$P_{1,2}P_{3,4} \psi^{11}\rangle = \frac{1}{\sqrt{2}}(\tilde{0}\tilde{-}\rangle - \tilde{-}\tilde{0}\rangle)$

Table 5.1: The four possible states resulting from the contraction and projection in Fig. 5.8. The states in the left column correspond to the four configurations of the top state in Fig. 5.8, where the middle neighboring particles are contracted into a singlet state and the endpoint particles can be combinations of spin up and spin down. The second column corresponds to the projected state, where the remaining spin-1/2 particles on the endpoints are projected into the spin-1 space with the particles on neighboring sites, bottom of Fig. 5.8. This example can be generalized to longer 1D chains of spin-1/2 particles.

The states in this table suggest some other properties of the AKLT state. For example, a +1 spin ($|\tilde{+}\rangle$) can be followed by any number of 0 spins ($|\tilde{0}\rangle$) then a -1 ($|\tilde{-}\rangle$) spin, but it is never followed by a +1 spin. Similarly, a -1 spin is never

followed by a -1 spin. Additionally, the parity of the two spin-1/2 particles on the endpoints tells us what the total spin of the spin-1 chain is: (i) if there is odd parity, then the total spin of the spin-1 chain is 0 (we have an equal number of $|\tilde{+}\rangle$ and $|\tilde{-}\rangle$); (ii) if the parity is even and they are both spin down, then the total spin of the spin-1 chain is -1 (if the number of $|\tilde{+}\rangle$ in the chain is n , then we have $n + 1$ number of $|\tilde{-}\rangle$); (iii) and if the parity is even and they are both spin up, then the total spin of the spin-1 chain is +1 (if the number of $|\tilde{-}\rangle$ in the chain is n , then we have $n + 1$ number of $|\tilde{+}\rangle$).

Representing the AKLT State as a MPS

In this section, we show how to represent the AKLT state as a MPS. Fig. 5.9 shows the overall process for constructing the AKLT as a MPS with periodic boundary conditions.

The top figure, Fig. 5.9a, shows the auxiliary system and the operators explicitly: the solid circles correspond to the auxiliary spin 1/2 particles, a_i and b_i , which are contracted into singlet states (solid bonds) and projected into the spin-1 space (circles). The solid bonds between the auxiliary particles represent operators, $\Sigma_{a,b}$, that project the spin-1/2 particles into the singlet state. The circles between the auxiliary particles represent operators, $M_{a,b}^\sigma$, that project the particles into the physical spin-1 space.

Each operator can be written as a tensor, where the legs of the tensor correspond to the indices of the tensor. For example, the operator $M_{a_1 b_1}^{\sigma_1}$ has legs on the left and the right, a_1 and b_1 , respectively, that contract with the Σ operators on either side. The σ indices are left open-ended, and they can be contracted with an operator that acts on the spin degrees of freedom. When the operators are written

in this form, we arrive at Fig. 5.9b, which represents the diagram in Fig. 5.9a. Fig. 5.9b is the MPS of the AKLT state.

We can simplify the network in Fig. 5.9b by contracting the $M_{a,b}^\sigma$ tensors with the Σ tensors. The simplified network is shown in Fig. 5.9c. Each A tensor has a single index σ that originated from the M tensor, and the first and last A matrices are connected, showing the periodic boundary. In some cases the σ indices are written out explicitly in the diagram and protrude out from the A matrices.

Finally we sum over the σ indices and arrive at Fig. 5.9d. This form suggests a trace of the product of the A matrices.

Now let us write these steps out in a mathematical way. We start by writing out the state of the auxiliary system of size spin-1/2 particles shown in Fig. 5.9a, $|\sigma\rangle = |a_1, b_1; a_2, b_2; a_3, b_3\rangle = |\sigma_1, \sigma_2, \sigma_3\rangle$. Each $\Sigma_{b_i, a_{i+1}}$ operator represented by a solid bond projects the auxiliary particles on neighboring sites (i.e. b_n, a_{n+1}) into the singlet state. We can represent spin-up by $|0\rangle$ and spin-down by $|1\rangle$, so these operators have the form,

$$\Sigma = \begin{bmatrix} 0 & \frac{1}{\sqrt{2}} \\ -\frac{1}{\sqrt{2}} & 0 \end{bmatrix} \quad (5.30)$$

Each M operator projects the system into the physical spin-1 space. Since there are three possibilities for the spin, we have three operators, M^+ , M^0 , and M^- . M^+ projects the auxiliary particles on a site (a_n, b_n) into the state $|00\rangle = |\tilde{+}\rangle$. We have,

$$M^+ = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}. \quad (5.31)$$

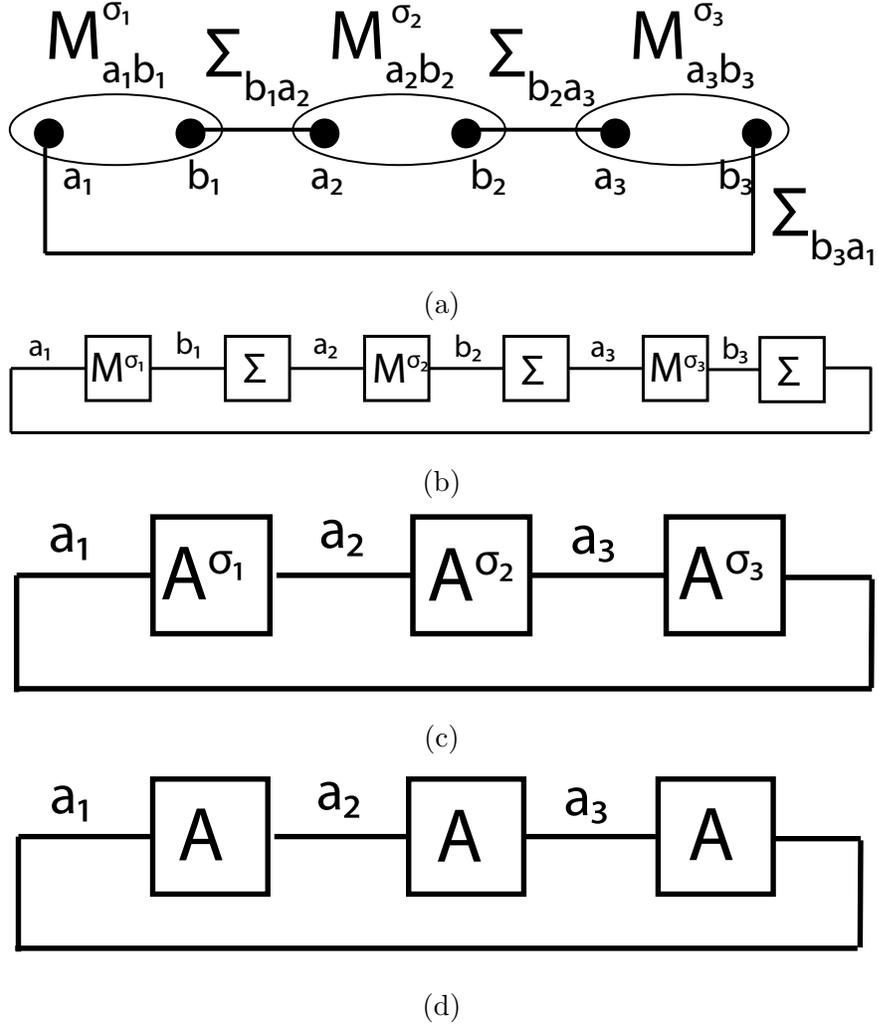


Figure 5.9: Tensor network representation for constructing the AKLT matrix product state. This construction follows the one shown in [10].

Similarly, M^0 projects the particles into the state $\frac{1}{\sqrt{2}}(|01\rangle + |10\rangle) = |\tilde{0}\rangle$. We have,

$$M^0 = \begin{bmatrix} 0 & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & 0 \end{bmatrix}. \quad (5.32)$$

Finally M^- projects the particles into the state $|11\rangle = |\tilde{-}\rangle$, and we have,

$$M^- = \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix}. \quad (5.33)$$

Using these forms for the operators, we can apply them to the state $|\sigma\rangle$ accord-

ing to the diagram in Fig. 5.9b. We have,

$$|\psi\rangle = \sum_{ab} \sum_{\sigma} M_{a_1 b_1}^{\sigma_1} \Sigma_{b_1 a_2} M_{a_2 b_2}^{\sigma_2} \Sigma_{b_2 a_3} M_{a_3 b_3}^{\sigma_3} \Sigma_{b_3 a_1} |\sigma\rangle. \quad (5.34)$$

Now we can simplify the network by contracting over the b 's, which is just matrix multiplication of M and Σ . After contraction, we arrive at Fig. 5.9c. We have,

$$A^+ = M^+ \Sigma = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} 0 & \frac{1}{\sqrt{2}} \\ -\frac{1}{\sqrt{2}} & 0 \end{bmatrix} = \begin{bmatrix} 0 & \frac{1}{\sqrt{2}} \\ 0 & 0 \end{bmatrix} \quad (5.35)$$

$$A^0 = M^0 \Sigma = \begin{bmatrix} 0 & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & 0 \end{bmatrix} \begin{bmatrix} 0 & \frac{1}{\sqrt{2}} \\ -\frac{1}{\sqrt{2}} & 0 \end{bmatrix} = \begin{bmatrix} -\frac{1}{2} & 0 \\ 0 & \frac{1}{2} \end{bmatrix} \quad (5.36)$$

$$A^- = M^- \Sigma = \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 0 & \frac{1}{\sqrt{2}} \\ -\frac{1}{\sqrt{2}} & 0 \end{bmatrix} = \begin{bmatrix} 0 & 0 \\ -\frac{1}{\sqrt{2}} & 0 \end{bmatrix} \quad (5.37)$$

We must normalize the A matrices. Both left and right normalization gives the same answer in this example. Explicitly, if we right normalize we have,

$$\sum_{\sigma} A^{\sigma} A^{\sigma\dagger} = \begin{bmatrix} 0 & \frac{1}{\sqrt{2}} \\ 0 & 0 \end{bmatrix} \begin{bmatrix} 0 & 0 \\ \frac{1}{\sqrt{2}} & 0 \end{bmatrix} + \begin{bmatrix} \frac{1}{4} & 0 \\ 0 & \frac{1}{4} \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ -\frac{1}{\sqrt{2}} & 0 \end{bmatrix} \begin{bmatrix} 0 & -\frac{1}{\sqrt{2}} \\ 0 & 0 \end{bmatrix} = \frac{3}{4} I \quad (5.38)$$

. If we left normalize we have,

$$\sum_{\sigma} A^{\sigma\dagger} A^{\sigma} = \begin{bmatrix} 0 & 0 \\ \frac{1}{\sqrt{2}} & 0 \end{bmatrix} \begin{bmatrix} 0 & \frac{1}{\sqrt{2}} \\ 0 & 0 \end{bmatrix} + \begin{bmatrix} \frac{1}{4} & 0 \\ 0 & \frac{1}{4} \end{bmatrix} + \begin{bmatrix} 0 & -\frac{1}{\sqrt{2}} \\ 0 & 0 \end{bmatrix} \begin{bmatrix} 0 & 0 \\ -\frac{1}{\sqrt{2}} & 0 \end{bmatrix} = \frac{3}{4} I \quad (5.39)$$

. The normalized A matrices are,

$$A^+ = \begin{bmatrix} 0 & \sqrt{\frac{2}{3}} \\ 0 & 0 \end{bmatrix} \quad (5.40)$$

$$A^0 = \begin{bmatrix} -\frac{1}{\sqrt{3}} & 0 \\ 0 & \frac{1}{\sqrt{3}} \end{bmatrix} \quad (5.41)$$

$$A^- = \begin{bmatrix} 0 & 0 \\ -\sqrt{\frac{2}{3}} & 0 \end{bmatrix} \quad (5.42)$$

Including the normalization, we are left with the state represented in Fig. 5.9c,

$$|\psi\rangle = \sum_a \sum_\sigma A_{a_1 a_2}^{\sigma_1} A_{a_2 a_3}^{\sigma_2} A_{a_3 a_1}^{\sigma_3} |\sigma\rangle. \quad (5.43)$$

We can rewrite the contraction of the A matrices over a indices in terms of a regular matrix product and the trace (because of the periodic boundary). We have,

$$|\psi\rangle = \sum_\sigma \text{Tr}(A^{\sigma_1} A^{\sigma_2} A^{\sigma_3}) |\sigma\rangle. \quad (5.44)$$

Expanding out the sum over σ gives,

$$|\psi\rangle = \text{Tr}((A^- |\tilde{-}\rangle + A^0 |\tilde{0}\rangle + A^+ |\tilde{+}\rangle)(A^- |\tilde{-}\rangle + A^0 |\tilde{0}\rangle + A^+ |\tilde{+}\rangle) \\ \times (A^- |\tilde{-}\rangle + A^0 |\tilde{0}\rangle + A^+ |\tilde{+}\rangle)).$$

If we let the A 's include the state vectors we get,

$$A = A^+ |\tilde{+}\rangle + A^0 |\tilde{0}\rangle + A^- |\tilde{-}\rangle = \frac{1}{\sqrt{3}} \begin{bmatrix} -|\tilde{0}\rangle & \sqrt{2}|\tilde{+}\rangle \\ -\sqrt{2}|\tilde{-}\rangle & |\tilde{0}\rangle \end{bmatrix} \quad (5.45)$$

and,

$$|\psi\rangle = \text{Tr}(AAA). \quad (5.46)$$

This is the tensor network represented by Fig. 5.9d.

A matrix product state representation can provide insight into the properties of the wavefunction. We see from Eqn. 5.46 that the AKLT state is translationally invariant (i.e. we can increase the number of particles in the AKLT state by including more A matrices). Also, the trace implies that the state has periodic boundary conditions.

Now let us find more properties of the AKLT state using the MPS form. We start by writing the state in terms of the 9 vectors that span the Hilbert space of two spin-1 particles (i.e. using the form $|s, m\rangle = \sum_i C_i |m_1 m_2\rangle$):

$$|2, 2\rangle = |\tilde{+}\tilde{+}\rangle \quad (5.47)$$

$$|2, 1\rangle = \frac{1}{\sqrt{2}} |\tilde{+}\tilde{0}\rangle + \frac{1}{\sqrt{2}} |\tilde{0}\tilde{+}\rangle \quad (5.48)$$

$$|2, 0\rangle = \frac{1}{\sqrt{6}} |\tilde{+}\tilde{-}\rangle + \sqrt{\frac{2}{3}} |\tilde{0}\tilde{0}\rangle + \frac{1}{\sqrt{6}} |\tilde{-}\tilde{+}\rangle \quad (5.49)$$

$$|2, -1\rangle = \frac{1}{\sqrt{2}} |\tilde{0}\tilde{-}\rangle + \frac{1}{\sqrt{2}} |\tilde{-}\tilde{0}\rangle \quad (5.50)$$

$$|2, -2\rangle = |\tilde{-}\tilde{-}\rangle \quad (5.51)$$

$$|1, 1\rangle = \frac{1}{\sqrt{2}} |\tilde{+}\tilde{0}\rangle - \frac{1}{\sqrt{2}} |\tilde{0}\tilde{+}\rangle \quad (5.52)$$

$$|1, 0\rangle = \frac{1}{\sqrt{2}} |\tilde{+}\tilde{-}\rangle - \frac{1}{\sqrt{2}} |\tilde{-}\tilde{+}\rangle \quad (5.53)$$

$$|1, -1\rangle = \frac{1}{\sqrt{2}} |\tilde{0}\tilde{-}\rangle - \frac{1}{\sqrt{2}} |\tilde{-}\tilde{0}\rangle \quad (5.54)$$

$$|0, 0\rangle = \frac{1}{\sqrt{3}} |\tilde{+}\tilde{-}\rangle - \frac{1}{\sqrt{3}} |\tilde{0}\tilde{0}\rangle + \frac{1}{\sqrt{3}} |\tilde{-}\tilde{+}\rangle \quad (5.55)$$

Taking two of the A matrices and multiply them together gives,

$$\begin{bmatrix} |\tilde{0}\rangle & \sqrt{2}|\tilde{+}\rangle \\ -\sqrt{2}|\tilde{-}\rangle & -|\tilde{0}\rangle \end{bmatrix} \begin{bmatrix} |\tilde{0}\rangle & \sqrt{2}|\tilde{+}\rangle \\ -\sqrt{2}|\tilde{-}\rangle & -|\tilde{0}\rangle \end{bmatrix} = \begin{bmatrix} |A\rangle & |B\rangle \\ |C\rangle & |D\rangle \end{bmatrix} \quad (5.56)$$

$$\begin{aligned} |A\rangle &= |\tilde{0}\tilde{0}\rangle - 2|\tilde{+}\tilde{-}\rangle = -\sqrt{3}|0, 0\rangle - \sqrt{2}|1, 0\rangle \\ |B\rangle &= \sqrt{2}|\tilde{0}\tilde{+}\rangle - \sqrt{2}|\tilde{+}\tilde{0}\rangle = -2|1, 1\rangle \\ |C\rangle &= -\sqrt{2}|\tilde{-}\tilde{0}\rangle + \sqrt{2}|\tilde{0}\tilde{-}\rangle = 2|1, -1\rangle \\ |D\rangle &= -2|\tilde{-}\tilde{+}\rangle + |\tilde{0}\tilde{0}\rangle = \sqrt{2}|1, 0\rangle - \sqrt{3}|0, 0\rangle \end{aligned} \quad (5.57)$$

The MPS form for the AKLT state is enlightening. We immediately see that up to a 4-fold degeneracy, the AKLT state is unique. We also see that the AKLT state can never be a spin-2 state: the states comprising the 4-fold degeneracy are

all in the spin-1 subspace. This form immediately suggests a generalization to states having more particles. If we stare at the matrix form, we can see AKLT states having more particles can easily be created by including more A matrices in the product. The larger state also retains its properties: multiplying more A matrices still gives a 4-fold degeneracy, and the four states always stay in the spin-1 subspace (for example, a $|\tilde{-}\rangle$ can be followed by a $|\tilde{+}\rangle$ or $|\tilde{0}\rangle$ but never another $|\tilde{-}\rangle$, etc.). If we consider periodic boundary conditions, the trace produces an AKLT state having zero spin.

5.9.1 Symmetry of AKLT state

In Sec. 5.9 we discovered some properties of the AKLT state using the MPS form. We discovered that the spin of the states are dependent on the parity of the auxiliary particles on the end points. Now we ask the question, Why is the spin of the physical system dictated by the auxiliary system? The answer is because both systems have the same symmetry. In this section, we aim to explain this reasoning by investigating some symmetry and conservation laws.

Conservation Laws and Symmetry

One rule of symmetry is: if we have an initial symmetry, then the final system will have that symmetry. For example, let particles A and B have a \mathbb{Z}_2 symmetry (i.e. they can be in the state +1 or -1). Say the particles A and B combine to form new particles, C and D ($A*B = C*D$). We see that the possible states of A*B are +1 or -1, so the possible states of C*D are also +1 and -1. The particles C and D also have \mathbb{Z}_2 symmetry.

A version of Noether's theorem says where there is a conservation law, there is a corresponding symmetry and vice versa. Two simple examples of symmetries that are related to conserved quantities from classical examples are: (i) translational symmetry and momentum (a particle with the same potential energy at point a and point b with the same mass and velocity will have the same momentum at those points) and (ii) rotational symmetry and angular momentum (a particle in a circular orbit will have the same angular momentum). So, if the Hamiltonian cannot distinguish between states of the system (there is some degeneracy), then there is a symmetry revealed by those states having the same energy. A quantum example is the Ising model,

$$H = - \sum_{i=1}^{n-1} \sigma_z^{(i)} \sigma_z^{(i+1)} \quad (5.58)$$

where the the ground states are,

$$|\psi^{(1)}\rangle = |\uparrow\uparrow \dots\rangle \quad (5.59)$$

$$|\psi^{(2)}\rangle = |\downarrow\downarrow \dots\rangle \quad (5.60)$$

The energy is the same for these two possible ground states, so there is a \mathbb{Z}_2 symmetry.

One method of finding symmetries (and therefore conserved quantities) is looking for things that commute with the Hamiltonian. For example, $\prod_{i=1}^n \sigma_x^{(i)}$ commutes with the Hamiltonian. If we apply that to one ground state, say $|\psi^{(1)}\rangle$, we get the other $|\psi^{(2)}\rangle$. *Parallel:* In language of quantum algorithms, the Hamiltonian represents the code and the degenerate states (and therefore symmetries and conserved quantities) are found using the logical operators of the code.

Now that we have some very basic idea of symmetry and conserved quantities, let us look at the AKLT state again.

Symmetry of the AKLT State

Here we prove that the symmetry of the physical system is dictated by the spin of the auxiliary system. Let $|\hat{\omega}\rangle = \frac{1}{\sqrt{2}}(|\uparrow\rangle - |\downarrow\rangle)$ be the singlet state of two auxiliary particles from adjacent sites. We know that $|\hat{\omega}\rangle$ is $SU(2)$ invariant. This is easy to show using the 2D representation (or spin-1/2 representation) of $SU(2)$. Let $u \in SU(2)$ where,

$$SU(2) = \left\{ \begin{bmatrix} \alpha & -\bar{\beta} \\ \beta & \bar{\alpha} \end{bmatrix} : \alpha, \beta \in \mathbb{C}, |\alpha|^2 + |\beta|^2 = 1 \right\}. \quad (5.61)$$

By multiplying this out and applying the conditions for the coefficients we see,

$$u \otimes u |\hat{\omega}\rangle = |\hat{\omega}\rangle. \quad (5.62)$$

Specifically, the 2D representation of $SU(2)$ has:

$$\text{basis: } \left| \frac{1}{2}, \frac{1}{2} \right\rangle = \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \left| \frac{1}{2}, -\frac{1}{2} \right\rangle = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \quad (5.63)$$

$$\text{generators: } J_1 = \frac{1}{2} \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}, J_2 = \frac{1}{2} \begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix}, J_3 = \frac{1}{2} \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} \quad (5.64)$$

These are the Pauli matrices. Now, all we need is a 3D representation (spin-1 representation) of $SU(2)$. It is,

$$\text{basis: } |1, 1\rangle = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}, |1, 0\rangle = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}, |1, -1\rangle = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \quad (5.65)$$

$$\text{generators: } J_1 = \frac{1}{\sqrt{2}} \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix}, J_2 = \frac{1}{\sqrt{2}} \begin{bmatrix} 0 & -i & 0 \\ i & 0 & -i \\ 0 & i & 0 \end{bmatrix}, J_3 = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & -1 \end{bmatrix}$$

To show that the symmetry is preserved, start by letting R_u : spin-1 representation of u and $P = \prod_{s=1} P_s$ be the projector into the spin-1 space, and $|\psi\rangle$ is the state of the physical system. The 2D and 3D representations of $SU(2)$ show us that acting on the physical system with a spin-1 rotation is the same as acting on the auxiliary system with spin-1/2 rotation on both particles. That is,

$$R_u P = P(u \otimes u). \quad (5.66)$$

Now we can show that if the auxiliary system has some rotational invariance (symmetry), then the physical system has that same symmetry.

$$\begin{aligned} R_u^{\otimes n} |\psi\rangle &= R_u^{\otimes n} (P^{\otimes n} |\hat{\omega}\rangle^{\otimes n}) \\ &= (R_u P)^{\otimes n} |\hat{\omega}\rangle^{\otimes n} \\ &= (P(u \otimes u))^{\otimes n} |\hat{\omega}\rangle^{\otimes n} \\ &= P^{\otimes n} |\hat{\omega}\rangle^{\otimes n} \\ &= |\psi\rangle, \end{aligned}$$

where in step 2-3 we used Eqn. 5.66.

