

UC Merced

UC Merced Electronic Theses and Dissertations

Title

Detector readout of an Analog Quantum Simulator

Permalink

<https://escholarship.org/uc/item/7t01k059>

Author

Monteros, Alessandro Luis

Publication Date

2021

Peer reviewed|Thesis/dissertation

UNIVERSITY OF CALIFORNIA, MERCED

Detector readout of an Analog Quantum Simulator

A dissertation submitted in partial fulfillment of the requirements
for the degree of Doctor of Philosophy

in

Physics

by

Alessandro Luis Monteros

Committee in charge:

Professor Jay E. Sharping, Chair

Professor Chih-Chun Chien

Professor Lin Tian

2020

©

Alessandro Luis Monteros 2020

All Rights Reserved

The dissertation of Alessandro Luis Monteros, titled Detector readout of an Analog Quantum Simulator, is approved, and it is acceptable in quality and form for publication.

(Professor Lin Tian) Principal Adviser

Date

(Professor Jay E. Sharping) Committee Chair

Date

(Professor Chih-Chun Chien) Committee Member

Date

University of California, Merced
2020

Dedication

First and foremost I dedicate this work to God

I also dedicate this to my family.

To my sister, Ramona, who helped raise me and without whom I may have never pursued higher education

To my eldest brother, Carlos, who helped raise me and without whom I may not have had the tools to complete the PhD

To my elder brother, Gabriel, who was the greatest of friends when we were young and whom keeps me in check today

To my mother and my father, Laura and Gerald, who brought me into this world and started my love of science, science fiction, and all things beautiful, rational, and holy

To all my friends in the PhD program; to Amanda, Som, Nivin, Johnathon, and even Katerina who gave me support, housing, and friendship.

I especially dedicate this work to my beautiful children, Elliott and Emilia, and their amazing mother, Ashlyn. They sacrificed alongside me on this journey. They comforted and loved me deeply. They inspired me and strengthened me. Watching my children smile and love, watching them be overwhelmed with excitement leads me to believe there is good and hope in this world

Ashlyn's patience and kindness, her love and her compassion, were integral to my success on this journey. I can't imagine a better mother to our children.

She is the proof that love and knowledge commute.

To them I owe the world.

Acknowledgments

I would like to acknowledge first and foremost my PhD Advisor, Dr. Lin Tian. She was a guiding force during the PhD. Her support and understanding were absolutely essential in the last leg of this journey. I learn a great deal in my time in this program. I would also like to acknowledge my Committee for their patience and understanding as well as their help and guidance.

I would like to thank my lab mates for their help and for the conversations we had about physics, research, and life. I would like to thank Palak for her support and for her ear. I would like to thank my friends in the PhD program who supported me.

I would like to thank my family who have been with me throughout my life. For my mother and her support in watching children and providing us an affordable place to live. For Robyn and Christopher Reagan who helped watch my children and who provided a home for us.

I thank my Master's adviser, Dr. Ionel Tifrea, for teaching me a great deal about physics and reminding me to focus on the fundamentals.

Finally I would like to thank my Physics professor from my undergraduate school, Dr. Fasel and my computer science professor Dr. Warford. They gave me lots to think about and really helped set a foundation for my academic career.

This dissertation was funded in part by the University of California, Merced Physics Graduate Student Summer Fellowship in 2017-2020.

Alessandro L. Monteros
Curriculum vitae
University of California, Merced
Department of Physics
Phone: (626) 201-2883
Email: amoneros@ucmerced.edu

EDUCATION:

Ph.D. Physics, University of California, Merced, 2020
M.S. Physics, California State University, Fullerton, 2014
B.S. Physics, Pepperdine University, 2011
Minor: Computer Science