We've shown that the two representations of the AKLT state have the same rotational symmetry and by Noether's theorem, they have conserved angular momentum. This is why the spin of the physical system is dictated by the spin of the auxiliary system.

5.9.2 W-state

For further understanding of the MPS, let us examine an example of a case when the MPS matrices are not the same on every site. One such example is the W-state.

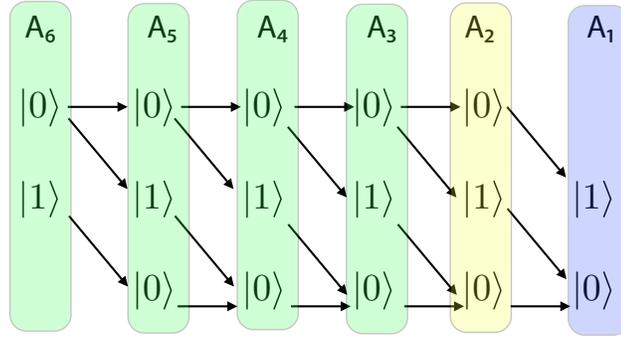


Figure 5.10: Binary tree for the W-state representing the structure of its MPS form.

The W state on 6 qubits is,

$$|W\rangle = \frac{1}{\sqrt{6}} (|100000\rangle + |010000\rangle + |001000\rangle + |000100\rangle + |000010\rangle + |000001\rangle)$$

and the structure of the kets are represented by the binary tree in Fig. 5.10:

To get the form for each MPS, one can start at the back of Fig. 5.10. The last number in a ket is either a 0 or a 1, so the A matrix is,

$$A_1 = \begin{bmatrix} |1\rangle & 0 \\ 0 & |0\rangle \end{bmatrix} \quad (5.67)$$

In the end, we only care about the trace, so we determine a and d in,

$$\begin{bmatrix} a & b \\ c & d \end{bmatrix} \begin{bmatrix} |1\rangle & 0 \\ 0 & |0\rangle \end{bmatrix} = \begin{bmatrix} a|1\rangle & b|0\rangle \\ c|1\rangle & d|0\rangle \end{bmatrix}. \quad (5.68)$$

Now to determine A_2 and A_3 , we use the structure of matrix multiplication,

$$\begin{bmatrix} a_3 & b_3 \\ c_3 & d_3 \end{bmatrix} \begin{bmatrix} a_2 & b_2 \\ c_2 & d_2 \end{bmatrix} \begin{bmatrix} |1\rangle & 0 \\ 0 & |0\rangle \end{bmatrix} = \begin{bmatrix} (a_3a_2 + b_3c_2)|1\rangle & (a_3b_2 + b_3d_2)|0\rangle \\ (c_3a_2 + d_3c_2)|1\rangle & (c_3b_2 + d_3d_2)|0\rangle \end{bmatrix}. \quad (5.69)$$

If the last number is 1, then all preceding numbers in the ket have to be 0, so we choose $a_3 = a_2 = |0\rangle$ and $b_3 = c_2 = 0$. We need both terms in the sum for the kets

that end in 0, so we also choose $c_3 = |1\rangle$, $b_2 = |0\rangle$, $d_3 = |0\rangle$ and $d_2 = |1\rangle$. This gives,

$$A_2 = \begin{bmatrix} |0\rangle & |0\rangle \\ 0 & |1\rangle \end{bmatrix} \quad (5.70)$$

$$A_3 = \begin{bmatrix} |0\rangle & 0 \\ |1\rangle & |0\rangle \end{bmatrix} \quad (5.71)$$

We realize that $A_3 = A_4 = A_5 = A_6$. Multiplying times A_3 always puts a zero on the upper left and right kets, and it puts a 1 on the upper right ket and a 0 on the lower right ket and adds them into the lower right ket. Because we take the trace, it does not matter what the lower left ket contains. We then repeat on each site and end up with,

$$|W\rangle = \text{Tr} \left(\begin{bmatrix} |0\rangle & 0 \\ |1\rangle & |0\rangle \end{bmatrix} \begin{bmatrix} |0\rangle & 0 \\ |1\rangle & |0\rangle \end{bmatrix} \begin{bmatrix} |0\rangle & 0 \\ |1\rangle & |0\rangle \end{bmatrix} \begin{bmatrix} |0\rangle & 0 \\ |1\rangle & |0\rangle \end{bmatrix} \begin{bmatrix} |0\rangle & |0\rangle \\ 0 & |1\rangle \end{bmatrix} \begin{bmatrix} |1\rangle & 0 \\ 0 & |0\rangle \end{bmatrix} \right).$$

5.10 Using DMRG to Model the Superinductor

Recent superconducting circuit designs require parts of the circuit to have a large number of degrees of freedom [9, 26, 27]. Such designs motivate the development of simulation toolboxes for modeling large circuits [8, 11]. Another motivation for simulating larger circuits is to create qubits with high coherence times (such as fluxonium [8, 9]), engineering topological protection (as in the current mirror [91]), and simulating multi-qubit systems (i.e. gates between qubits).

Some software programs widely available to the public, see [87], use approximate methods, such as those discussed in Sec. 5.4. We saw that these approximate methods are useful for weakly non-linear circuits.

We would like to find a method that can simulate the full circuit model. In this section, we adopt the results from [11], where we discussed a tool for constructing and analyzing the exact model of large superconducting circuits. This talk introduced Circuitizer, which uses QuTiP [104, 105], an open-source software for simulating the dynamics of open quantum systems. Circuitizer was combined with ITensor [12], a C++ library for implementing tensor network calculations, in an effort to simulate circuits beyond the boundary where exact diagonalization fails (i.e. ten degrees of freedom requires on the order of a terabyte per eigenstate). By combining the modeling techniques, we calculated the low-energy spectrum of a family of superinducting circuits having a local Hilbert space of 21 with up to 64 degrees of freedom (corresponding to a 21^{64} dimensional Hilbert space). We also include error estimates for these results.

5.10.1 The DMRG Software Tool

The toolbox discussed in [11] implements a software flow diagram, see Fig. 5.11. The software flow starts by utilizing Circuitizer and QuTip starts to create a Python model of a superconducting circuit that the user design within the tool. Analysis of this user-specified circuit design produces the Hamiltonian for the circuit. An MPO for this Hamiltonian is constructed in Python using the techniques we discussed in Sections 5.7 and 5.8. This MPO is imported into ITensor [12, 98] using the json format.

At this time the user can specify a MPO bond dimension reduction (which sometimes results in approximation). If the bond dimension reduction is requested, then the MPO is compressed in ITensor [106]. A DMRG algorithm is performed in ITensor and it produces the MPS eigenstates.

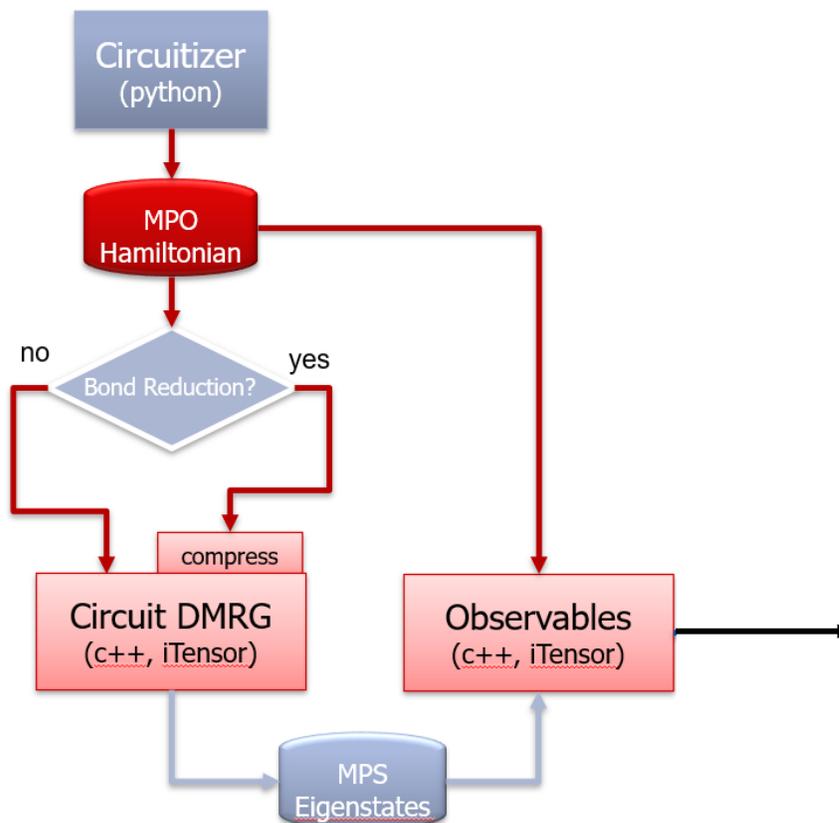


Figure 5.11: Software flow diagram of our software tool developed for simulating superconducting circuits. This image is adopted from [11]. In Python, the user draws a superconducting circuit in Circuitizer. Circuitizer then interfaces with the program QuTip to produce the Hamiltonian. An MPO form of this Hamiltonian is produced using algorithms derived from the techniques described in Sections 5.7 and 5.8. Then, the MPO is stored in a json file and imported into ITensor. ITensor performs DMRG and gives the eigenstates in MPS form. The MPS is contracted with the exact Hamiltonian (before compression) to determine the convergence properties of the simulation (i.e. at each iterative step during DMRG, the variance is calculated using the exact Hamiltonian MPO and the eigenstate produced at that step). Upon reaching a stopping criterion, the loop halts. At this time, observables of interest can be calculated using the output MPS. For example, the MPO form for the charge operator, which can include offset charges on the junctions, can be produced in Python, imported into ITensor, and contracted with the MPS to find the charge dispersion effects due to offset charges.

Observables specified by the user are calculated using the MPS eigenstates. An error estimate is calculated using the variance. The variance is calculated using the Hamiltonian MPO before compression. (Note: With highly-accurate calculations producing very small variances, catastrophic cancellation can be encountered producing negative values of the variance near machine precision, which should not be cause for alarm. It is also useful to note that there are several ways to contract the MPS-MPO-MPS tensor network in ITensor, and slight variations in the results from these different contraction methods, which are almost always below the desired precision, and should also not be cause for alarm. These deviations are to be expected with highly-accurate calculations of the MPS, which produce energy variances which are zero to machine precision. If accurate values of the variance near machine precision are desired for some reason, one can easily implement numerically stable algorithms for calculating the variance that mitigate loss of precision, see [107, 108].) Some other observables of interest that can be calculated are the lowest-lying energies, the zero-point fluctuations of the circuit in charge or phase, and the charge dispersion due to offset charges.

One can also find an upper bound on the error in a DMRG simulation by using the smallest gap between states (assuming a state has not been skipped) and the variance. The chosen stopping criterion for the simulations in [11] is an accuracy better than 10^{-6} . The difference between the actual energy and the calculated energy for the i th state is, $\Delta E \leq \frac{\sigma_i^2}{\Delta_i}$, where σ_i^2 is the variance and Δ_i is the smallest energy gap between the i th state and a neighboring state.

Estimating the error using an error bound can be expensive since computing the variance could involve contractions of large tensor networks (having large bond and local Hilbert space dimensions). In the literature, there are approaches to

calculating the expectation values used in the variance which are efficient [109]. One strategy to estimate the error in energy is to quantify the difference in error as at various bond dimensions: one can run multiple DMRG simulations at various bond dimensions and plot the energies and extrapolate using a polynomial fit. The difference in energy between the extrapolated value and the value having the largest bond dimension gives the error bar. Another method for estimating error is to take multiple data sets under various conditions and obtain the error bar from the range of values [98].

The bond dimension is related to the truncation error, which is $\frac{\sum_{n \in \text{discarded}} \lambda_n^2}{\sum_n \lambda_n^2} \leq \epsilon$, where ϵ is the user-defined cutoff, and λ_n are the smallest singular values such that the truncation error is less than ϵ [98]. In other words, in the SVD, $M = UDV^\dagger$, values we set the smallest diagonal elements of D to zero so that the new matrix $M' = UD'V^\dagger$ has a zero column. This column can then be truncated, reducing the dimension of the original matrix M with minimal approximation.

The truncation error is the sum of the squares of the singular values we set to zero – it is the norm of the difference in the M matrices normalized by the sum of the squares of all of the singular values [98]. So, one way to quantify the error from bond dimension truncation is to take the norm of the difference $\|M - M'\|^2$, which should be small.

The DMRG simulation time is dependent on the local Hilbert space dimension, the number of degrees of freedom, the MPS and MPO bond dimension, the specified target variance, and the initial state. In some cases, a well-chosen initial state can reduce simulation time. Running multiple simulations using a large number of different initial states, however, is one of the best methods for demonstrating

convergence [98].

5.10.2 Low-Energy Spectrum for a Family of Superinductors

In this section, we use the software tool introduced in Sec. 5.10.1 to find the low-energy spectrum for a family of superinductors having 4 to 64 degrees of freedom and a local Hilbert space dimension of 21.

In Fig. 5.12a, we plot the energy spectrum of a family of superinducting circuits increasing in size. These superinducting circuits are efficiently described by an MPO with Nd^2m^2 parameters and have wavefunctions efficiently represented using Ndk^2 parameters, where N is the number of nodes, k is the MPS bond dimension, m is the MPO bond dimension, and d is the local Hilbert space dimension, see Fig. 5.6. The low-energy spectrum for a family of these superinducting circuits is shown in Fig. 5.12b.

The upper error bound on the energy is shown in Fig. 5.13a. Since the bound is dependent on the energy difference between nearest states, states that are nearly degenerate suffer from a larger upper bound. This effect is particularly prominent where the energies of the second and third excited states coincide at 30 nodes, see Fig. 5.12b.

The variance used to calculate the bound is shown in Fig. 5.13b. The first stopping criterion for this calculation is a variance below 10^{-6}GHz^2 . For circuits having a node number larger than 30, the target variance was not obtained, and the simulations were halted after reaching the same specified sweep number (at

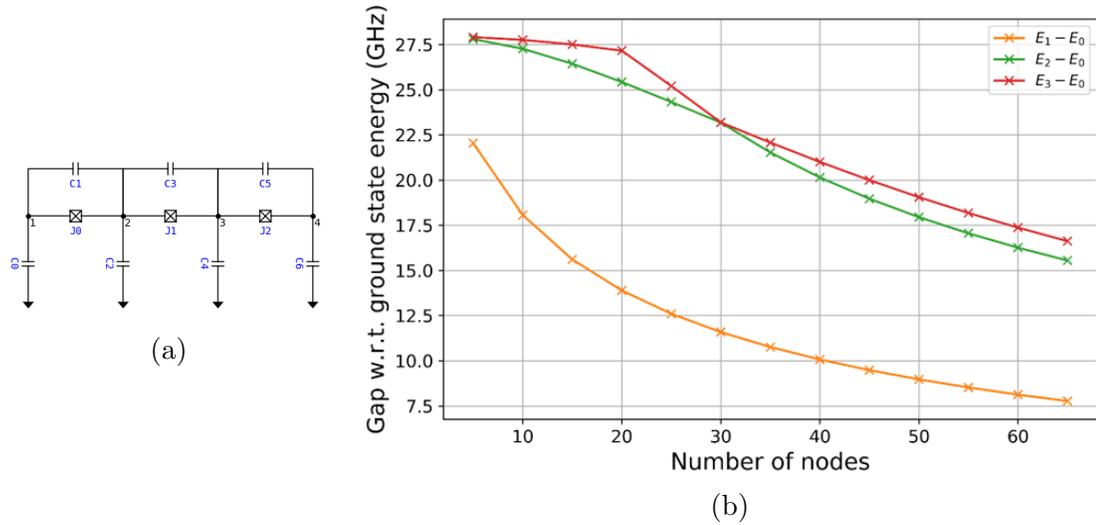


Figure 5.12: Left: A four-node superinductor drawn in Circuitizer. The user specifies the junction properties and capacitances. Right: Transition energies from the ground state for the first three excited states for a family of superinducting circuits from 5-64 nodes, where the junctions are in the transmon regime. Excited states are obtained by adding a penalty term, see [12]. Image adopted from [11].

least 150 sweeps).

To validate the error bound for a small superinductor, we calculate an actual error between exact diagonalization and DMRG and compare this with calculated upper bound. The absolute actual error remains below the calculated upper bound, see Fig. 5.13c.

We can also see that the truncation error affects the error in the energy. In Fig. 5.14, we incrementally decrease the truncation error as the sweep number increases and show how the error bound reduces.

We compare the DMRG data to our harmonic mode analysis of the circuit, see the method described in Sec. 5.4. In Fig. 5.15, we plot the normal modes of the circuit along with the energy spectrum calculated from DMRG. We see that the normal modes coincide well with the DMRG results. We can also see

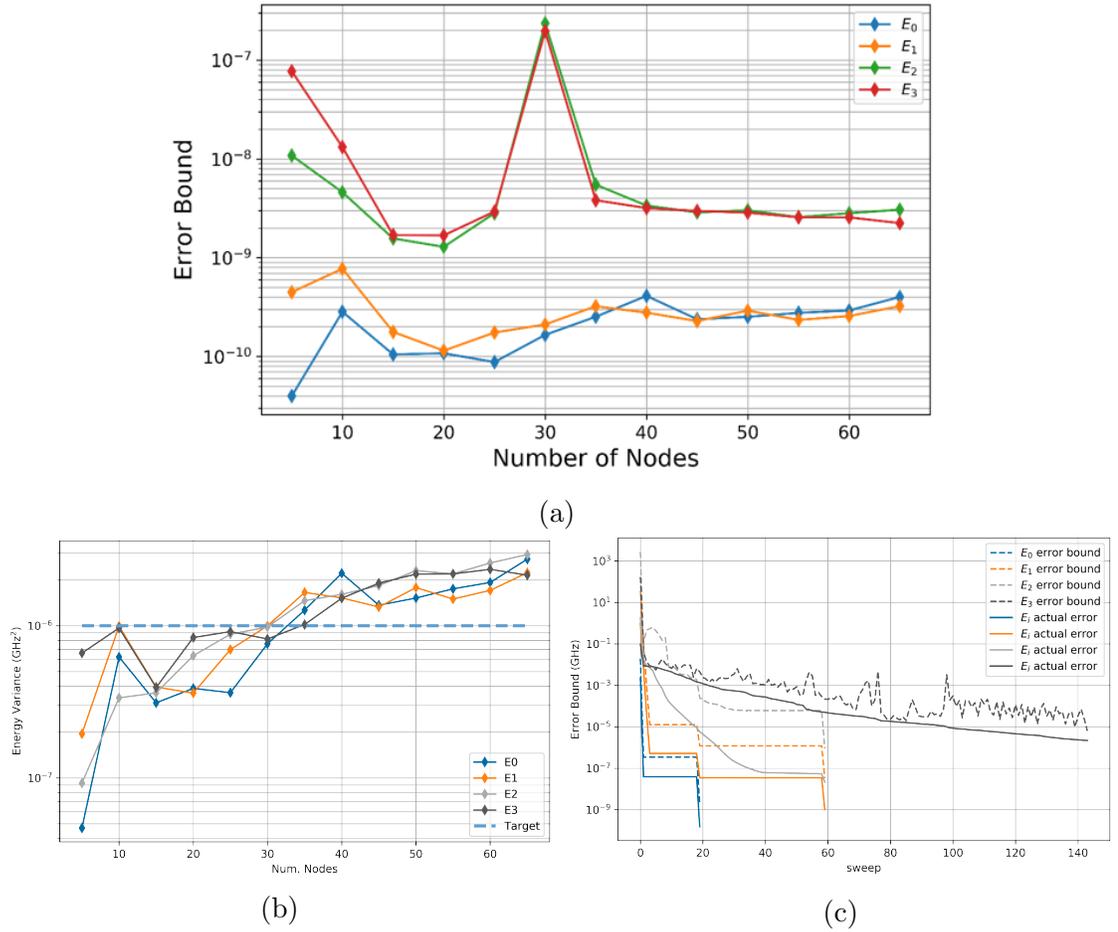


Figure 5.13: Top: Upper bound on the energy difference (in GHz) between the calculated and actual energies for a family of superinductors having 5-64 nodes.. The worst case errors have an upper error bound of 500Hz corresponding to the peak at 30 nodes for the second and third excited states. Bottom (left): Variance at the last sweep used in calculating the error bound in the top figure. Bottom (right): Comparison of upper error bound and the actual error between DMRG results and exact diagonalization for a five-node superinductor.

that the energy spectrum calculated using DMRG captures the nonlinearity of the Josephson junctions, since the DMRG energies lie below the harmonic modes. At circuit sizes below 20 nodes, the DMRG results show that more harmonic levels are required to capture the third excited state.