WORK EXPERIENCE

- ❑ **Teaching Assistant. University of California, Merced, 2014 - 2020**
 - ❑ Introductory Physics/Math, Labs and Discussion
 - ❑ PHYS 008,018,009,019; MATH 011,021,022
 - ❑ Graduate/Undergraduate Statistical Mechanics
 - ❑ PHYS 112/212
 - ❑ Graduate/Undergraduate Atomic, Molecular, and Optical Physics
 - ❑ PHYS 144/244
- ❑ **Graduate Researcher, University of California, Merced, 2015**
 - ❑ Funded through a Defence Threat Reduction Agency (DTRA) grant for the summer
- ❑ **Teaching Assistant. California State University, Fullerton, 2012 - 2014**
 - ❑ Introductory Physics, Labs
- ❑ **Teaching Assistant. Pepperdine University, 2009-2010**
 - ❑ Introductory Physics, Labs

ADDITIONAL COURSES AND CONTINUOUS LEARNING:

- ❑ Quantum Machine Learning, EdX, 2019
- ❑ Introduction to Complex Analysis, Coursera, 2016
- ❑ Nielson's online textbook *Neural Networks and Deep Learning*, 2019

SKILLS:

Primary Programming: Python, Numpy, Sympy

Programming: FORTRAN, LaTeX, LabView, PyQuil

Basic Programming: Assembly, Prolog, Lisp, C++, Java,

Notable Programming Projects:

- ❑ Wrote a simple compiler in Java for a computer science course
- ❑ Used FORTRAN to implement numerical self consistent calculations
- ❑ Drew and analyzed Feynman diagrams using Python
- ❑ Numerical calculations in Python
- ❑ Created a graphical user interface for data analysis in Python
- ❑ Created various experimental setups in LabView

Experimental:

- ❑ Designed and created an optical set-up to perform photoluminescence measurements
- ❑ Designed and created a set-up to perform MOKE measurements using a photoelastic modulator and a lock-in amplifier
- ❑ Worked with, maintained, and troubleshot various pieces of equipment including but not limited to an optical spectrometer, liquid nitrogen CCD, and an ion-argon laser

RESEARCH

Tian Group, University of California, Merced, 2016-2020

- ❑ Studied the limits of detection in quantum simulators
- ❑ Studied the cavity and qubit interaction in various approximations
- ❑ Found numerical solutions to an eigenvalue problem
- ❑ Found all connected 4th order Feynman graphs using Python
- ❑ Created a GUI in Python with Tinkter to display and examine Feynman graphs
- ❑ Symbolically solved Matubara summations using a program I built around SymPy
- ❑ Presented research and literature reviews to the group

Quantum Matter Group, University of California, Merced, 2014-2016

- ❑ Designed and set up experiments to study single layer thick materials
- ❑ Troubleshot and fixed equipment
- ❑ Collaborated with lab members in lab to generate solutions to our problems
- ❑ Presented weekly updates and literature reviews to the group
- ❑ Created a GUI in Python to assist in data analysis
- ❑ Managed and analyzed spectroscopy data in Python
- ❑ Automated experiments using LabView

Tifrea Group, California State University Fullerton, 2012-2014

- ❑ Calculated the transport properties of a T-shaped quantum dot system
- ❑ Delegated research responsibilities among lab group members
- ❑ Numerically solved self-consistent equations using FORTRAN and Python
- ❑ Published a paper on our findings

PUBLICATIONS:

Papers:

1. A. Monteros et al. Thermoelectric transport properties of a T-shaped double quantum dot system in the Coulomb blockade regime. *Eur. Phys. J. B* (2014) 87: 302
2. Monteros and Tian, Detector backaction on a quantum Ising model, in preparation
3. Monteros and Tian, Detector readout of superconducting qubits in the ultrastrong coupling limit, in preparation

PRESENTATIONS:

Detection of Analog Quantum Simulators

Poster board: SQulnT Conference, Albuquerque New Mexico 2019

Detection of Analog Quantum Simulators

Talk: APS March Meeting, Los Angeles March 2018

Cloud Quantum Computing

Guest lecture for PHYS 172/292 Quantum Information Science, University of California, Merced November 2017

Cloud Quantum Computing

Seminar Talk: University of California, Merced, December 2017

Solar Wind Interaction with the Magnetosphere: PMAF Events and Their Link to Magnetic Reconnection on the Dayside

Poster board: SCCUR, Malibu, CA, November 2010.