In Fig. 5.16, we show the time to solution in hours for a family of superin-

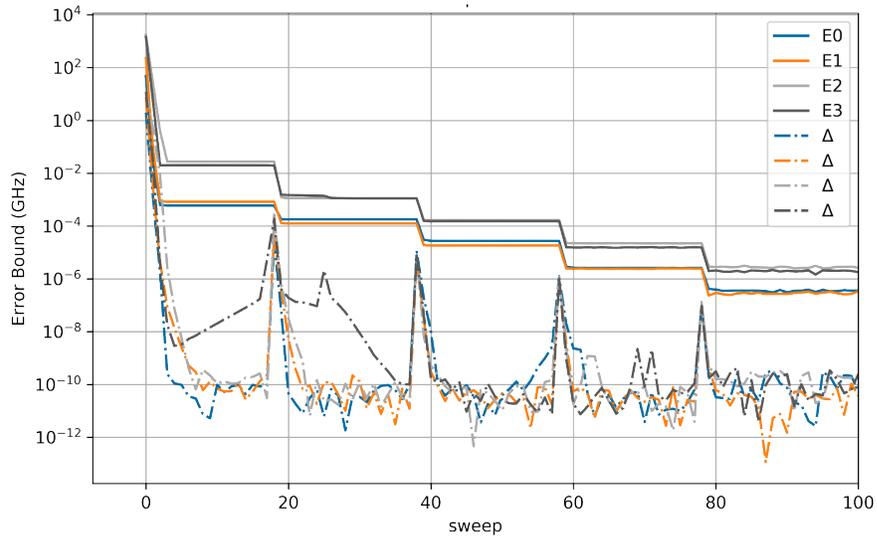


Figure 5.14: Error bound and the change in energy between neighboring sweeps versus sweep number for the first four states. Each step down in the in the error bound corresponds to a drop in the truncation error by one order of magnitude. The truncation error begins at 10^{-8} and ends at 10^{-12} .

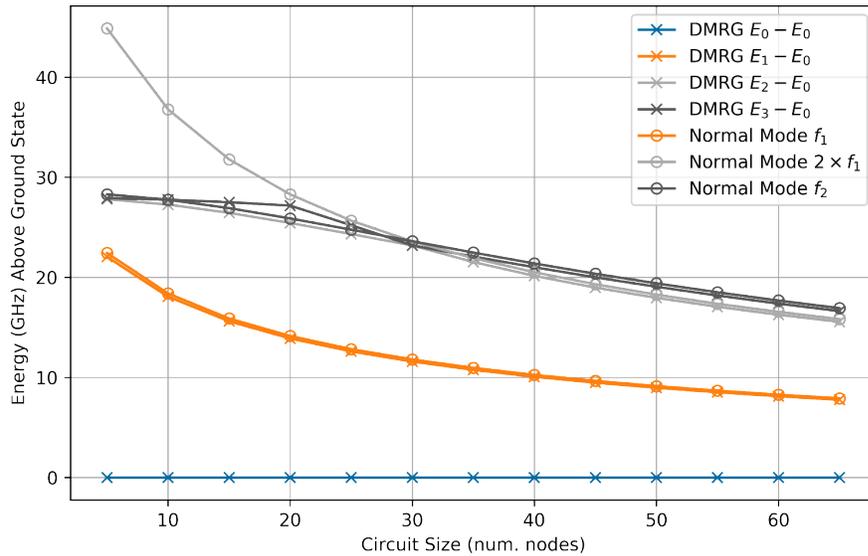


Figure 5.15: Comparison of the harmonic analysis and DMRG results for the lowest-lying energies of various superinductors.

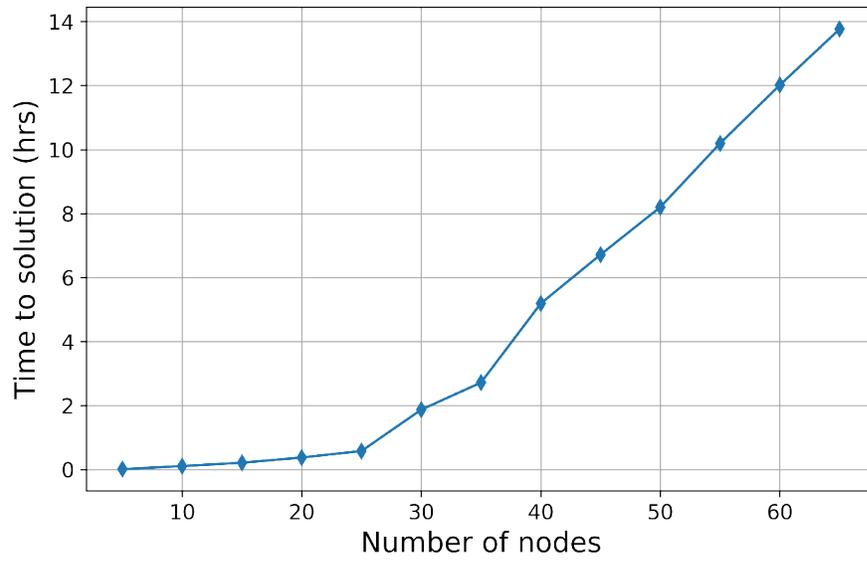


Figure 5.16: Time to solution for the DMRG simulation versus superinductor size. The calculation includes the time required to calculate the variance.

ductors. The initial state for all the simulations is chosen at random. The full simulation time to simulate these circuits is impressive – it takes about 14 hours to calculate the lowest-lying energy spectrum and validate the simulation error for a superinducting circuit having a Hilbert space of 21^{64} .

CHAPTER 6

DISSIPATION ENGINEERING IN OPEN QUANTUM SYSTEMS

In elementary physics classes, much of the material learned encapsulates properties of closed systems and neglects the effects of the environment. For example, a typical elementary homework problem involving the calculation of the velocity of an object sliding down a slope usually neglects air resistance. However, including environmental effects in the calculation provides insight into the paradigm of such calculations – including air resistance can allow one to discover the beneficial effects of using a parachute to slow a shuttle landing in a finite-sized air strip. Thus, simulation of open systems can lead one to discover convenient effects.

Here we consider an open system in quantum mechanics. On one hand, entanglement of the system with the environment can destroy the coherence of the quantum system. On the other hand, dissipation can produce longer-lasting quantum memories, decreased quantum computation time, and creating exotic quantum phases of matter. Studying open quantum systems allows us to understand the advantages and limits of environmental effects on quantum systems.

The total open system consists of two parts – the subsystem whose dynamics we are interested in, and the environment external to the system. The external environment is referred to as a reservoir which acts on the subsystem of interest. Interactions between the subsystem and the reservoir can have a recursive effect, called non-Markovian interactions. The interactions can also have no memory of previous interactions, and in this case, these interactions between the subsystem and reservoir are called Markovian interactions. Such an example of a Markovian interaction between the subsystem and the environment is dissipation.

Dissipation can be incredibly useful. One application for dissipation is stabilizing resource states for quantum computing. In the next chapter, Ch. 7, we discuss our proposal for the dissipative preparation of the Laughlin state, a fractional quantum Hall state.

CHAPTER 7

DRIVEN DISSIPATIVE PREPARATION OF FEW-BODY LAUGHLIN STATES OF RYDBERG POLARITONS IN TWISTED CAVITIES

We present a driven dissipative protocol for creating an optical analog of the Laughlin state in a system of Rydberg polaritons in a twisted optical cavity. We envision resonantly driving the system into a 4-polariton state by injecting photons in carefully selected modes. The dissipative nature of the polariton-polariton interactions leads to a decay into a two-polariton analog of the Laughlin state. Generalizations of this technique could be used to explore fractional statistics and anyon based quantum information processing. We also model recent experiments that attempt to coherently drive into this same state. This chapter is adapted from [110].

7.1 Introduction

Efforts to produce analogs of fractional quantum Hall effects [111] in atomic [112–114, 114–125] and optical settings [126–133] are driven by the desire to directly observe and manipulate “anyon” excitations, which act as particles with unconventional quantum statistics [134–137]. In addition to being of fundamental interest, these anyons can be useful for quantum information processing [60, 138–140]. Here we give a driven dissipative protocol for producing a minimal quantum Hall state in a system of Rydberg polaritons in a non-coplaner optical cavity.

The quantum Hall effect is seen in 2D electron systems in large magnetic fields. As argued by Schine et al. [133], there is a one-to-one correspondence between

quantum state of such electrons and the optical modes of a non-planar (or twisted) cavity. This correspondence has led to a number of proposals to use twisted cavities to reproduce quantum Hall physics [127, 141]. In order to produce an analog of electron-electron interactions, these proposals hybridize the photon modes in the cavity with long-lived Rydberg excitations of an atomic gas. The resulting Rydberg polaritons interact via a strong dipole-dipole interaction, which can be modelled as a short-range repulsion [142]. At particular filling factors, the ground state is expected to be a fractional quantum Hall state.

A key difference between the electronic and optical system is that polariton number is not conserved: One readily injects polaritons into the cavity by shining an appropriately tuned laser on the system; Polaritons decay due to the finite cavity lifetime, or in inelastic scattering processes. Here we take advantage of this feature, and show that one can drive this system in such a way that it naturally evolves into an analog of the Laughlin state, one of the most iconic examples of fractional quantum Hall states [143].

Our scheme is in the spirit of other driven-dissipative state preparation protocols [144–155], which have analogs in autonomous error correction [140, 156–163]. The basic idea is that you want to construct a situation where the state of interest is a “dark state” which is neither excited by the drive, nor decays through the dissipative channel. We rely upon one of the key features of the Laughlin state, namely that it vanishes when two electrons come close together. In the Rydberg polariton context, this means that inelastic two-body loss is strongly suppressed in the Laughlin state. We construct a drive which takes the system to an excited state, which then decays into the Laughlin state.

In section 7.2 we present our model of Rydberg Polaritons in twisted cavities.

We analyze the dynamics via a Lindblad equation, and in section 7.3 we describe our numerical approach. In section 7.4 we validate our model by reproducing the observations from [13], including calculating the pair correlations functions of emitted photons, one of the key signatures used in the experiment. Section 7.5 uses our validated model to introduce our driven dissipative protocol for producing the Laughlin state and provides our numerical study of the dynamics. Section 7.6 briefly describes how one can verify that that our approach is successful.

7.2 Modeling

Researchers in Chicago have designed an optical cavity where the modes are in one-to-one correspondence to the quantum states of 2D harmonically trapped charged particle in a magnetic field.

An atomic gas is placed in the cavity, and the cavity modes hybridize with a long-lived Rydberg excitation. An idealized effective model [141] for the resulting polaritons, in the cavity's image plane, is $H = H_0 + H_{\text{int}}$ with

$$\begin{aligned} H_0 &= \int d^2r \frac{1}{2m_0} \hat{\psi}^\dagger \left(\frac{1}{i} \nabla - eA \right)^2 \hat{\psi} + \frac{1}{2} m_0 \omega^2 r^2 \hat{\psi}^\dagger \hat{\psi} \\ H_{\text{int}} &= \int dr dr' V(r - r') \hat{\psi}^\dagger(r) \hat{\psi}^\dagger(r') \hat{\psi}(r') \hat{\psi}(r). \end{aligned} \quad (7.1)$$

The operator $\hat{\psi}(r)$ annihilates a polariton at location r , defined in the 2D plane. The effective vector potential A corresponds to a uniform effective magnetic field, and is related to the non-planarity of the cavity [164]. We use the symmetric gauge and take $A = -\frac{B}{2}y\hat{x} + \frac{B}{2}x\hat{y}$. The coefficients m_0 and ω are related to the cavity geometry and the amount of mixing between the photon modes and the Rydberg excitation. The polariton-polariton interaction $V(r)$ comes from the dipole-dipole interaction between the Rydberg excitations. It is short-ranged, and as long as the

cavity is sufficiently large it can be replaced by a local potential $V(r) \approx U\delta(r)$. Some technical details of arriving at this description are given in Appendix 7.9.1.

The modes which diagonalize Eq. (7.1) have frequencies $\omega_{mn} = (\omega_c + \Omega_0)n + (\omega_c - \Omega_0)m$, where $\Omega_0 = \sqrt{\omega_c^2 + \omega^2}$, and $\omega_c = \frac{eB}{2m_0}$ which is half the cyclotron frequency [165, 166]. When $\omega_c \approx \Omega_0$ (or equivalently $\omega \approx 0$), modes with different m have the same energy. These nearly-degenerate modes make up the lowest Landau level, and have wavefunctions,

$$\phi_m(x, y) = (x + iy)^m e^{-(x^2 + y^2)/d^2} \quad (7.2)$$

with normalization,

$$1/c_m^2 = \int |\phi_m(x, y)|^2 dx dy = \frac{d^{2(m+1)} \pi m!}{2^{m+1}}. \quad (7.3)$$

The characteristic length is $\frac{1}{d^2} = \frac{1}{2} m_0 \Omega_0$.

When multiple polaritons are in the cavity, the interaction term, Eq. (7.1), breaks this degeneracy. The preeminent role of the interactions leads to highly non-trivial physics. The ground state of this model when there is one polariton for every two modes is the $\nu = 1/2$ Laughlin state [13, 143]. It boasts fractionally charged anyonic excitations. When there is one polariton per mode, and the interactions are appropriate, one instead finds the $\nu = 1$ Pfaffian, with non-Abelian excitations [167]. The goal of these experiments is to produce such states and demonstrate their exotic properties.

The defining feature of the $\nu = 1/2$ state is that it is the highest density state which vanishes whenever two polaritons touch:

$$\Psi_L(z_1, z_2, \dots, z_N) = \prod_{i < j} (z_i - z_j)^2 e^{-\sum_j |z_j|^2/d^2}. \quad (7.4)$$

As a first step, Clark et al. have isolated three of the modes from Eq. (7.2), corresponding to $m = 3, 6, 9$ (all with $n = 0$). This is sufficient to produce a 2-polariton analog of the Laughlin state,

$$\psi_L(z_1, z_2) \propto z_1^3 z_2^3 (z_1^3 - z_2^3)^2 e^{-|z_1|^2/4d^2 - |z_2|^2/4d^2}. \quad (7.5)$$

Given the three accessible modes, this is the state with the largest number of polaritons that vanishes when two polaritons touch. It can also be written as

$$|L\rangle = \frac{|66\rangle - \sqrt{2.1}|39\rangle}{\sqrt{3.1}}, \quad (7.6)$$

where $|66\rangle = \hat{a}_6^\dagger \hat{a}_6^\dagger |\text{vac}\rangle / \sqrt{2}$ and $|39\rangle = \hat{a}_3^\dagger \hat{a}_9^\dagger |\text{vac}\rangle$.

Our goal is to construct a protocol for deterministically producing the two-photon state, $|L\rangle$. Once this goal is achieved, one can consider generalizations that will lead to the many-body state in Eq. (7.4).

We project H_{int} into the relevant space, writing

$$\hat{\psi}(x, y) = \sum_m c_m \hat{a}_m \phi_m(x, y) \quad (7.7)$$

where c_m and ϕ_m are given by Eqs. (7.2) and (7.3). This equation defines the annihilation operators \hat{a}_m . In order to model the experiment, we restrict the sum to be over $m = 3, 6, 9$ – but in a future experiment it could run over the entire lowest Landau level.

By substituting Eq. (7.7) into Eq. (7.1) and performing the resulting Gaussian integrals one arrives at

$$H_{\text{int}} = U \sum_{m_1 m_2 m_3 m_4} \Lambda_{m_1 m_2}^{m_3 m_4} \hat{a}_{m_1}^\dagger \hat{a}_{m_2}^\dagger \hat{a}_{m_3} \hat{a}_{m_4} \quad (7.8)$$

where the coefficient $\Lambda_{m_1 m_2}^{m_3 m_4}$ vanishes unless $m_1 + m_2 = m_3 + m_4$, in which case it is

$$\Lambda_{m_1 m_2}^{m_3 m_4} = \frac{1}{\pi d^2 2^{m_1 + m_2}} \frac{(m_1 + m_2)!}{(m_1! m_2! m_3! m_4!)^{1/2}}. \quad (7.9)$$

Although we do not make use of it, this interaction is separable: $\Lambda_{m_1 m_2}^{m_3 m_4} = \lambda_{m_1 m_2} \lambda_{m_3 m_4}$. One can readily verify that Eq. (7.6) is in the null-space of \hat{H}_{int} .

Polaritons can be injected into the cavity by shining lasers with appropriately shaped wavefronts onto the cavity mirror. This can be modeled by

$$\hat{H}_{\text{drive}} = \sum_m \lambda_m e^{i\nu_m t} \hat{a}_m + \text{h.c.} \quad (7.10)$$

Here ν_m is the frequency of the laser which couples to mode m . As long as $\nu_3 + \nu_9 = 2\nu_6$ one can remove all time dependence by shifting to a rotating frame, in which case the single-particle Hamiltonian is $H_1 = H_0 + H_{\text{drive}}$ with

$$\begin{aligned} H_0 &= -\omega_3 \hat{a}_3^\dagger \hat{a}_3 - \omega_6 \hat{a}_6^\dagger \hat{a}_6 - \omega_9 \hat{a}_9^\dagger \hat{a}_9 \\ H_{\text{drive}} &= \lambda_3 \left(\hat{a}_3 + \hat{a}_3^\dagger \right) + \lambda_6 \left(\hat{a}_6 + \hat{a}_6^\dagger \right) + \lambda_9 \left(\hat{a}_9 + \hat{a}_9^\dagger \right), \end{aligned} \quad (7.11)$$

here ω_m is the detuning of the drive from mode m .

Polaritons are lost through several processes, which we model using a Lindblad rate equation for the density matrix ρ ,

$$\frac{\partial \rho}{\partial t} = -i[H, \rho] + \left. \frac{\partial \rho}{\partial t} \right|_{\text{incoherent}} \quad (7.12)$$

Both the cavity modes and the the Rydberg excitations have finite lifetimes. These are modeled by the jump operators $\hat{L}_m = \sqrt{\gamma_1^{(m)}} \hat{a}_m$. They contribute

$$\left. \frac{\partial \rho}{\partial t} \right|_{\text{incoherent}}^{(1)} = \sum_m D_{\hat{L}_m, \hat{L}_m^\dagger}[\hat{\rho}] \quad (7.13)$$

where

$$D_{\hat{A}, \hat{B}}[\hat{\rho}] = \hat{A} \hat{\rho} \hat{B} - \frac{1}{2} \hat{B} \hat{A} \hat{\rho} - \frac{1}{2} \hat{\rho} \hat{B} \hat{A}. \quad (7.14)$$

We treat the single-photon loss rate as independent of the mode index, $\gamma_1^{(m)} = \gamma_1$.

We also model collisional losses. As argued in Appendix 7.9.1, one can engineer the system so that the dominant collisional loss channel involves a process where

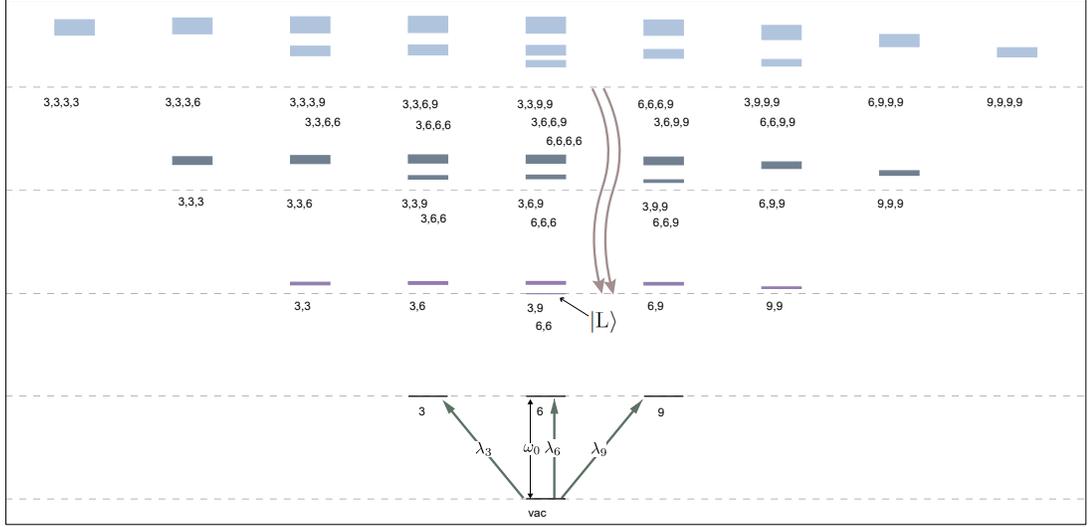


Figure 7.1: (color online) Schematic level diagram for three-mode system ($|3\rangle, |6\rangle, |9\rangle$) with no more than four polaritons. Vertical axis represents energy, and horizontal axis is the angular momentum relative to a state formed from atoms in the $|6\rangle$ mode: $L = 6N$. Dashed horizontal lines are separated by ω_0 , the mode frequencies in the absence of interactions. Numbers under each stack of levels represent the angular momenta of the polaritons in the system. For example, the states labeled with 3, 9 and 6, 6 are superpositions of the states $|3, 9\rangle$ and $|6, 6\rangle$. These are split by an energy proportional to the interactions U . The decay rate due to inelastic two-body collisions are represented by the width of each line. There are four long lived states: the vacuum, the single polariton states, and the two polariton Laughlin state. Green arrows are representative single-photon transitions, and brown wavy arrows indicate decay channels from the four photon manifold to the Laughlin state.

two proximate polaritons are simultaneously lost. This is described by

$$\left. \frac{\partial \rho}{\partial t} \right|_{\text{incoherent}}^{(2)} = \gamma_2 \int d^2r D_{\psi(r)\psi(r), \psi^\dagger(r)\psi^\dagger(r)}[\rho] \quad (7.15)$$

$$= \gamma_2 \sum_j \Lambda_{j_1 j_2}^{j_3 j_4} D_{\hat{a}_{j_1} \hat{a}_{j_2}, \hat{a}_{j_3}^\dagger \hat{a}_{j_4}^\dagger}[\rho]. \quad (7.16)$$

Crucially, the decay rate is proportional to the interaction energy, and states with zero interaction energy are infinitely long-lived. This means that in the absence of single particle loss the Laughlin state has an infinite lifetime. When restricted to three modes ($m = 3, 6, 9$), the only long-lived states will be the vacuum, the single-polariton states, and the two-photon Laughlin state in Eq. (7.6). All others

decay due to these two-polariton processes

We will use these incoherent processes as resources to produce the $\nu = 1/2$ Laughlin state by engineering a drive which will excite the system to the four-polariton manifold, relying on loss to populate the long-lived Laughlin state.

Figure 7.1 shows the energy eigenvalues of \hat{H} , with 4 or fewer polaritons. The vertical axis corresponds to energy, and the horizontal axis to angular momentum $L - 6N$. As depicted by the three arrows extending from the vacuum state, the λ_6 -drive couples states which are vertically separated, while the λ_3 and λ_9 drives cause diagonal transitions. The thickness of each energy level corresponds to the decay rate from the collisional two-polariton loss.

In this diagram, sets of lines have labels which correspond to the angular momenta of the polaritons in the state. For example, the states above the label 3,9 and 6,6 are the Laughlin state $|L\rangle \propto |66\rangle - \sqrt{2.1}|39\rangle$, and the state $|\bar{L}\rangle \propto \sqrt{2.1}|66\rangle + |39\rangle$. As denoted by dashed horizontal lines, in the absence of interactions, all states with N polaritons, would have energy $N\omega_0$, where ω_0 is the bare cavity frequency. The collisional decay rate is proportional to the interaction energy, so the linewidth grows as one moves away from the dashed line.

7.3 Numerical Techniques

We use two different numerical techniques to analyze the Lindblad equation. First, we numerically integrate Eq. (7.12), using a Runge Kutta finite difference scheme. This gives us the full time evolution of the $D \times D$ density matrix $\hat{\rho}$. In our full model, $D = 35$.

Alternatively, we vectorize the Lindblad equation interpreting $\hat{\rho}$ as a length 35^2 vector, $\vec{\rho}$. The Lindblad equation becomes $\partial\vec{\rho}/\partial t = \mathcal{L}\rho$, with

$$\begin{aligned} \mathcal{L} = & -iH \otimes \mathbb{1} + i\mathbb{1} \otimes H^T + \Gamma \sum_{\nu} L_{\nu} \otimes (L_{\nu}^{\dagger})^T \\ & - \frac{1}{2} \left(L_{\nu}^{\dagger} L_{\nu} \otimes \mathbb{1} + \mathbb{1} \otimes (L_{\nu}^{\dagger} L_{\nu})^T \right). \end{aligned} \quad (7.17)$$

The steady state corresponds to the kernel of the $35^2 \times 35^2$ matrix \mathcal{L} . We numerically find the null-space of \mathcal{L} using standard linear algebra libraries.

7.4 Validation

Before implementing our dissipative approach to generating the Laughlin state, we validate parts of our model by reproducing the experimental observations in [13]. In addition to giving us the opportunity to compare to experimental results, the simpler nature of this experiment lets us explicitly write down all expressions, making our arguments more concrete.

In [13] the experimentalists resonantly drive the $|6\rangle$ mode: $\omega_6 = \nu_6 - \omega_0 = 0$. The other two modes are undriven: $\lambda_3 = \lambda_9 = 0$. Since the states $|\text{vac}\rangle, |6\rangle$, and $|L\rangle$ are resonantly coupled, and the drive strengths are low, there is essentially zero probability to end up in any other state except these three, and the states they decay into: $|3\rangle$ and $|9\rangle$. It therefore suffices to truncate to the simplified model depicted in Fig. 7.2.

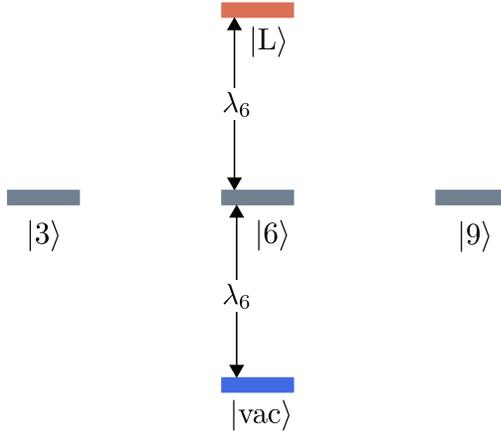


Figure 7.2: (color online) Energy-level diagram depicting the relevant states in the experiment of Clark et al. [13], analyzed in Sec. 7.4. A resonant drive connects $|\text{vac}\rangle$, $|6\rangle$ and $|L\rangle$. The modes $|3\rangle$ and $|9\rangle$ can become occupied through single-photon decay from $|L\rangle$.

In the rotating frame, the Hamiltonian is

$$H = \begin{bmatrix} 0 & 0 & \Omega & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ \Omega & 0 & 0 & 0 & \Omega' \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \Omega' & 0 & 0 \end{bmatrix} \quad (7.18)$$

where the basis states are $|\text{vac}\rangle, |3\rangle, |6\rangle, |9\rangle, |L\rangle$. The matrix elements are

$$\Omega = \langle \text{vac} | \lambda \hat{a} | 6 \rangle = \lambda \quad (7.19)$$

$$\Omega' = \langle 6 | \lambda \hat{a} | L \rangle = \sqrt{\frac{2}{3.1}} \lambda, \quad (7.20)$$

None of these states experience collisional losses, so we only need to consider

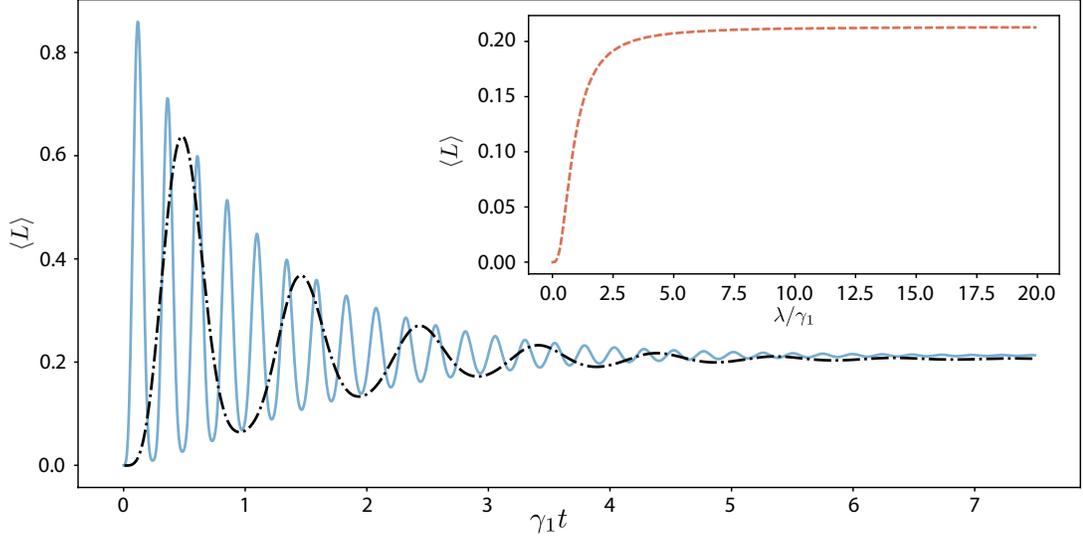


Figure 7.3: (color online) Time evolution of the probability of being in the Laughlin state for the scenario in Fig. 7.2. Here γ_1 is the single-polariton loss rate, and $\lambda = \lambda_6$ is the drives strength. Solid blue: $\lambda/\gamma_1 = 20$; Dashed-dotted black: $\lambda/\gamma_1 = 5$. Inset: Steady-state probabilities of the Laughlin state as a function of drive strengths.

the single-photon loss terms, corresponding to jump operators

$$\hat{L}_3 = |\text{vac}\rangle \langle 3| - \frac{\sqrt{2.1}}{\sqrt{3.1}} |9\rangle \langle L| \quad (7.21)$$

$$\hat{L}_6 = |\text{vac}\rangle \langle 6| + \frac{\sqrt{2}}{\sqrt{3.1}} |6\rangle \langle L| \quad (7.22)$$

$$\hat{L}_9 = |\text{vac}\rangle \langle 9| - \frac{\sqrt{2.1}}{\sqrt{3.1}} |3\rangle \langle L| \quad (7.23)$$

We numerically integrate the Lindblad equation,

$$\partial_t \hat{\rho} = i[\rho, H] + \gamma_1 (D_{L_3, L_3^\dagger} \rho + D_{L_6, L_6^\dagger} \rho + D_{L_9, L_9^\dagger} \rho), \quad (7.24)$$

Starting from the vacuum state at time $t = 0$. Figure 7.3 shows typical results for the time evolution of the probability of being in the Laughlin state $\langle L \rangle = \langle L | \hat{\rho} | L \rangle$.

If the drive is weak compared to the single-photon loss rate, $\lambda \ll \gamma_1$, then very few polaritons are ever in the cavity: As soon as a polariton is created, it decays. In the opposite limit $\lambda \gg \gamma_1$, damped Rabi oscillations are seen. As shown in the

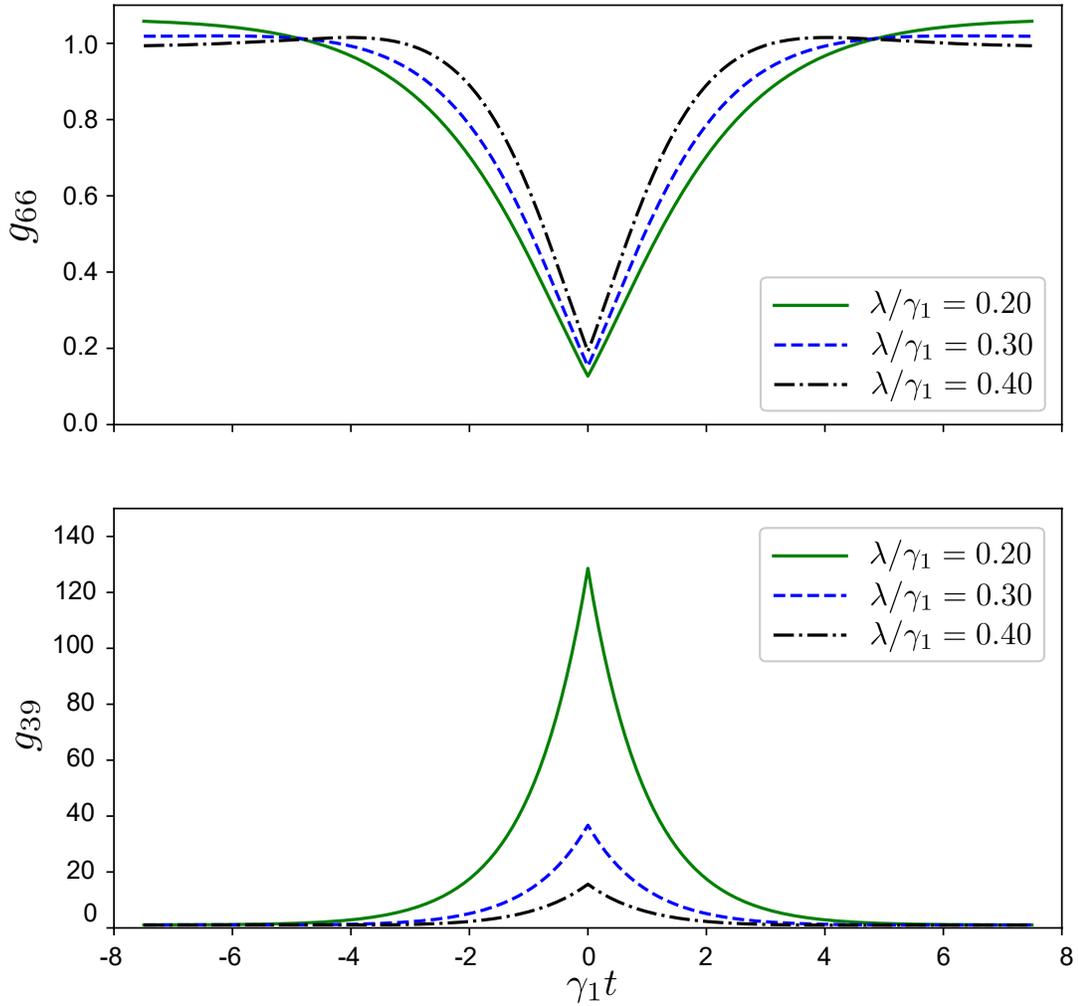


Figure 7.4: (color online) Correlation functions $g_{ij} = c_{ij}/\Gamma_i\Gamma_j$ corresponding to the scenario in Fig. 7.2. Here $c_{ij}(t)$ is the joint probability of emitting a photon into mode i and a second into j , with a time delay of t ; Γ_i is the steady state probability of emitting into state i . The single-photon decay rate is γ_1 , and the drive strength is λ .

inset, the steady-state population of the two-photon Laughlin state is a monotonic function of the drive amplitude, saturating at roughly 21% when the drive is very strong. In the experiment [13], $\lambda/\gamma_1 \sim 0.35$, and one therefore expects that the steady-state ensemble only has a 1.8% probability of being in the Laughlin state. To verify that they have produced the Laughlin state, the experimentalists use a correlation measurement: If two photons are simultaneously emitted from the

cavity, then the cavity must have been in a two-polariton state. Aspects of the two-photon state can be extracted from the mode-structure of the out-coming photons. This can be thought of as a form of post-selecting: “What is the quantum state when two polaritons are in the cavity?”

More precisely, the experimentalists measure the steady state correlation function $g_{jk}(\tau) = c_{jk}(\tau)/\Gamma_j\Gamma_k$, where $c_{jk}(\tau)$ is the joint probability of measuring a photon in mode j at time t and another in mode k at time $t + \tau$. This quantity is normalized by the the product of the probabilities of measuring each of these events separately, Γ_j and Γ_k . In steady-state the denominator is time independent.

Results from our numerical calculations are shown in Fig. 7.4. In the top panel one sees that sequential photons in the 6-mode are anti-correlated. At delay $\tau = 0$,

$$g_{66}(0) = \frac{(\gamma_{L \rightarrow 6})(\gamma_{6 \rightarrow L})P_L}{((\gamma_{6 \rightarrow \text{vac}})P_6 + (\gamma_{L \rightarrow 6})P_L)^2} = \frac{\frac{1}{3.1}P_L}{(P_6 + \frac{2}{3.1}P_L)^2}, \quad (7.25)$$

where P_6 and P_L are the steady state probabilities of being in the $|6\rangle$ or $|L\rangle$ states. The rate of emitting a 6 photon from $|L\rangle$ is $\gamma_{L \rightarrow 6} = 2\gamma_1/3.1$, while the rate for emitting from $|6\rangle$ is $\gamma_{6 \rightarrow \text{vac}} = \gamma_1$. For large delay, $g_{66}(\infty) = 1$, as the two events are uncorrelated. The photons are anticorrelated at short times because once a photon is emitted from the $|6\rangle$ state, a second cannot be emitted until another photon enters the cavity. For larger λ/γ_1 one sees weak oscillations in the correlation function, as Rabi oscillations between the states make certain delays less likely than others.

The photons in the $|3\rangle$ and $|9\rangle$ modes are positively correlated. Whenever a $|3\rangle$ photon is emitted, a $|9\rangle$ photon must follow at a later time. At zero time delay

$$\begin{aligned} g_{39}(0) &= \frac{(\gamma_{L \rightarrow 3})(\gamma_{3 \rightarrow \text{vac}})P_L}{((\gamma_{3 \rightarrow \text{vac}})P_3 + \gamma_{L \rightarrow 9}P_L)((\gamma_{9 \rightarrow \text{vac}})P_9 + \gamma_{L \rightarrow 3}P_L)} \\ &= \frac{\frac{2.1}{3.1}P_L}{(P_3 + \frac{2.1}{3.1}P_L)(P_9 + \frac{2.1}{3.1}P_L)}, \end{aligned} \quad (7.26)$$

where, as before, the rate of emission from i to j is $\gamma_{i \rightarrow j}$: $\gamma_{L \rightarrow 3} = \gamma_{L \rightarrow 9} = \gamma_1(2.1/3.1)$ and $\gamma_{3 \rightarrow vac} = \gamma_{9 \rightarrow vac} = \gamma_1$. Equation (7.26) can be somewhat simplified by using the principle of balance: $P_L \gamma_{L \rightarrow 3} = P_3 \gamma_{3 \rightarrow vac}$, and $P_L \gamma_{L \rightarrow 9} = P_9 \gamma_{9 \rightarrow vac}$. This yields

$$g_{39}(0) = \frac{1}{4} \frac{3.1}{2.1} \frac{1}{P_L}. \quad (7.27)$$

When P_L is small, then the time between emission of the individual photons in a 3-9 pair is short compared to the time between event pairs. This leads to a large correlation peak. At long times $g_{39}(\infty) = 1$, as one is detecting photons from different pairs. The correlation function should just fall exponentially between these values, with decay time τ_1 .

To numerically calculate $g_{jk}(\tau)$, we first integrate Eq. (7.24) for a long time to produce the steady state density matrix $\hat{\rho}_\infty$. We calculate the emission rates $\Gamma_j = \gamma_1 \text{Tr} \hat{a}_j^\dagger \hat{a}_j \hat{\rho}_\infty = \text{Tr} \hat{a}_j \hat{\rho}_\infty \hat{a}_j^\dagger$. The density matrix immediately following a j -photon emission event is $\hat{\rho}_j = \hat{a}_j \hat{\rho}_\infty \hat{a}_j^\dagger / \Gamma_j$. We then evolve this density matrix for time τ to calculate $c_{jk} = \Gamma_j \text{Tr} \hat{a}_k^\dagger \hat{a}_k \hat{\rho}_j(\tau)$, and $g_{jk} = c_{jk} / \Gamma_j \Gamma_k$.

Another important probe in [13] is to compare the probability of simultaneously observing a 3 and 9 photon, to that of finding two 6 photons,

$$R = \frac{c_{39}(0) + c_{93}(0)}{c_{66}(0)}. \quad (7.28)$$

Within this truncated model, the only contributions to the correlation functions come from the diagonal element of $\hat{\rho}$ corresponding to the Laughlin state,

$$R = \frac{P_L \langle L | a_3^\dagger a_9^\dagger a_9 a_3 | L \rangle + P_L \langle L | a_9^\dagger a_3^\dagger a_3 a_9 | L \rangle}{P_L \langle L | a_6^\dagger a_6^\dagger a_6 a_6 | L \rangle} \quad (7.29)$$

$$= 2.1. \quad (7.30)$$

The experimental results are within error bars of this number, which is consistent with producing the Laughlin state.

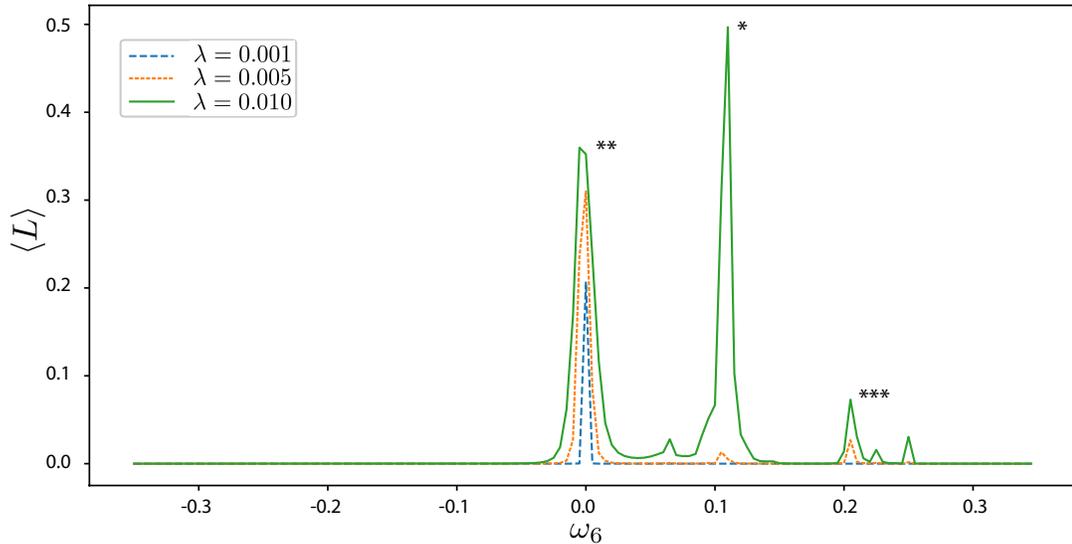


Figure 7.5: (color online) Steady state probabilities of the Laughlin state as a function of the detuning. Each of the three modes are driven with equal amplitude: $\lambda_3 = \lambda_6 = \lambda_9 = \lambda$. The $m = 6$ detuning, ω_6 is displayed on the horizontal axis, and the $m = 3$ and $m = 9$ drives are chosen such that the four-body state $|A\rangle = -0.362|3339\rangle - 0.932|3366\rangle$ is on resonance. The energy level diagram corresponding to this drive condition is depicted in Fig. 7.6. For reference, the three highest peaks are labeled by different numbers of asterisks.

7.5 Driven-Dissipative Preparation of the 2-polariton Laughlin State

As illustrated by the model in Sec. 7.4, in the absence of careful timing, one cannot efficiently produce the Laughlin state by resonant coupling. This is a generic feature of coherent quantum systems: For example, resonantly coupling the levels of a two-level system will not create a steady-state inverted population. On the other hand, a laser can be made from a 3-level atom. There one produces a large occupation of an excited metastable state by driving to a third level which decays into it. We will explore the many-body analog of this approach. Four-polariton states play the analog of the unstable excited level, and the Laughlin state plays

the role of the metastable excited level. Collisional two-polariton loss provides the decay from the four-polariton manifold to the Laughlin state.

The challenge in designing this protocol is that many levels are involved. Figure 7.1 shows a complicated network of 35 levels. Each N polariton states couple to three states with $N+1$ polaritons. This gives 60 distinct transition matrix elements. One needs to be able to efficiently couple the vacuum state to the four-photon manifold without coupling the Laughlin state to any other level. Moreover there are a large number of parameters, including the amplitude and frequencies of the three drives, the interaction strength, and the various decay constants.

The challenge would be greatly reduced if we had a technique to coherently inject 4-photons, directly coupling the vacuum state to the four-polariton manifold. Appendix 7.9.2 analyzes that simplified case. Here we tackle the full problem, where the drive is of the form in Eq. (7.11).

Exciting the system into the 4-polariton manifold is a fourth order process – which will be very strongly suppressed when the drive is weak. One cannot simply increase the drive strength, as a strong drive will also deplete the Laughlin state. Our strategy for overcoming this difficulty is to engineer a string of intermediate states. Ideally the photons would be absorbed through four resonant (or near resonant) transitions. Simultaneously we need the Laughlin state to be spectrally isolated.

We adjust the detunings ω_3 , ω_6 , and ω_9 to optimize this process. As previously explained, in order to have a well-defined rotating frame, we require $\omega_3 + \omega_9 = 2\omega_6$. Furthermore, we require that the four-photon transition is resonant. There are ten 4-polariton states which have a finite probability to decay into the Laughlin state.

As a concrete example, consider $|A\rangle = -0.362|3339\rangle - 0.932|3366\rangle$, with interaction energy $E_A = 1.245U$. To resonantly couple to this state, we need $3\omega_3 + \omega_9 = 2\omega_3 + 2\omega_6 = E_A$. Throughout we use units where $U = 1$ and take $\gamma_2 = 0.001$. We neglect single-polariton loss, taking $\gamma_1 = 0$.

Figure 7.5 illustrates the role of resonances by plotting the steady-state probability of being in the Laughlin state as a function of ω_6 , calculated by the technique in Sec. 7.3. Here we fix $\omega_3 = E_A/2 - \omega_6$, so that the state $|A\rangle$ remains degenerate with the vacuum state, in the rotating frame. One sees at least six discrete peaks, corresponding to when various intermediate states become resonant. The most prominent peaks correspond to the simultaneous alignment of multiple intermediate states. Figure 7.6 shows energy level diagrams corresponding to the three largest peaks. These depict the rotating-frame energy of the states from Fig. 7.1, separated into columns, each of which represents a different number of polaritons. As illustrated by the arrows, the drive changes the polariton number by 1, and hence connects states in neighboring columns. Two-polariton collisional loss causes an incoherent decay into a state which is two columns to the left. Energy conservation is relevant to the coherent drive processes, but not the loss processes. As before, the line-widths are shown by the thickness of the lines. These three energy level diagrams illustrate the central design principles behind our approach.

The most important feature of the level diagrams in Fig. 7.6 is the fact that there is a direct chain of near-resonant excitations that takes one to the four-polariton manifold. The closer these intermediate states are to resonance, the higher the effective transition rate to the four-polariton state. The second most important feature is how spectrally isolated the Laughlin state is. The reason that the peak marked (*) is higher than the one marked (***) is that the Laughlin state

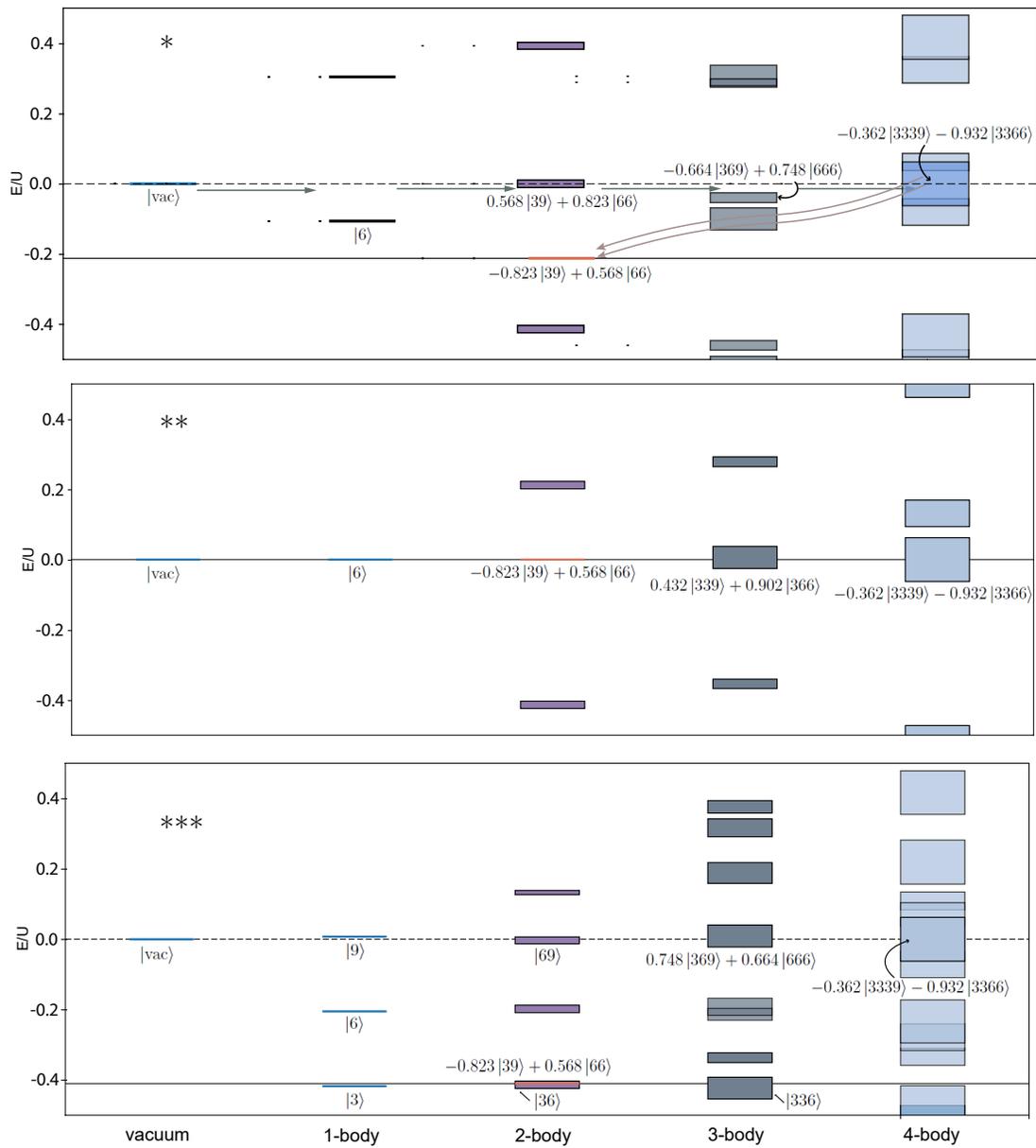


Figure 7.6: (color online) Rotating frame energy level diagrams corresponding to the three most prominent peak Fig. 7.5, marked by (*) top, (**) middle, and (***) bottom. As illustrated by horizontal arrows in the top panel, a single-photon drive sequentially couples states which differ by one polariton. The Laughlin state is populated by two-body loss from the four-body manifold (denoted by the double arrow).

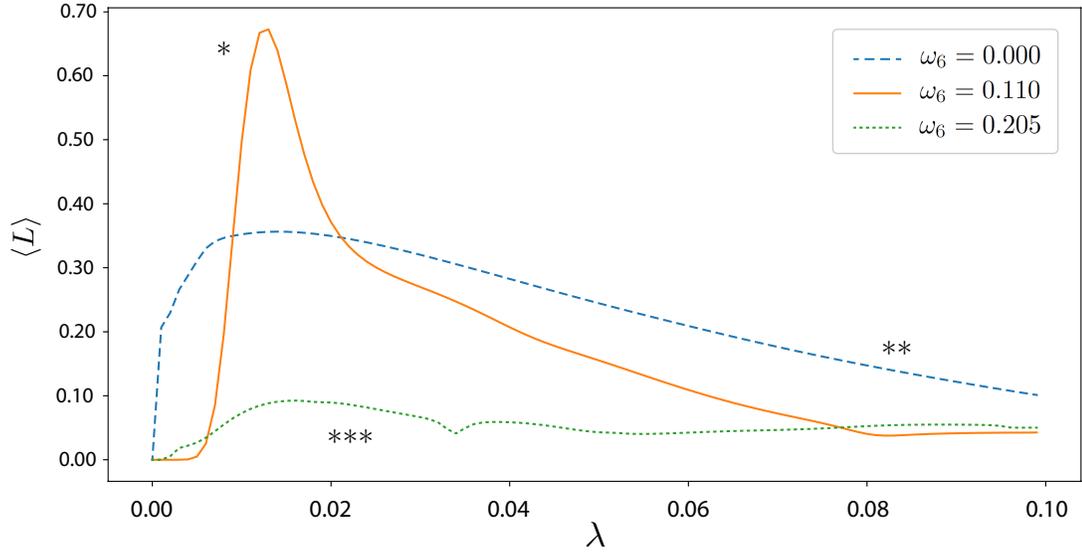


Figure 7.7: (color online) Steady state probability of the Laughlin state for the three largest peaks in Fig. 7.5 as a function of drive amplitude $\lambda_3 = \lambda_6 = \lambda_9 = \lambda$ for $\gamma_2 = 0.001$.

is more isolated. One is led to the intuitive recipe: Align the the levels needed to absorb photons, but keep the Laughlin state isolated.

The second highest (**) peak in Fig. 7.5 is qualitatively different than the others, as the Laughlin state is coherently coupled to the vacuum. In that sense it is more similar to the scenario in Sec. 7.4, and it shares the qualitative features of that model.

There are however two differences. First, in this section we are driving with λ_3 , λ_6 , and λ_9 , while in Sec. 7.4 only λ_6 was nonzero. The three drives add coherently, and result in a somewhat larger population of the Laughlin state. The other difference between this scenario and Sec. 7.4 is that there are 3-polariton and 4-polariton states which are also resonant. The minimal model here requires including these two additional states, resulting in a 7 dimensional Hilbert space.

As shown in Fig. 7.7, the probability of ending up in the Laughlin state is

a non-monotonic function of the drive-strength. For weak drive, the bottle-neck is populating the four-photon manifold. In this regime, the population of the Laughlin state increases with drive strength. The rate of increase depends on the accuracy of the alignment of the intermediate states: Thus for small λ the solid red curve (corresponding to the peak with a single asterisk in Fig. 7.5) rises slower than the dotted green curve (corresponding to three asterisks). As should be apparent, the physics of the dashed curve is somewhat different, as the Laughlin state is resonantly excited.

If the drive is made too strong, the probability of ending up in the Laughlin state falls. This occurs because the drive excites the system out of the Laughlin state.

The story is somewhat complicated by the fact that there are three different λ 's. Figure 7.8 further explores the amplitude dependence of the (*) peak in Fig. 7.5. We set $\lambda_9 = 0$ and vary λ_3 and λ_6 . The most prominent feature is a peak at $\lambda_6 = 0.010, \lambda_3 = 0.013$, and a ridge which extends out from it at an angle. This ridge is a quantum interference effect: For a certain ratio of λ_6/λ_3 , the excitations to the four-polariton manifold are enhanced. Due to the presence of many different intermediate states, and many states which couple to the Laughlin state, the structure can be quite rich: For example, the dotted green curve in Fig. 7.7 has a dip near $\lambda = 0.035$, and the red curve has several kink-like features.

We systematically explored all ten potential four-photon transitions, producing the analog of Fig. 7.5 for each of them. For each of the large peaks we numerically optimized the values of $\lambda_3, \lambda_6, \lambda_9$ and ω_6 . The most favorable example is the one in Fig. 7.6 which resulted in a steady-state occupation of the Laughlin state of 83.9% with $\lambda_6 = 0.0086, \lambda_3 = 0.0129$, and $\lambda_9 = 0.0014$.

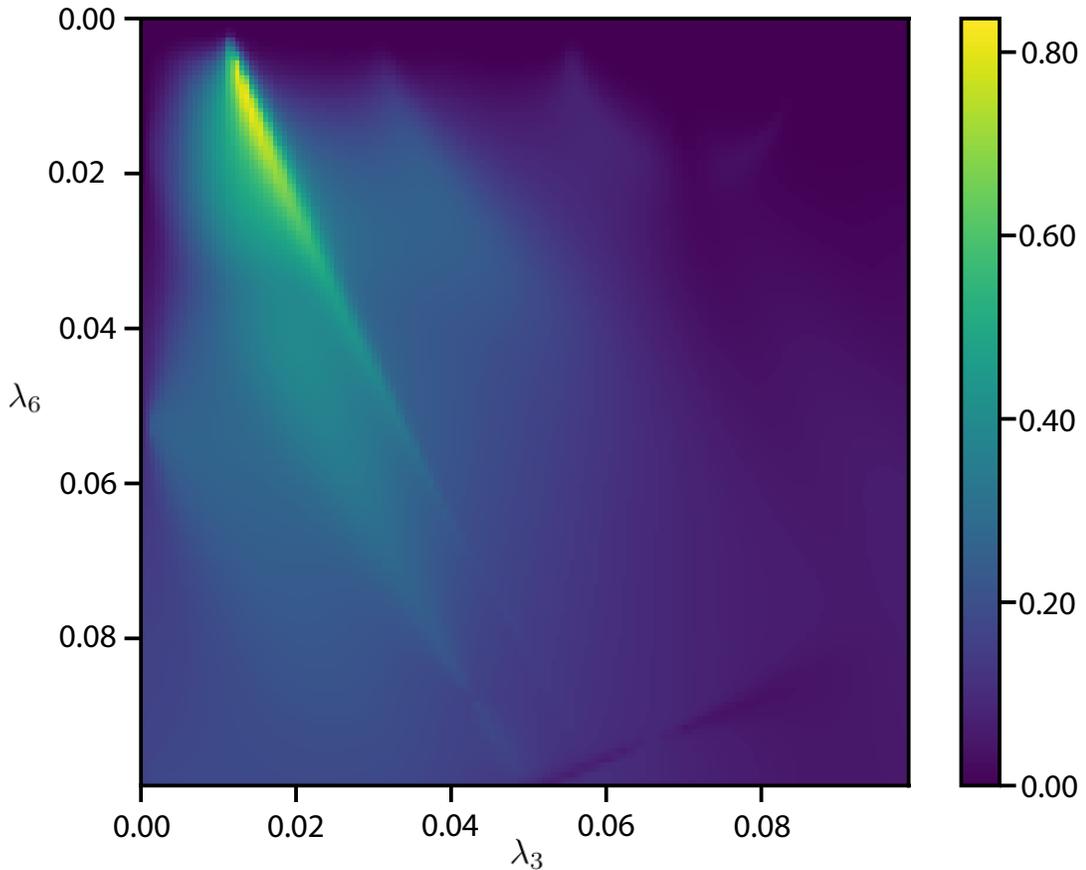


Figure 7.8: (color online) Steady state probability of being in the Laughlin state for detuning $\omega_6 = 0.1064$ (corresponding to the (*) state in Fig. 7.5 as a function of λ_3 and λ_6 drive amplitudes with $\lambda_9 = 0$. The peak probability is 83.7% at $\lambda_6 = 0.010$ and $\lambda_3 = 0.013$.

7.6 Measurements

The most easily interpreted probe of our system is monitoring the light emitted by the two-polariton loss events. No such events occur in the Laughlin state, and the disappearance of this signal is a direct measurement of the short range polariton-polariton correlation function. Analogous correlation measurements were carried out in atomic systems [125]. The challenge of this measurement is that the light is presumably emitted in random directions. Furthermore, this signal cannot

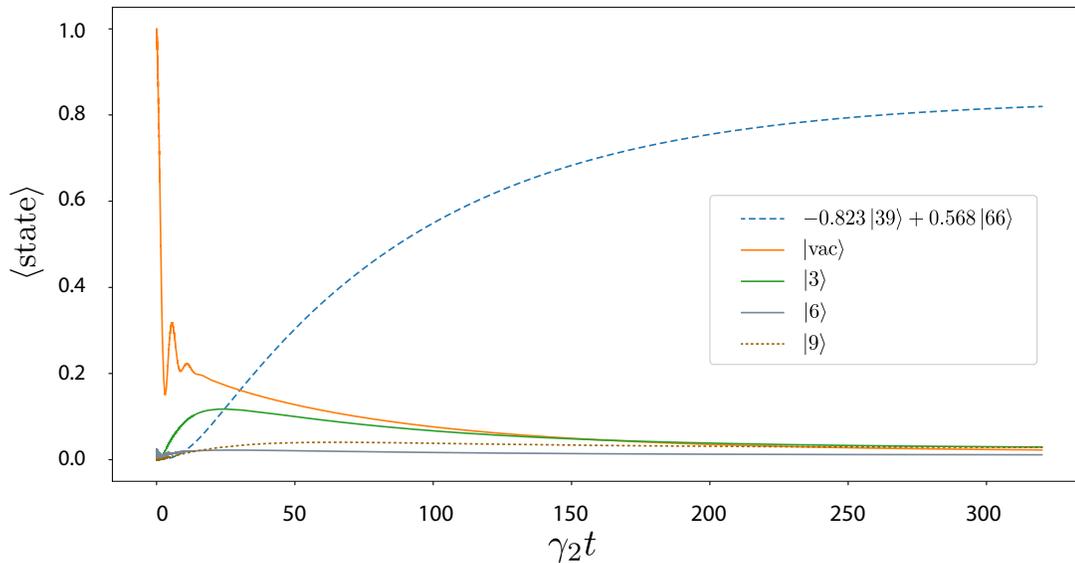


Figure 7.9: (color online) Probability of the long-lived states $|\text{vac}\rangle$, $|3\rangle$, $|6\rangle$, $|9\rangle$ and $|L\rangle$ as the system in the (*) configuration evolves according to the Lindblad master equation. The time scale for the finite evolution is in terms of the two-body loss rate. The evolution uses optimized parameters for this configuration. The system configuration at long times contains a non-zero population of the long-lived single-body states.

distinguish between different dark states.

A slightly more indirect, but simpler measurement is monitoring the reflected light from the drive. Any photons which enter the cavity show up as an attenuation of this reflected light. Thus when the system reaches the Laughlin state, which is dark, the reflected intensity increases. Again, any dark state would give the same signal.

A third approach is to monitor the intensity and correlations of the light emitted through single-polariton losses, as was discussed in Sec. 7.4. Since this probe is harder to interpret, it is not as useful as a “smoking gun.” Its primary advantage is that the measurement is relatively straightforward.

7.7 Summary and Discussion

We presented a driven dissipative approach to producing the two-particle Laughlin state of polaritons in a twisted cavity. This is part of a broader goal of producing and braiding anyons in an AMO system.

Producing the two-particle Laughlin state is a modest, but necessary step towards this goal. Our technique is a useful complement to existing proposals, and it brings with it a number of pros and cons.

Most importantly, our technique is self-correcting: If a perturbation (such as decay of a polariton) knocks the system out of the Laughlin state, the driven-dissipative process will bring you back. It also does not rely upon any careful timing.

Unfortunately, our technique does require fine-tuning the of the drive frequencies and amplitudes. In an experiment, this fine-tuning would need to be done by trial and error, as our models have finite accuracy. These experiments also require a hierarchy of energies which may be difficult to attain – namely: $\gamma_1 \ll \lambda, \gamma_2 \ll U$. Here γ_1 is the rate of single-photon loss, λ is the drive strength, γ_2 is the rate of interaction-driven two-photon loss, and U is the interaction strength. The most challenging aspect, requiring novel cavities, is making γ_1 smaller than the other scales. The relative size of γ_2 and U can be controlled by adjusting the detuning of the EIT transition used for the Rydberg polaritons (Appendix 7.9.1). The drive strengths λ are under direct experimental control.

A crucial question is the scalability of the approach. The biggest impediment here is that as one adds more states, the energy-level spectrum becomes

quite dense. Consequently even more fine-tuning would be needed to create larger Laughlin states. Nonetheless, this driven-dissipative approach is a valuable tool for state creation and stabilization.

7.8 Acknowledgments

We thank Jon Simon and his group for fruitful discussions. This work was supported by NSF PHY-1806357.

7.9 Appendix

7.9.1 Microscopic Model of Interacting Rydberg Polaritons in Nearly Degenerate Multimode Cavity

For completeness we give a brief derivation of the model from Sec. 7.2. We make no attempt at mathematical rigor – more thorough treatments can be found elsewhere [168–172].

The modes of an optical cavity are labeled by a longitudinal index k , and transverse indices α . In the image plane, where the atoms sit, the electric field has magnitude $E_\alpha^{(k)}(x, y)$. We are considering the case where several modes with the same k are nearly degenerate. We consider only those modes, and drop the longitudinal index in our expressions. For simplicity we will assume that the E_α form a complete set of states in the $x - y$ plane. If they do not, we can always formally add the missing states, giving them an infinite energy.

The operator which creates a photon in mode α is $\hat{\chi}_\alpha$, and can define an operator which creates a maximally focused beam as

$$\hat{\chi}(x, y) \propto \sum_{\alpha} E_{\alpha}(x, y) \hat{\chi}_{\alpha}. \quad (7.31)$$

By the completeness property, these obey the standard Bose commutation relations. The Hamiltonian can be expressed as

$$\hat{H}_{\gamma} = \sum_{\alpha} \epsilon_{\alpha} \hat{\chi}_{\alpha}^{\dagger} \chi_{\alpha} \quad (7.32)$$

$$= \int dr dr' \hat{\chi}^{\dagger}(r) H_{\chi}(r, r') \hat{\chi}(r'), \quad (7.33)$$

which defines the kernel H_{χ} .

At every point in space we envision a large number of 3-level atoms, with ground state $|g\rangle$, a short lived excited state $|p\rangle$, and a long-lived excited Rydberg state $|s\rangle$. The cavity photons couple the states $|g\rangle$ and $|p\rangle$, and a control beam couples $|p\rangle$ and $|s\rangle$. We introduce operators $\hat{\phi}_s(r)$ and $\hat{\phi}_p(r)$ that change the number of excitations at r , and work in a rotating frame. The Hamiltonian describing these degrees of freedom are

$$\hat{H}_r = \int dr \left[g(\hat{\chi}^{\dagger} \hat{\phi}_p + \hat{\phi}_p^{\dagger} \hat{\chi}) + \Omega(\hat{\phi}_s^{\dagger} \hat{\phi}_p + \hat{\phi}_p^{\dagger} \hat{\phi}_s) + \Delta \hat{\phi}_p^{\dagger} \hat{\phi}_p \right] \quad (7.34)$$

Here $\Delta = \delta - i\gamma$, where $\delta = E_p - E_s + \omega_c$ is the detuning of the p-state from resonance and $\tau = 1/(2\gamma)$ is the lifetime of that state. Here E_p and E_s are the energies of the states in the lab frame, and ω_c is the frequency of the control laser. The Rabi frequencies from the cavity and control lasers are g and Ω . Since this Hamiltonian is strictly local, we have left off the position arguments in the operators.

It is convenient to transform to a basis which diagonalizes Eq. (7.34),

$$\hat{\psi} \propto \Omega \hat{\chi} - g \hat{\phi}_s \quad (7.35)$$

$$\hat{\phi}_+ \propto g \hat{\chi} + E_+ \hat{\phi}_p + \Omega \hat{\phi}_s \quad (7.36)$$

$$\hat{\phi}_- \propto g \hat{\chi} + E_- \hat{\phi}_p + \Omega \hat{\phi}_s \quad (7.37)$$

These represent the long-lived dark polaritons, with energy $E_{\text{dark}} = 0$ and the short-lived bright polaritons with energies

$$E_{\pm} = \frac{\Delta \pm \sqrt{\Delta^2 + 4g^2 + 4\Omega^2}}{2}. \quad (7.38)$$

The relationships in Eq. (7.35) through (7.37) can be inverted to write

$$\hat{\chi} = \frac{\Omega}{\sqrt{\Omega^2 + g^2}} \hat{\psi} + \eta_+ \hat{\phi}_+ + \eta_- \hat{\phi}_- \quad (7.39)$$

where $\eta_{\pm} = g/\sqrt{\Omega^2 + g^2 + |E_{\pm}|^2}$. We substitute this into Eq. (7.33). There are nine terms, but the bright polaritons are far off resonance. Keeping only the terms involving the dark polaritons,

$$\hat{H}_{\gamma} = \int dr dr' \hat{\psi}^{\dagger}(r) H_{\psi}(r, r') \hat{\psi}(r'), \quad (7.40)$$

where

$$H_{\psi}(r, r') = \frac{\Omega^2}{\Omega^2 + g^2} [H_{\chi}(r, r') - \omega_0 \delta(r - r')] \quad (7.41)$$

and $\omega_0 = E_p - E_g - \omega_c$ is the difference between the Rydberg excitation energy and the control frequency. It appears due to transforming into the rotating frame. It simply shifts the entire spectrum, and has no physical consequences.

Our truncation is only valid if the spectrum of $H_{\chi} - \omega_0 \mathcal{I}$ is small compared to the splitting between polaritons $\sim \sqrt{\Omega^2 + g^2}$. The influence of the discarded modes can be treated perturbatively [168].

The Rydberg atoms interact via a dipole-dipole interaction,

$$\hat{H}_{\text{int}} = \int dr dr' V(r - r') \phi_s^\dagger(r) \phi_s^\dagger(r') \phi_s(r') \phi_s(r), \quad (7.42)$$

where the interaction is typically modeled as $V(r) = C_6 r^{-6}$ – though the exact potential is more complicated [173]. The challenge here is that V can be much larger than any of the other scales, and hence mixes in the other polariton modes. It is large, however, only in a small region of space. Thus its effect on the modes is captured by a zero-range potential

$$\hat{H}_{\text{int}}^{\text{eff}} = U \int dr \psi^\dagger(r) \psi^\dagger(r) \psi(r) \psi(r). \quad (7.43)$$

The coefficient U is found by matching the low-energy scattering phase shifts, for example by summing a set of ladder diagrams [169]. Due to mixing in the dark polariton states, these interactions generically contain an inelastic piece. As we argue below, when δ is small, both of the polaritons in the collision are lost. In a Lindblad formalism, this corresponds to the jump operator in Eq. (7.15).

While the full analysis of the scattering problem is tedious, the central physics is apparent by considering two polaritons at fixed locations, r and r' . The Hilbert space is then 9-dimensional, as all three flavors of polaritons will be mixed. We are considering the case where δ is small, so the state with two dark polaritons $|dd\rangle$ is nearly degenerate with the state containing one of each flavor of bright polaritons $|+-\rangle$. When V is small compared to $\sqrt{\Omega^2 + g^2}$, we can truncate to these two states. In the basis $|+-\rangle, |dd\rangle$, the local Hamiltonian is then

$$H_{\text{local}} \approx \begin{pmatrix} V\Upsilon_{\pm\pm} + \Delta & V\Upsilon_{d\pm} \\ V\Upsilon_{d\pm} & V\Upsilon_{dd} \end{pmatrix} \quad (7.44)$$

with

$$\Upsilon_{\pm\pm} = \frac{1}{2} \frac{\Omega^4}{(\Omega^2 + g^2)^2} \quad (7.45)$$

$$\Upsilon_{dd} = \frac{g^4}{(\Omega^2 + g^2)^2} \quad (7.46)$$

$$\Upsilon_{d\pm} = \frac{1}{\sqrt{2}} \frac{g^4}{(\Omega^2 + g^2)^2}. \quad (7.47)$$

If $V = 0$ the state $|dd\rangle$ is an eigenstate, with energy 0. For non-zero interactions the state continuously connected with that one has an admixture of $|+-\rangle$ and its energy is found by solving a quadratic equation. For $V \ll |\Delta|$, the mixing is weak, and the potential that the polariton feels is just the Rydberg-Rydberg coupling scaled by the overlap with the Rydberg state

$$E_{\text{local}}^{\text{weak}} = V \frac{g^4}{(\Omega^2 + g^2)^2}. \quad (7.48)$$

When the interaction is strong, $V \gg |\Delta|$, the eigenstate is independent of V and the effective interaction is proportional to $\Delta = \delta + i\gamma$,

$$E_{\text{local}}^{\text{strong}} = \Delta \frac{2g^4}{2g^4 + \Omega^4}. \quad (7.49)$$

Within this framework, the fact that this energy is complex indicates there is particle loss. Since the loss comes from coupling to the state with two bright polaritons, two particles are lost.

As argued in [168], when V is sufficiently large, one will also have collisional loss events where only one of the polaritons are lost. These will dominate if $\delta \gg \sqrt{\Omega^2 + g^2}$. Thus for our driven-dissipative state preparation scheme to work, we require that we are in the regime where δ is small.

7.9.2 Multi-photon Drive

Our protocol faces two challenges: (1) efficiently exciting from the vacuum state to the metastable 4-polariton manifold, and (2) avoiding driving the system out of the Laughlin state. Both of these difficulties would be avoided if we could directly inject 4 photons into the system, for example with a drive term of the form

$$H_4 = \lambda \hat{a}_6^\dagger \hat{a}_6^\dagger \hat{a}_6^\dagger \hat{a}_6^\dagger + \text{c.c.} \quad (7.50)$$

Under those circumstances the Laughlin state is a true dark state (in the absence of single-particle loss). In this Appendix we briefly explore the dynamics under this hypothetical drive, contrasting it with a two-photon drive

$$H_2 = \lambda \hat{a}_6^\dagger \hat{a}_6^\dagger + \text{c.c.} \quad (7.51)$$

and the single-photon drive used in the main text. In this latter case, we take $\lambda_3 = \lambda_9 = 0$.

As in the main text, we use units where $U = 1$, and take $\gamma_1 = 0$, but we use a significantly stronger 2-body decay rate $\gamma_2 = 0.1$. The latter sets the characteristic scale for the incoherent transitions from the four-polariton manifold to the Laughlin state, and the widths of various resonances.

There are three different four-polariton states which can be excited by these drives. They are resonant when $\omega_6 = 0.31, 0.19, 0.12$. Figure 7.10 shows the probability of being in the Laughlin state after a time $\gamma_2 t = 1000$, for different drive strengths. We choose to look at the finite-time probabilities, as in our 4-photon drive model the Laughlin state is a true dark state, and if one waits long enough the system will have a 100% chance of occupying that state.

The top panel of Fig. 7.10 shows results for the four-photon drive. The dominant feature is a large peak near $\omega_6 = 0.19$. This corresponds to the transition with the largest coupling matrix element. Smaller peaks are visible near the other resonances. As the drive amplitude increase, the maximum probability saturates at 100%, and spreads into a plateau.

The two-photon drive (middle panel) shows the same three peaks, with an additional feature closer to $\omega_6 \approx 0.1$. This corresponds to a two-photon resonance. Generically the probabilities of occupying the Laughlin state are smaller for the two-photon drive than for the four-photon drive. Note that interactions in the intermediate two-polariton states mix the different polariton flavors, enhancing the $\omega_6 = 0.12$ compared to what is seen in the top figure.

The single-photon drive (bottom panel) shows essentially zero occupation of the Laughlin state, unless the drive is very strong. This is because with only the λ_6 drive there is no way to arrange a set of resonant intermediate states. The location of the peak that appears near $\omega_6 \approx 0.09$ for very strong drive is a compromise between the final-state and intermediate-state resonances.

7.9.3 Lindblad Superoperator

Sometimes we would like to create a simpler problem from a more complicated one. In our case, we are interested in a subspace of solutions to the Lindblad evolution, i.e. the steady-state solutions. Solutions to an eigenvalue problem where the eigenvalues are zero would produce such a subspace. In this section, we show how to transform the Lindblad equation to a linear transformation using some tensor network algebra.

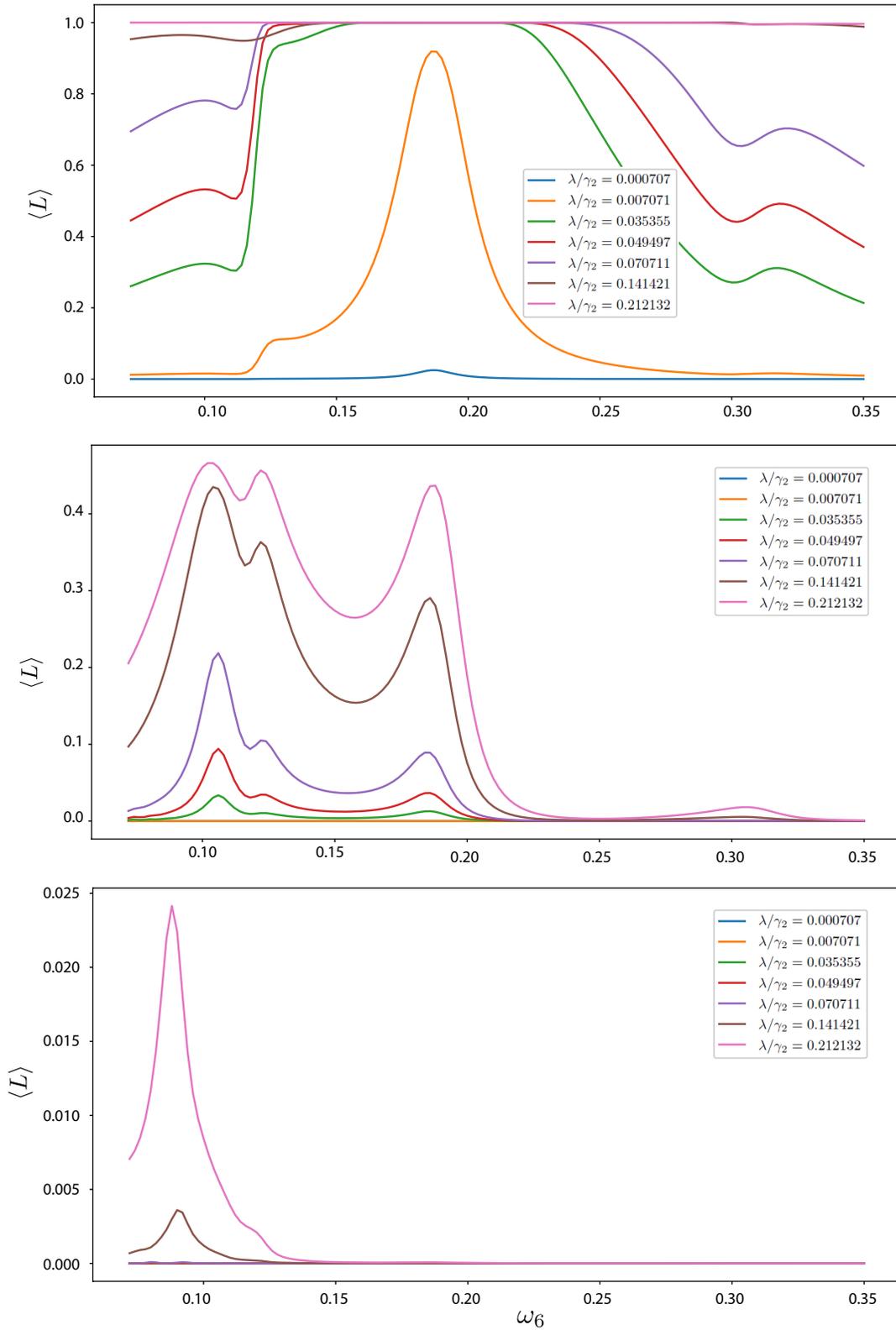


Figure 7.10: (color online) Probability of occupying the Laughlin state at time $t = 1000/\gamma_2$ for four-photon (top), two-photon (middle), and single-photon (bottom) drive. In all cases only the $m = 6$ mode is driven, and $\gamma_2 = 0.10$. The different curves correspond to various drive amplitudes λ relative to the two-body loss rate γ_2 .

We would like to create the linear algebra problem, $\frac{\partial \rho}{\partial t} = \mathcal{L}(t)\rho$ and find where $\frac{\partial \rho}{\partial t}$ is zero. To equip ourselves, we should first look at some simple examples of tensor network algebra.

Let us start by looking at an example of a matrix, A_{ik} , and its transpose, $(A_{ik})^T = A_{ki}$, see Fig. 7.11a. A matrix is a rank-2 tensor, and it has two indices labeled with arrows that denote the direction of operation. The transpose is created by switching the direction of the arrows.

Now let us look at an example of a tensor contraction, see Fig. 7.11b. We have the usual representation for matrix multiplication, where the matrix A is formed from the tensor contraction of matrices B and C over the index j , i.e. $A_{ik} = \sum_j B_{ij}C_{jk}$.

One useful technique is knowing how to represent a matrix as a vector. Fig. 7.11c shows the matrix A represented as a matrix on the left as a vector on the right. We show how the elements transform in Fig. 7.11d.

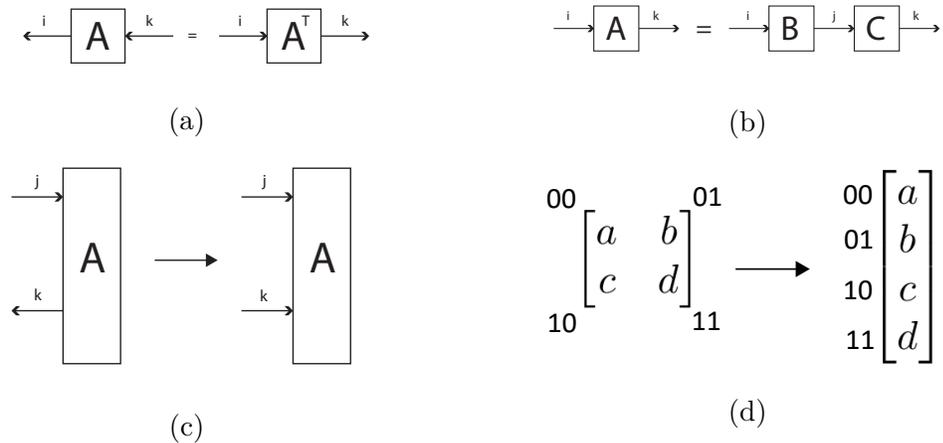


Figure 7.11: Some useful tensor network identities.

We are now in a place to form the linear algebra problem using the techniques in Fig. 7.11.

The first term in the Lindblad equation, Eqn. 7.12, is the commutator $[H(t), \rho] = H(t)\rho - \rho H(t)$. The first term in the commutator, $H(t)\rho$, has the ρ on the right. Transforming this term into the form we need is simple – we vectorize ρ and change the shape of $H(t)$ accordingly, see Fig. 7.12. Transforming the second term, $\rho H(t)$, requires a bit more manipulation.

Let us start with the first term. We simply need to write $H(t)\rho$ in terms of tensors and contractions, see Fig. 7.12a. We vectorize ρ in Fig. 7.12b by moving the k index of ρ to the left side and flip its direction (refer to Figs. 7.11c and Fig. 7.11d). The identity is inserted as a placekeeper, so we have $H(t) \otimes \mathbb{1}\rho$, see Fig. 7.12c.

Now we can transform the second term from the commutator, $\rho H(t)$, see Fig. 7.13. We start by writing the $\rho H(t)$ term as a tensor network, see Fig. 7.13a. We rearrange the indices on ρ and insert the identity on the top index. Then, we take the transpose of $H(t)$. This vectorizes ρ , see Fig. 7.13b. Finally, we can express the tensors as $\mathbb{1} \otimes H^T(t)\rho$, see Fig. 7.13c.

Terms involving the jump operators, L_j , are manipulated in a similar fashion, see Fig. 7.14. Fig. 7.14a shows the manipulation $L_j^\dagger \rho L_j \rightarrow L_j \otimes \left(L_j^\dagger\right)^T \rho$. Fig. 7.14b shows the manipulation $L_j^\dagger L_j \rho \rightarrow L_j^\dagger L_j \otimes \mathbb{1}\rho$. Fig. 7.14c shows the manipulation $\rho L_j^\dagger L_j \rightarrow \mathbb{1} \otimes \left(L_j^\dagger L_j\right)^T \rho$.

We now have the superoperator form for the Lindblad operator,

$$\mathcal{L}(t) = -iH(t) \otimes \mathbb{1} + i\mathbb{1} \otimes H^T(t) + \sum_j L_j \otimes \left(L_j^\dagger\right)^T - \frac{1}{2} \left(L_j^\dagger L_j \otimes \mathbb{1} + \mathbb{1} \otimes \left(L_j^\dagger L_j\right)^T \right). \quad (7.52)$$

It is interesting to note that quantum circuit diagrams are often represented as simplified examples of these more general tensor networks, where the length of

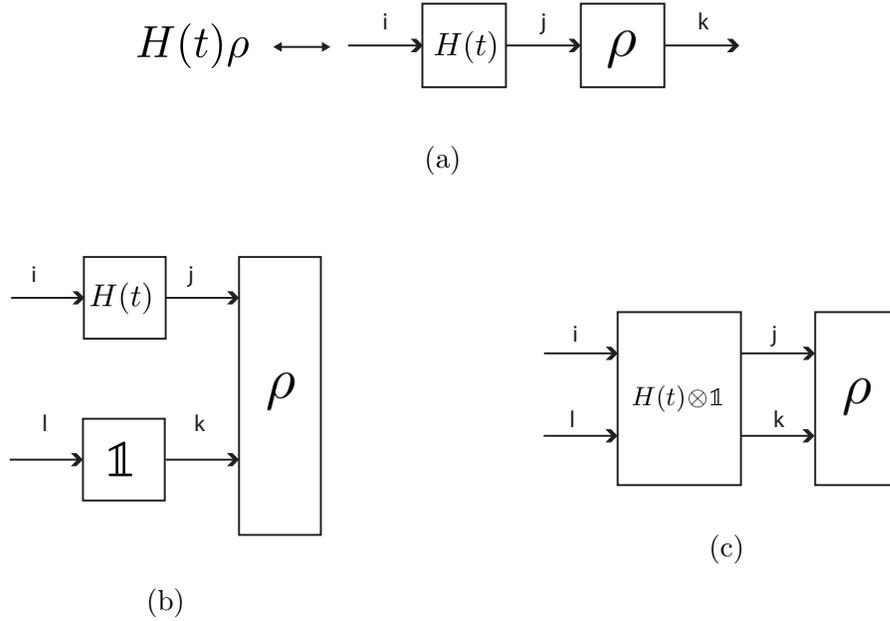


Figure 7.12: Tensor network manipulation of the term $H(t)\rho$ to obtain the form $H(t) \otimes \mathbb{1}$.

the indices on the tensors are two for qubits, three for qudits, and so on. Operations between different quantum channels or between quantum bits, for example controlled-NOT gates, are superoperators! Also note that the tensor network identities do not change additive properties, they change only the form of the contractions.

7.9.4 A Note about Long-lived Rydberg States

In this section, we have a short heuristic note on the lifetime of Rydberg excitations.

Intuitively one might think that a highly-excited electron would decay quickly into a lower-energy orbital. The difference in energy between the initial and final states is large, but this is only half of the picture. The electron decay rate can be found using Fermi's golden rule, which involves the product of two quantities: the

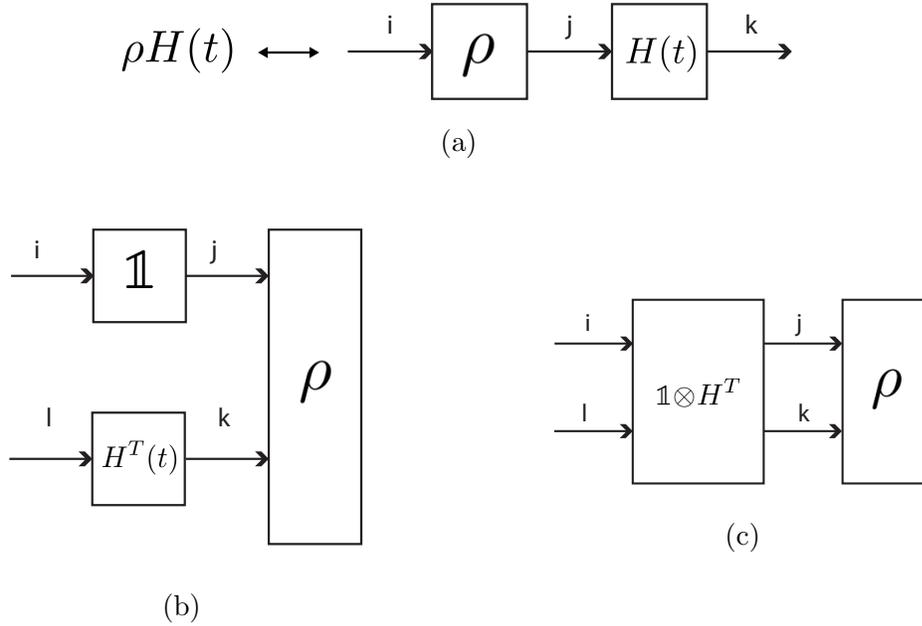


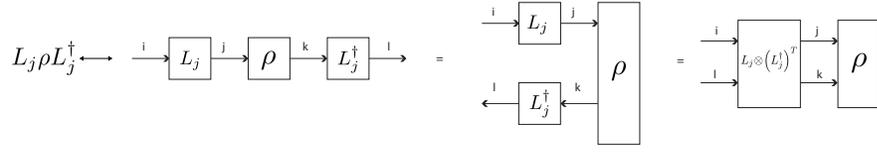
Figure 7.13: Manipulation of the term $\rho H(t)$ using tensor network algebra to obtain the equivalent form $\mathbb{1} \otimes H^T(t)$.

difference in energies of the initial and final states and the overlap of the initial and final states.

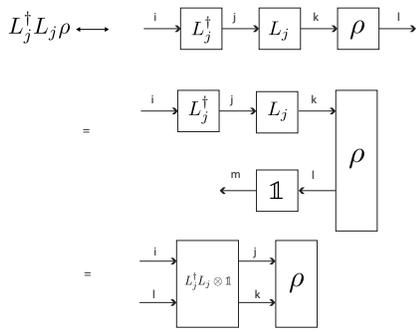
Let us consider three possible electron excitations, and see what Fermi's golden rule says about each decay rate.

An electron in a low orbital that is excited into a nearby orbital has a high decay rate. This is because the energy difference is somewhat large and the overlap between the two orbitals is large. The decay rate of an electron in a highly-excited orbital ($n \approx 100$) into a low orbital is low. Although the energy difference between the final and initial states is large, the overlap between the states is low. The decay rate of a highly-excited electron into an orbital with high energy (say for example, $n_0 = 100$ and $n_1 = 101$), is low. The state overlap is high but the energy difference between the initial and final states is small.

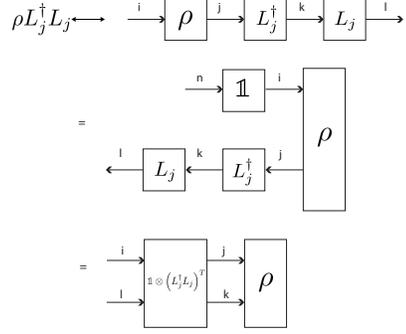
There is no path of decay within the atom that can lead to a large decay rate



$$(a) L_j^\dagger \rho L_j \rightarrow L_j \otimes (L_j^\dagger)^T \rho$$



$$(b) L_j^\dagger L_j \rho \rightarrow L_j^\dagger L_j \otimes \mathbb{1} \rho$$



$$(c) \rho L_j^\dagger L_j \rightarrow \mathbb{1} \otimes (L_j^\dagger L_j)^T \rho$$

Figure 7.14: Tensor network manipulations of the jump operator terms.

for electrons in highly-excited states.

7.9.5 Additional Notes on the Microscopic Model of Interacting Rydberg Polaritons

In this section, we supplement the analysis in Sec. 7.9.1 with plots.

In Sec. 7.9.1, we considered the microscopic model of a Rydberg atom. We modeled the atom as shown in Fig. 7.15 and the microscopic Hamiltonian by Eqn. 7.34. We saw that coupling the atom with light gives three polariton modes – upper bright, dark, and lower bright. The modes are given by Eqn. 7.35 and their energies are given by Eqn. 7.38.

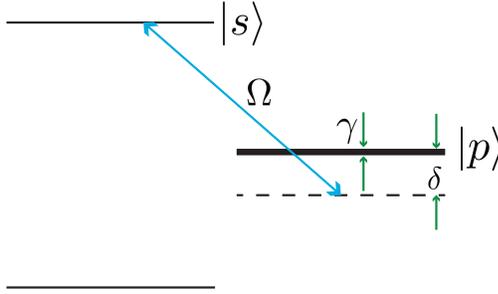


Figure 7.15: Energy level model of the Rydberg atom. The state without a label corresponds to the ground state of the atom. This is the state in which most of the background sea of atoms in the cavity reside. With selection rules, the atom is pumped into the lossy p -state, $|p\rangle$. A control beam, Ω , at detuning δ pumps the atom from the p -state into the highly-excited Rydberg state, $|s\rangle$. Interference between the p and s levels, induced by a cavity photon, allows for an electromagnetically induced transparency configuration, where a polariton with no p -state component exists (i.e. the ‘dark’ polariton).

In Fig. 7.16, we plot the energies of the dark and bright polaritons versus the detuning of the p -state from resonance, as given by Eqn. 7.38. Curves with nonzero width have a p -state component and are called bright polaritons. Bright polaritons decay and fluorescence, so they are, well, bright. The dark polaritons have no p -state component. They do not decay and so they are dark. The energy separation between the bright and dark polaritons increases with coupling strength. At $\delta = 0$ the polaritons are in an electromagnetically induced transparency (EIT) configuration. On EIT resonance and at sufficiently strong couplings, short-lived bright polaritons are far off-resonance from the long-lived dark polariton.

Now we shift our attention to the two-body problem. Let us consider the two-body Hamiltonian given by Eqn. 7.42. We write out this Hamiltonian as a matrix in the basis, $|n_\chi, n_p, n_s\rangle = (\chi^\dagger)^{n_\chi} (\phi_p^\dagger)^{n_p} (\phi_s^\dagger)^{n_s} |\text{vac}, \text{vac}, \text{vac}\rangle$, which specifies the number of photons, p -excitations, and Rydberg-excitations. This gives the two-

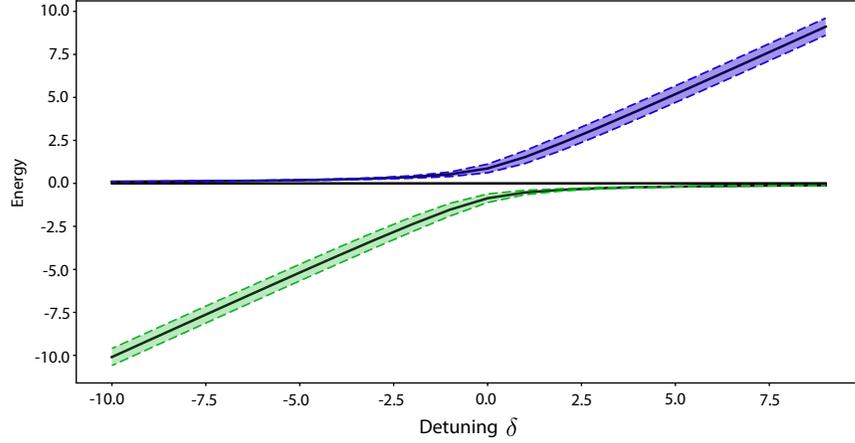


Figure 7.16: Energies of the bright and dark polaritons versus detuning, given by Eqn. 7.35. Widths of the lines are related to the p-state component and thus the decay of the polariton states. The points where $\delta = 0$ correspond to the electromagnetically induced transparency (EIT) configuration, where both bright polaritons are off-resonant from the long-lived dark polariton.

body Hamiltonian,

$$H_{2\text{-body}} = \begin{bmatrix} 0 & 0 & 0 & \sqrt{2}g & 0 & 0 \\ 0 & 2(\delta - i\gamma) & 0 & \sqrt{2}g & 0 & \sqrt{2}\Omega \\ 0 & 0 & 2V & 0 & 0 & \sqrt{2}\Omega \\ \sqrt{2}g & \sqrt{2}g & 0 & \delta - i\gamma & \Omega & 0 \\ 0 & 0 & 0 & \Omega & 0 & g \\ 0 & \sqrt{2}\Omega & \sqrt{2}\Omega & 0 & g & \delta - i\gamma \end{bmatrix} \quad (7.53)$$

In Fig. 7.17, we plot the eigenenergies of the 2-body Hamiltonian from Eqn. 7.53. At high values of the potential, we can see states with a high Rydberg component splitting away from the photon and p-state states. We know these states have a high Rydberg component since the Rydberg state is the only state which varies with a dipole interaction potential, V . A p-state component corresponds to decay, so curves that have width will decay.

Now look at the curves near small values of energy (the red and yellow curves).

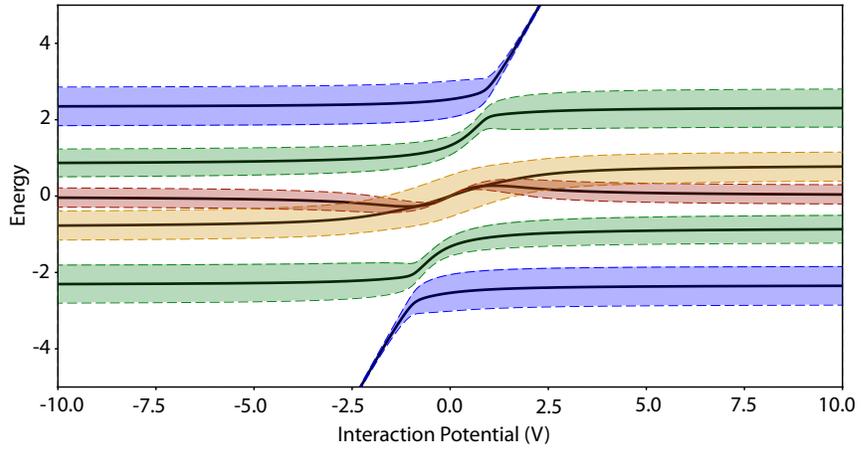


Figure 7.17: Eigenenergies of the two-body Hamiltonian, Eqn. 7.53, as a function of the Rydberg dipole interaction potential.

The yellow curve has a large p -state component, and it represents two bright polaritons. The red curve has a small p -state component and a large s -state component, and it represents two dark polaritons. These two curves are on resonance at some small positive value of the interaction energy. At this resonance, the two dark polaritons can interact and decay into two bright polaritons. This is the two-body polariton loss that occurs from collisions we are considering, i.e. γ_2 .

BIBLIOGRAPHY

- [1] A. M. Meier, B. Eastin, and E. Knill, “Magic-state distillation with the four-qubit code,” Apr 2012.
- [2] S. Bravyi and A. Kitaev, “Universal quantum computation with ideal clifford gates and noisy ancillas,” *Phys. Rev. A*, vol. 71, p. 022316, Feb 2005.
- [3] P. W. Shor, “Fault-tolerant quantum computation,” *Proceedings of the 37th Annual Symposium on Foundations of Computer Science*, p. 56, 1996.
- [4] P. Aliferis, D. Gottesman, and J. Preskill, *Quantum Accuracy Threshold for Concatenated Distance-3 Codes*, vol. 6, pp. 097–165. Rinton Press, 2006.
- [5] R. S. Andrist, J. R. Wootton, and H. G. Katzgraber, “Error thresholds for abelian quantum double models: Increasing the bit-flip stability of topological quantum memory,” *Phys. Rev. A*, vol. 91, p. 042331, Apr 2015.
- [6] S. Girvin, “Circuit qed: superconducting qubits coupled to microwave photons,” vol. 96, July 2011.
- [7] J. Koch, T. M. Yu, J. Gambetta, A. A. Houck, D. I. Schuster, J. Majer, A. Blais, M. H. Devoret, S. M. Girvin, and R. J. Schoelkopf, “Charge-insensitive qubit design derived from the cooper pair box,” *Phys. Rev. A*, vol. 76, p. 042319, Oct 2007.
- [8] A. D. Paolo, T. E. Baker, A. Foley, D. Sénéchal, and A. Blais, “Efficient modeling of superconducting quantum circuits with tensor networks,” *npj Quantum Inf*, vol. 7, January 2021.
- [9] V. E. Manucharyan, J. Koch, L. I. Glazman, and M. H. Devoret, “Fluxonium: Single cooper-pair circuit free of charge offsets,” *Science*, vol. 326, pp. 113–116, October 2009.
- [10] U. Schollwöck, “The density-matrix renormalization group in the age of matrix product states,” *Annals of Physics*, vol. 326, no. 1, pp. 96–192, 2011. January 2011 Special Issue.
- [11] K. R. Colladay, M. Weippert, D. Ferguson, and R. J. Epstein, “Quantization of large superconducting circuits with tensor networks,” *APS March Meeting 2019*, vol. 64, March 2019.

- [12] M. Fishman, S. R. White, and E. M. Stoudenmire, “The ITensor software library for tensor network calculations,” 2020.
- [13] L. W. Clark, N. Schine, C. Baum, J. Ningyuan, and J. Simon, “Observation of laughlin states made of light,” *Nature*, vol. 582, 06 2020.
- [14] P. W. Shor, “Scheme for reducing decoherence in quantum computer memory,” *Phys. Rev. A*, vol. 52, pp. R2493–R2496, Oct 1995.
- [15] Y. Huang and P. Love, “Feynman-path-type simulation using stabilizer projector decomposition of unitaries,” *Phys. Rev. A*, vol. 103, p. 022428, Feb 2021.
- [16] F. Arute, K. Arya, R. Babbush, D. Bacon, J. C. Bardin, R. Barends, R. Biswas, F. G. S. L. Boixo, Sergio Brandao, D. A. Buell, B. Burkett, Y. Chen, Z. Chen, B. Chiaro, R. Collins, W. Courtney, A. Dunsworth, E. Farhi, B. Foxen, A. Fowler, C. Gidney, M. Giustina, R. Graff, K. Guerin, S. Habegger, M. P. Harrigan, M. J. Hartmann, A. Ho, M. Hoffmann, T. Huang, T. S. Humble, S. V. Isakov, E. Jeffrey, Z. Jiang, D. Kafri, K. Kechedzhi, J. Kelly, P. V. Klimov, S. Knysh, A. Korotkov, F. Kostritsa, D. Landhuis, M. Lindmark, E. Lucero, D. Lyakh, S. Mandrà, J. R. McClean, M. McEwen, A. Megrant, X. Mi, K. Michielsen, M. Mohseni, J. Mutus, O. Naaman, M. Neeley, C. Neill, M. Y. Niu, E. Ostby, A. Petukhov, J. C. Platt, C. Quintana, E. G. Rieffel, P. Roushan, N. C. Rubin, D. Sank, K. J. Satzinger, V. Smelyanskiy, K. J. Sung, M. D. Trevithick, A. Vainsencher, B. Villalonga, T. White, Z. J. Yao, P. Yeh, A. Zalcman, H. Neven, and J. M. Martinis, “Quantum supremacy using a programmable superconducting processor,” *Nature*, vol. 574, pp. 505–510, Oct 2019.
- [17] D. P. DiVincenzo, “The physical implementation of quantum computation,” *Fortschritte der Physik*, vol. 48, no. 9-11, pp. 771–783, 2000.
- [18] R. P. Feynman, “Simulating physics with computers,” *International Journal of Theoretical Physics*, vol. 21, pp. 467–488, June 1982.
- [19] R. Raussendorf, D. E. Browne, and H. J. Briegel, “Measurement-based quantum computation on cluster states,” *Phys. Rev. A*, vol. 68, p. 022312, Aug 2003.
- [20] R. Raussendorf, J. Harrington, and K. Goyal, “Topological fault-tolerance in cluster state quantum computation,” *New Journal of Physics*, vol. 9, pp. 199–199, jun 2007.

- [21] P. Walther, K. Resch, T. Rudolph, E. Schenck, H. Weinfurter, V. Vedral, M. Aspelmeyer, and A. Zeilinger, “Experimental one-way quantum computing,” *Nature*, vol. 434, pp. 169–176, March 2005.
- [22] R. Raussendorf and H. J. Briegel, “A one-way quantum computer,” *Physical Review Letters*, vol. 86, pp. 5188–5191, May 2001.
- [23] R. Raussendorf and H. J. Briegel, “A one-way quantum computer,” *Phys. Rev. Lett.*, vol. 86, pp. 5188–5191, May 2001.
- [24] D. Gross, J. Eisert, N. Schuch, and D. Perez-Garcia, “Measurement-based quantum computation beyond the one-way model,” *Physical Review A*, vol. 76, p. 052315, Nov. 2007.
- [25] D. Gottesman, “The heisenberg representation of quantum computers,” *arXv:quant-ph/9807006v1*, 1998.
- [26] P. Brooks, A. Kitaev, and J. Preskill, “Protected gates for superconducting qubits,” *Phys. Rev. A*, vol. 87, p. 052306, May 2013.
- [27] L. B. Nguyen, Y.-H. Lin, A. Somoroff, R. Mencia, N. Grabon, and V. E. Manucharyan, “High-coherence fluxonium qubit,” *Phys. Rev. X*, vol. 9, p. 041041, Nov 2019.
- [28] Q. Ficheux, L. B. Nguyen, A. Somoroff, H. Xiong, K. N. Nesterov, M. G. Vavilov, and V. E. Manucharyan, “Fast logic with slow qubits: Microwave-activated controlled-z gate on low-frequency fluxoniums,” *Phys. Rev. X*, vol. 11, p. 021026, May 2021.
- [29] M. Peruzzo, A. Trioni, F. Hassani, M. Zemlicka, and J. M. Fink, “Surpassing the resistance quantum with a geometric superinductor,” *Phys. Rev. Applied*, vol. 14, p. 044055, Oct 2020.
- [30] T. M. Hazard, A. Gyenis, A. Di Paolo, A. T. Asfaw, S. A. Lyon, A. Blais, and A. A. Houck, “Nanowire superinductance fluxonium qubit,” *Phys. Rev. Lett.*, vol. 122, p. 010504, Jan 2019.
- [31] A. J. Annunziata, D. F. Santavicca, L. Frunzio, G. Catelani, M. J. Rooks, A. Frydman, and D. E. Prober, “Tunable superconducting nanoinductors,” *IOP Nanotechnology*, vol. 21, p. 445202, October 2010.

- [32] A. Kitaev, “Protected qubit based on a superconducting current mirror,” September 2006.
- [33] L. Hormozi, N. E. Bonesteel, and S. H. Simon, “Topological quantum computing with read-rezayi states,” *Phys. Rev. Lett.*, vol. 103, p. 160501, Oct 2009.
- [34] E. Knill, “Fault-tolerant postselected quantum computation: Schemes,” Feb 2004.
- [35] H. Goto, “Step-by-step magic state encoding for efficient fault-tolerant quantum computation - scientific reports,” *Scientific Reports*, vol. 4, p. 7501, December 2014.
- [36] E. T. Campbell, B. M. Terhal, and C. Vuillot, “Roads towards fault-tolerant universal quantum computation,” *Nature*, vol. 549, pp. 172–179, September 2017.
- [37] A. Paler, S. J. Devitt, and A. G. Fowler, “Synthesis of arbitrary quantum circuits to topological assembly,” *Scientific Reports*, vol. 6, p. 30600, August 2016.
- [38] Y. Hwang, B.-S. Choi, Y.-c. Ko, and J. Heo, “Fault-tolerant conversion between stabilizer codes by clifford operations,” *arXiv:1511.02596v1*, 2015.
- [39] J. T. Anderson, G. Duclos-Cianci, and D. Poulin, “Fault-tolerant conversion between the steane and reed-muller quantum codes,” *Physical Review Letters*, vol. 113, p. 080501, Aug. 2014.
- [40] H. Bombin, “Gauge color codes: optimal transversal gates and gauge fixing in topological stabilizer codes,” *New Journal of Physics*, vol. 17, p. 83002, 2015.
- [41] D.-X. Quan, L.-L. Zhu, C.-X. Pei, and B. C. Sanders, “Fault-tolerant conversion between adjacent reed–muller quantum codes based on gauge fixing,” *Journal of Physics A: Mathematical and Theoretical*, vol. 51, p. 115305, 2018.
- [42] B. K. Eastin, “Efficient resource state distillation,” 2015.
- [43] K. R. Colladay and E. J. Mueller, “Rewiring stabilizer codes,” *New J. Phys*, vol. 20, p. 083030, August 2018.

- [44] H. Bombin and M. Martin-Delgado, “Quantum measurements and gates by code deformation,” *J. Phys. A: Math. Theor.*, vol. 42, p. 095302, 2009.
- [45] H. Bombin, “Clifford gates by code deformation,” *New Journal of Physics*, vol. 13, p. 043005, 2011.
- [46] A. M. Stephens, Z. W. E. Evans, S. J. Devitt, and L. C. L. Hollenberg, “Asymmetric quantum error correction via code conversion,” *Physical Review A*, vol. 77, p. 062335, Jun. 2008.
- [47] C. D. Hill, A. G. Fowler, D. S. Wang, and L. C. L. Hollenberg, “Fault-tolerant quantum error correction code conversion,” *arXiv: 1112.2417v1*, 2011.
- [48] E. Nikahd, M. S. Zamani, and M. Sedighi, “A low-overhead hybrid approach for universal fault-tolerant quantum computation,” *arXiv:1610.03309v2*, 2017.
- [49] T. J. Yoder, R. Takagi, and I. L. Chuang, “Universal fault-tolerant gates on concatenated stabilizer codes,” *Physical Review X*, vol. 6, p. 031039, Sep. 2016.
- [50] A. Paetznick and B. W. Reichardt, “Universal fault-tolerant quantum computation with only transversal gates and error correction,” *Physical Review Letters*, vol. 111, p. 090505, Aug. 2013.
- [51] B. Eastin and E. Knill, “Restrictions on transversal encoded quantum gate sets,” *Physical Review Letters*, vol. 102, p. 110502, Mar. 2009.
- [52] D. Gottesman and I. Chuang, “Demonstrating the viability of universal quantum computation using teleportation and single-qubit operations,” *Nature*, vol. 402, pp. 390–393, Nov. 1999.
- [53] E. Knill, R. Laflamme, and G. J. Milburn *Nature*, vol. 409, pp. 46–52, Jan. 2001.
- [54] M. A. Nielsen, “Quantum computation by measurement and quantum memory,” *Physics Letters A*, vol. 308, pp. 96–100, Feb. 2003.
- [55] D. W. Leung, “Quantum computation by measurements,” *International Journal of Quantum Information*, vol. 02, pp. 33–43, Mar. 2004.

- [56] F. Verstraete and J. I. Cirac, “Valence-bond states for quantum computation,” *Physical Review A*, vol. 70, p. 060302(R), Dec. 2004.
- [57] D. Gross and J. Eisert, “Novel schemes for measurement-based quantum computation,” *Physical Review Letters*, vol. 98, p. 220503, May 2007.
- [58] H. J. Briegel, D. E. Browne, W. Dür, R. Raussendorf, and M. V. den Nest, “Measurement-based quantum computation,” *Nature Physics*, vol. 5, pp. 19–26, Jan. 2009.
- [59] P. Aliferis and D. W. Leung, “Computation by measurements: A unifying picture,” *Physical Review A*, vol. 70, p. 062314, Dec. 2004.
- [60] A. Kitaev, “Fault-tolerant quantum computation by anyons,” *Annals of Physics*, vol. 303, no. 1, pp. 2–30, 2003.
- [61] M. A. Nielsen and I. Chuang, *Quantum Computing and Quantum Information*. Cambridge University Press, 10th anniversary ed., 2010.
- [62] C. Huang and M. Newman, “Transversal switching between generic stabilizer codes,” *arXiv: 1709.09282v4*, 2018.
- [63] D. Poulin, “Stabilizer formalism for operator quantum error correction,” *Physical Review Letters*, vol. 95, no. 23, p. 0230504, 2005.
- [64] A. Kubica and M. E. Beverland, “Universal transversal gates with color codes: A simplified approach,” *Physical Review A*, vol. 91, p. 032330, 2015.
- [65] H. Bombin, “Topological subsystem codes,” *Physical review A*, vol. 81, p. 032301, 2010.
- [66] H. Bombin, “Dimensional jump in quantum error correction,” *New Journal of Physics*, vol. 18, p. 043038, 2016.
- [67] C. Knapp, M. Beverland, D. I. Pikulin, and T. Karzig, “Modeling noise and error correction for majorana-based quantum computing,” *arXiv:1806.01275v1*, 2018.
- [68] A. G. Fowler, M. Mariantoni, J. M. Martinis, and A. N. Cleland, “Surface codes: Towards practical large-scale quantum computation,” *Phys. Rev. A*, vol. 86, p. 032324, 2012.

- [69] H. Bombin, “Topological order with a twist: Ising anyons from an abelian model,” *Physical Review Letters*, vol. 105, p. 030403, Jul. 2010.
- [70] “Full length article,” *Chaos, Solitons & Fractals*, vol. 10, no. 10, pp. 1749–1758, 1999.
- [71] H. Anwar, B. J. Brown, E. T. Campbell, and D. E. Browne, “Fast decoders for qudit topological codes,” *New Journal of Physics*, vol. 16, p. 063038, jun 2014.
- [72] E. Dennis, A. Kitaev, A. Landahl, and J. Preskill, “Topological quantum memory,” *Journal of Mathematical Physics*, vol. 43, 2002.
- [73] K. Takeda and H. Nishimori, “Duality of the random model and the quantum toric code,” *Journal of the Physical Society of Japan*, vol. 74, no. Suppl, pp. 115–118, 2005.
- [74] H. Nishimori and K. Nemoto, “Duality and multicritical point of two-dimensional spin glasses,” *Journal of the Physical Society of Japan*, vol. 71, no. 4, pp. 1198–1199, 2002.
- [75] C. H. Bennett, D. P. DiVincenzo, J. A. Smolin, and W. K. Wootters, “Mixed-state entanglement and quantum error correction,” *Phys. Rev. A*, vol. 54, pp. 3824–3851, Nov 1996.
- [76] M. M. Wilde, *Quantum Communication*, p. 642–678. Cambridge University Press, 2 ed., 2017.
- [77] H. Bombin, R. S. Andrist, M. Ohzeki, H. G. Katzgraber, and M. A. Martin-Delgado, “Strong resilience of topological codes to depolarization,” *Phys. Rev. X*, vol. 2, p. 021004, Apr 2012.
- [78] S. Bravyi, M. Suchara, and A. Vargo, “Efficient algorithms for maximum likelihood decoding in the surface code,” *Phys. Rev. A*, vol. 90, p. 032326, Sep 2014.
- [79] P. Sarvepalli and A. Klappenecker, “Degenerate quantum codes and the quantum hamming bound,” *Phys. Rev. A*, vol. 81, p. 032318, Mar 2010.
- [80] A. Ekert and C. Macchiavello, “Error correction in quantum communication,” 1996.

- [81] D. Gottesman, “Class of quantum error-correcting codes saturating the quantum hamming bound,” *Physical Review A*, vol. 54, p. 1862–1868, Sep 1996.
- [82] D. Marcuse, “Light transmission optics,” *New York: Van Nostrand Reinhold*, vol. 2nd ed, p. 115, 1982.
- [83] R. M. Foster, “A reactance theorem,” *Bell Labs Technical Journal*, vol. 3, pp. 259–267, April 1924.
- [84] U. Vool and M. Devoret, “Introduction to quantum electromagnetic circuits,” *International Journal of Circuit Theory and Applications*, vol. 45, 10 2016.
- [85] S. E. Nigg, H. Paik, B. Vlastakis, G. Kirchmair, S. Shankar, L. Frunzio, M. H. Devoret, R. J. Schoelkopf, and S. M. Girvin, “Black-box superconducting circuit quantization,” *Phys. Rev. Lett.*, vol. 108, p. 240502, Jun 2012.
- [86] M. H. Ansari, “Exact quantization of superconducting circuits,” *arXiv e-prints*, p. arXiv:1807.00792, July 2018.
- [87] M. F. Gely and G. A. Steele, “QuCAT: quantum circuit analyzer tool in python,” *New Journal of Physics*, vol. 22, p. 013025, jan 2020.
- [88] D. Ding, H.-S. Ku, Y. Shi, and H.-H. Zhao, “Free-mode removal and mode decoupling for simulating general superconducting quantum circuits,” *Phys. Rev. B*, vol. 103, p. 174501, May 2021.
- [89] M. Lee, M.-S. Choi, and M. Y. Choi, “Quantum phase transitions and persistent currents in josephson-junction ladders,” *Phys. Rev. B*, vol. 68, p. 144506, Oct 2003.
- [90] A. J. Kerman, “Efficient numerical simulation of complex josephson quantum circuits,” 2020.
- [91] D. K. Weiss, A. C. Y. Li, D. G. Ferguson, and J. Koch, “Spectrum and coherence properties of the current-mirror qubit,” *Phys. Rev. B*, vol. 100, p. 224507, Dec 2019.
- [92] V. Ambegaokar and A. Baratoff, “Tunneling between superconductors,” *Phys. Rev. Lett.*, vol. 10, pp. 486–489, Jun 1963.
- [93] Y. Nakamura, Y. Pashkin, and J. Tsai, “Coherent control of macroscopic

- quantum states in a single-cooper-pair box,” *Nature*, vol. 398, pp. 786–788, March 1994.
- [94] J. I.-J. Wang, D. Rodan-Legrain, L. Bretheau, D. L. Campbell, B. Kannan, M. K. David Kim, P. Krantz, G. O. Samach, F. Yan, J. L. Yoder, K. Watanabe, T. Taniguchi, T. P. Orlando, S. Gustavsson, P. Jarillo-Herrero, and W. D. Oliver, “Coherent control of a hybrid superconducting circuit made with graphene-based van der waals heterostructures,” *Nature*, vol. 14, pp. 120–125, December 2018.
- [95] S. J. Weber, “Gatemons get serious,” *Nature Nanotechnology*, vol. 13, pp. 877–878, July 2018.
- [96] L. Casparis, M. R. Connolly, M. Kjaergaard, N. J. Pearson, A. Kringhøj, T. W. Larsen, F. Kuemmeth, T. Wang, C. Thomas, S. Gronin, G. C. Gardner, M. J. Manfra, C. M. Marcus, and K. D. Petersson, “Superconducting gatemon qubit based on a proximitized two-dimensional electron gas,” *Nature Nanotechnology*, vol. 13, pp. 915–919, July 2018.
- [97] S. R. White, “Density matrix renormalization group algorithms with a single center site,” *Phys. Rev. B*, vol. 72, p. 180403, Nov 2005.
- [98] 2021.
- [99] J. C. Bridgeman and C. T. Chubb, “Hand-waving and interpretive dance: and introductory course on tensor networks,” *Journal of Physics A: Mathematical and Theoretical*, vol. 50, p. 223001, May 2017.
- [100] I. Affleck, T. Kennedy, E. H. Lieb, and H. Tasaki, “Rigorous results on valence-bond ground states in antiferromagnets,” *Phys. Rev. Lett.*, vol. 59, pp. 799–802, Aug 1987.
- [101] T.-C. Wei, I. Affleck, and R. Raussendorf, “Two-dimensional affleck-kennedy-lieb-tasaki state on the honeycomb lattice is a universal resource for quantum computation,” *Phys. Rev. A*, vol. 86, p. 032328, Sep 2012.
- [102] T.-C. Wei, P. Haghnegahdar, and R. Raussendorf, “Hybrid valence-bond states for universal quantum computation,” *Phys. Rev. A*, vol. 90, p. 042333, Oct 2014.
- [103] A. Miyake, “Quantum computational capability of a 2d valence bond solid

- phase,” *Annals of Physics*, vol. 326, no. 7, pp. 1656–1671, 2011. July 2011 Special Issue.
- [104] J. Johansson, P. Nation, and F. Nori, “Qutip: An open-source python framework for the dynamics of open quantum systems,” *Computer Physics Communications*, vol. 183, no. 8, pp. 1760–1772, 2012.
- [105] J. Johansson, P. Nation, and F. Nori, “Qutip 2: A python framework for the dynamics of open quantum systems,” *Computer Physics Communications*, vol. 184, no. 4, pp. 1234–1240, 2013.
- [106] E. M. Stoudenmire and S. R. White, “Sliced basis density matrix renormalization group for electronic structure,” *Phys. Rev. Lett.*, vol. 119, p. 046401, Jul 2017.
- [107] D. E. Knuth, *Art of Computer Programming, Volume 2: Seminumerical Algorithms, The, 3rd Edition*. London: Addison-Wesley, 1998.
- [108] B. P. Welford, “Note on a method for calculating corrected sums of squares and products,” *Technometrics*, vol. 4, no. 3, pp. 419–420, 1962.
- [109] C. Hubig, J. Haegeman, and U. Schollwöck, “Error estimates for extrapolations with matrix-product states,” *Phys. Rev. B*, vol. 97, p. 045125, Jan 2018.
- [110] K. R. Colladay and E. J. Mueller, “Driven dissipative preparation of few-body Laughlin states of Rydberg polaritons in twisted cavities,” 2021.
- [111] H. L. Stormer, D. C. Tsui, and A. C. Gossard, “The fractional quantum hall effect,” *Rev. Mod. Phys.*, vol. 71, pp. S298–S305, Mar 1999.
- [112] I. Bloch, J. Dalibard, and W. Zwerger, “Many-body physics with ultracold gases,” *Rev. Mod. Phys.*, vol. 80, p. 885, 07 2008.
- [113] I. Bloch, J. Dalibard, and S. Nascimbène, “Quantum simulations with ultracold quantum gases,” *Nature Physics*, vol. 8, pp. 267–276, 04 2012.
- [114] N. Wilkin and J. Gunn, “Condensation of “composite bosons” in a rotating BEC,” *Physical Review Letters*, vol. 84, pp. 6–9, 01 2000.
- [115] B. Paredes, P. Zoller, and J. Cirac, “Fractional quantum hall regime of a

- gas of ultracold atoms,” *Solid State Communications*, vol. 127, pp. 155–162, July 2003.
- [116] B. Paredes, P. Fedichev, J. I. Cirac, and P. Zoller, “ $\frac{1}{2}$ -anyons in small atomic bose-einstein condensates,” *Phys. Rev. Lett.*, vol. 87, p. 010402, Jun 2001.
- [117] N. Cooper, “Rapidly rotating atomic gases,” *Advances in Physics*, vol. 57, no. 6, pp. 539–616, 2008.
- [118] N. Barberán, M. Lewenstein, K. Osterloh, and D. Dagnino, “Ordered structures in rotating ultracold bose gases,” *Phys. Rev. A*, vol. 73, p. 063623, Jun 2006.
- [119] N. R. Cooper, N. K. Wilkin, and J. M. F. Gunn, “Quantum phases of vortices in rotating bose-einstein condensates,” *Phys. Rev. Lett.*, vol. 87, p. 120405, Aug 2001.
- [120] A. S. Sørensen, E. Demler, and M. D. Lukin, “Fractional quantum hall states of atoms in optical lattices,” *Phys. Rev. Lett.*, vol. 94, p. 086803, Mar 2005.
- [121] M. Hafezi, A. S. Sørensen, E. Demler, and M. D. Lukin, “Fractional quantum hall effect in optical lattices,” *Phys. Rev. A*, vol. 76, p. 023613, Aug 2007.
- [122] J. Zhang, J. Beugnon, and S. Nascimbene, “Creating fractional quantum hall states with atomic clusters using light-assisted insertion of angular momentum,” *Phys. Rev. A*, vol. 94, p. 043610, Oct 2016.
- [123] N. R. Cooper and J. Dalibard, “Reaching fractional quantum hall states with optical flux lattices,” *Phys. Rev. Lett.*, vol. 110, p. 185301, Apr 2013.
- [124] M. Roncaglia, M. Rizzi, and J. Dalibard, “From rotating atomic rings to quantum hall states,” *Sci Rep*, vol. 1, July 2011.
- [125] N. Gemelke, E. Sarajlic, and S. Chu, “Rotating few-body atomic systems in the fractional quantum hall regime,” 2010.
- [126] T. Ozawa, H. M. Price, A. Amo, N. Goldman, M. Hafezi, L. Lu, M. C. Rechtsman, D. Schuster, J. Simon, O. Zilberberg, , and I. Carusotto, “Topological photonics,” *Rev. Mod. Phys.*, vol. 91, p. 015006, 03 2019.
- [127] M. F. Maghrebi, N. Y. Yao, M. Hafezi, T. Pohl, O. Firstenberg, and A. V.

- Gorshkov, “Fractional quantum hall states of rydberg polaritons,” *Physical Review A*, vol. 91, p. 033838, March 2015.
- [128] B. M. Anderson, R. Ma, C. Owens, D. I. Schuster, and J. Simon, “Engineering topological many-body materials in microwave cavity arrays,” *Phys. Rev. X*, vol. 6, p. 041043, Dec 2016.
- [129] M. Hafezi, M. D. Lukin, and J. M. Taylor, “Non-equilibrium fractional quantum hall state of light,” *New Journal of Physics*, vol. 15, p. 063001, jun 2013.
- [130] D. E. Chang, V. Gritsev, G. Morigi, V. Vuletić, M. D. Lukin, and E. A. Demler, “Crystallization of strongly interacting photons in a nonlinear optical fibre,” *Nature Physics*, vol. 4, pp. 884–889, August 2008.
- [131] K. M. Birnbaum, A. Boca, R. Miller, A. D. Boozer, T. E. Northup, and H. J. Kimble, “Photon blockade in an optical cavity with one trapped atom,” *Nature*, vol. 436, pp. 87–90, July 2005.
- [132] Y.-H. Wu, H.-H. Tu, and G. J. Sreejith, “Fractional quantum hall states of bosons on cones,” *Phys. Rev. A*, vol. 96, p. 033622, Sep 2017.
- [133] N. Schine, A. Ryou, A. Gromov, A. Sommer, and J. Simon, “Synthetic landau levels for photons,” *Nature (London)*, vol. 534, pp. 671–675, June 2016.
- [134] D. Arovas, J. R. Schrieffer, and F. Wilczek, “Fractional statistics and the quantum hall effect,” *Phys. Rev. Lett.*, vol. 53, pp. 722–723, Aug 1984.
- [135] A. Stern, “Anyons and the quantum hall effect—a pedagogical review,” *Annals of Physics*, vol. 323, no. 1, pp. 204–249, 2008. January Special Issue 2008.
- [136] F. E. Camino, W. Zhou, and V. J. Goldman, “Realization of a laughlin quasiparticle interferometer: Observation of fractional statistics,” *Phys. Rev. B*, vol. 72, p. 075342, Aug 2005.
- [137] A. Muñoz de las Heras, E. Macaluso, and I. Carusotto, “Anyonic molecules in atomic fractional quantum hall liquids: A quantitative probe of fractional charge and anyonic statistics,” *Phys. Rev. X*, vol. 10, p. 041058, Dec 2020.
- [138] L. S. Georgiev, “Topological quantum computation with non-abelian anyons in fractional quantum hall states,” in *Quantum Systems in Physics, Chemistry, and Biology* (A. Tadjer, R. Pavlov, J. Maruani, E. J. Brändas, and

- G. Delgado-Barrio, eds.), (Cham), pp. 75–94, Springer International Publishing, 2017.
- [139] E. Kapit, P. Ginsparg, and E. Mueller, “Non-abelian braiding of lattice bosons,” *Phys. Rev. Lett.*, vol. 108, p. 066802, Feb 2012.
- [140] E. Kapit, M. Hafezi, and S. H. Simon, “Induced self-stabilization in fractional quantum hall states of light,” *Phys. Rev. X*, vol. 4, p. 031039, Sep 2014.
- [141] S. Dutta and E. J. Mueller, “Coherent generation of photonic fractional quantum hall states in a cavity and the search for anyonic quasiparticles,” *Phys. Rev. A*, vol. 97, p. 033825, March 2018.
- [142] L. Béguin, A. Vernier, R. Chicireanu, T. Lahaye, and A. Browaeys, “Direct measurement of the van der waals interaction between two rydberg atoms,” *Phys. Rev. Lett.*, vol. 110, p. 263201, June 2013.
- [143] R. B. Laughlin, “Anomalous quantum hall effect: An incompressible quantum fluid with fractionally charged excitations,” *Phys. Rev. Lett.*, vol. 50, pp. 1395–1398, May 1983.
- [144] S. Diehl, A. Micheli, A. Kantian, B. Kraus, H. P. Büchler, and P. Zoller, “Quantum states and phases in driven open quantum systems with cold atoms,” *Nature Physics*, vol. 4, no. 11, pp. 878–883, 2008.
- [145] B. Kraus, H. P. Büchler, S. Diehl, A. Kantian, A. Micheli, and P. Zoller, “Preparation of entangled states by quantum markov processes,” *Phys. Rev. A*, vol. 78, p. 042307, Oct 2008.
- [146] F. Verstraete, M. M. Wolf, and I. J. Cirac, “Quantum computation and quantum-state engineering driven by dissipation,” *Nature Physics*, vol. 5, pp. 633–636, Sept. 2009.
- [147] S. Diehl, E. Rico, M. A. Baranov, and P. Zoller, “Topology by dissipation in atomic quantum wires,” *Nature Physics*, vol. 7, no. 12, pp. 971–977, 2011.
- [148] Z.-P. Cian, G. Zhu, S.-K. Chu, A. Seif, W. DeGottardi, L. Jiang, and M. Hafezi, “Photon pair condensation by engineered dissipation,” *Phys. Rev. Lett.*, vol. 123, p. 063602, Aug 2019.
- [149] V. Sharma and E. J. Mueller, “Driven-dissipative control of cold atoms in tilted optical lattices,” 2021.

- [150] J. C. Budich, P. Zoller, and S. Diehl, “Dissipative preparation of chern insulators,” *Phys. Rev. A*, vol. 91, p. 042117, Apr 2015.
- [151] C.-E. Bardyn, M. A. Baranov, C. V. Kraus, E. Rico, A. İmamoglu, P. Zoller, and S. Diehl, “Topology by dissipation,” *New Journal of Physics*, vol. 15, p. 085001, aug 2013.
- [152] F. Reiter, D. Reeb, and A. S. Sørensen, “Scalable dissipative preparation of many-body entanglement,” *Phys. Rev. Lett.*, vol. 117, p. 040501, Jul 2016.
- [153] R. O. Umucal ilar and I. Carusotto, “Fractional quantum hall states of photons in an array of dissipative coupled cavities,” *Phys. Rev. Lett.*, vol. 108, p. 206809, May 2012.
- [154] J. Lebreuilly, M. Wouters, and I. Carusotto, “Towards strongly correlated photons in arrays of dissipative nonlinear cavities under a frequency-dependent incoherent pumping,” *Comptes Rendus Physique*, vol. 17, no. 8, pp. 836–860, 2016. Polariton physics / Physique des polaritons.
- [155] M. Hafezi, P. Adhikari, and J. M. Taylor, “Chemical potential for light by parametric coupling,” *Phys. Rev. B*, vol. 92, p. 174305, Nov 2015.
- [156] V. V. Albert, S. O. Mundhada, A. Grimm, S. Touzard, M. H. Devoret, and L. Jiang, “Pair-cat codes: autonomous error-correction with low-order nonlinearity,” *Quantum Science and Technology*, vol. 4, p. 035007, jun 2019.
- [157] C. D. Freeman, C. M. Herdman, and K. B. Whaley, “Engineering autonomous error correction in stabilizer codes at finite temperature,” *Phys. Rev. A*, vol. 96, p. 012311, Jul 2017.
- [158] J. M. Gertler, B. Baker, J. Li, S. Shirol, J. Koch, and C. Wang, “Protecting a bosonic qubit with autonomous quantum error correction,” *Nature*, vol. 590, pp. 243–248, February 2021.
- [159] J. Combes, “A self-correcting qubit,” *Nature Physics*, February 2021.
- [160] Y. Liu, S. Shankar, N. Ofek, M. Hatridge, A. Narla, K. M. Sliwa, L. Frunzio, R. J. Schoelkopf, and M. H. Devoret, “Comparing and combining measurement-based and driven-dissipative entanglement stabilization,” *Phys. Rev. X*, vol. 6, p. 011022, Mar 2016.
- [161] J. Cohen and M. Mirrahimi, “Dissipation-induced continuous quantum error

- correction for superconducting circuits,” *Phys. Rev. A*, vol. 90, p. 062344, Dec 2014.
- [162] M. Gau, R. Egger, A. Zazunov, and Y. Gefen, “Towards dark space stabilization and manipulation in driven dissipative majorana platforms,” *Phys. Rev. B*, vol. 102, p. 134501, Oct 2020.
- [163] M. Gau, R. Egger, A. Zazunov, and Y. Gefen, “Driven dissipative majorana dark spaces,” *Phys. Rev. Lett.*, vol. 125, p. 147701, Oct 2020.
- [164] A. Sommer and J. Simon, “Engineering photonic floquet hamiltonians through fabry–pérot resonators,” *New Journal of Physics*, vol. 18, p. 035008, mar 2016.
- [165] V. Fock, “Bemerkung zur quantelung des harmonischen oszillators im magnetfeld,” *Zeitschrift für Physik*, vol. 47, no. 5, pp. 446–448, 1928.
- [166] C. G. Darwin, “Free motion in the wave mechanics,” *Proc. Royal Soc. London*, vol. 117, p. 258, 1927.
- [167] N. Regnault and T. Jolicoeur, “Quantum hall fractions in rotating bose-einstein condensates,” *Phys. Rev. Lett.*, vol. 91, p. 030402, Jul 2003.
- [168] A. Georgakopoulos, A. Sommer, and J. Simon, “Theory of interacting cavity rydberg polaritons,” *Quantum Science and Technology*, vol. 4, p. 014005, oct 2018.
- [169] P. Bienias, S. Choi, O. Firstenberg, M. F. Maghrebi, M. Gullans, M. D. Lukin, A. V. Gorshkov, and H. P. Büchler, “Scattering resonances and bound states for strongly interacting rydberg polaritons,” *Phys. Rev. A*, vol. 90, p. 053804, Nov 2014.
- [170] M. J. Gullans, J. D. Thompson, Y. Wang, Q.-Y. Liang, V. Vuletić, M. D. Lukin, and A. V. Gorshkov, “Effective field theory for rydberg polaritons,” *Phys. Rev. Lett.*, vol. 117, p. 113601, Sep 2016.
- [171] A. V. Gorshkov, J. Otterbach, M. Fleischhauer, T. Pohl, and M. D. Lukin, “Photon-photon interactions via rydberg blockade,” *Phys. Rev. Lett.*, vol. 107, p. 133602, Sep 2011.
- [172] M. F. Maghrebi, M. J. Gullans, P. Bienias, S. Choi, I. Martin, O. Firstenberg, M. D. Lukin, H. P. Büchler, and A. V. Gorshkov, “Coulomb bound states

of strongly interacting photons,” *Phys. Rev. Lett.*, vol. 115, p. 123601, Sep 2015.

- [173] S. Weber, C. Tresp, H. Menke, A. Urvoy, O. Firstenberg, H. P. Büchler, and S. Hofferberth, “Calculation of rydberg interaction potentials,” *Journal of Physics B: Atomic, Molecular and Optical Physics*, vol. 50, p. 133001, jun 2017.