Abstract

Detector readout of an Analog Quantum Simulator

by

Alessandro Luis Monteros

Doctor of Philosophy in Physics

University of California, Merced

Professor Jay Sharping, Chair

An important question in quantum simulation is the certification of the quantum simulators with proper readout. We examine how a detector's correlator changes when coupled to a quantum simulator using a diagrammatic technique. From the correlation functions calculated from the diagrammatic technique, we can determine whether or not reliable detection of the simulator's correlator can be achieved. When reliable detection is not possible due to detector back-action, we examine the situations when the back-action can be negligible. In particular, we study a cavity detector coupled to a Transverse Field Ising Model. We use a similar diagrammatic technique to study the interaction between a cavity and a qubit in the ultrastrong coupling regime. This cavity-qubit system is of importance in quantum computing and is a fundamental model in cavity QED. Ultrastrong coupling strength enables novel approaches for quantum logic operations. Our approach provides a fresh perspective on calculating the transmission spectra and the impact of the ultrastrongly coupled cavity on the qubit behavior.

Contents

Acknowledgments	v
Curriculum Vitae	vi
Abstract	ix
1 Introduction	1
1.1 Detection	3
1.2 Outline	4
2 Diagrammatic Perturbation Theory	6
2.1 Fundamentals	6
2.2 Dyson's equation	11
2.3 Cavity-Bath Coupling	12
2.4 Matsubara Summations	14
3 Transverse Field Ising Model	16
3.1 Introduction	16
3.2 Theory of the Transverse Field Ising Model	18
3.2.1 Model	18
3.2.2 Transformations	18
3.3 Perturbative expansion	20
3.4 Approximate detection	25
3.5 Conclusions	38

4	Qubit	40
4.1	Introduction	40
4.2	First approach	41
4.2.1	4-point correlator	46
4.2.2	Higher order correlators	47
4.3	Second Approach	49
4.4	Majorana Approach	52
4.4.1	Second Order	53
4.4.2	Fourth Order	56
4.4.3	Effective Interaction	57
4.4.4	Self Consistent Solution	59
4.5	Conclusions	62
5	Conclusion	64
A	Wicks Theorem for Spin	66
B	TFIM Transforms	69
B.1	Shifting Transformation	75
B.2	Effect of the shift on the bath	77
C	Code	79
C.1	Operator	79
C.2	Diagrams	87
C.3	Matsubara Summations	95
	Bibliography	111

List of Figures

2.1	Poles at $i\omega_n$, z_i , and $i\omega_n - z_i$	15
3.1	Quantum system (blue box) within a cavity. The photon's spectral function is measured in the hope of determining the quantum system's correlator	16
3.2	Schematic of the coupling between the cavity, simulator, and the bath	17
3.3	Transverse Field Ising Model. The blue squares represent a qubit with each line representing a state. The wavy line represents the interaction between qubits and the magnetic field is in the transverse direction. .	18
3.4	spectral density for $N=20$	26
3.5	The spectral function is shown for cases with different classes of diagrams included.	27
3.6	Spectral density for different number of sites. As the number of sites increases the number of lines in the spectrum also increase. However, the splittings caused by the interaction with the cavity does not change. Parameters are $\frac{\omega_c}{J} = 12$, $\frac{\lambda}{J} = 0.04$, $\frac{h_x}{2J} = 0.02$, $\frac{\kappa}{J} = 10^{-4}$, $T = 0.1K$. . .	28
3.7	Using equation 3.16 we can calculate the self energy. This eliminates the splittings in the peaks caused by the cavity allowing for less ambiguous results. The parameters are the same as in figure 3.6 with $N = 20$	28
3.8	The spectral function for varying values of cavity frequency. As the cavity frequency increase the cavity-simulator peaks pull away from the simulator peaks.	30

3.9	The percent difference in the simulator's frequency as a function of cavity frequency	31
3.10	The spectral function for varying values of magnetic fields. As the magnetic field increases the frequency of the simulator spreads out, eventually causing the simulator peaks to overlap with the cavity-simulator peaks.	32
3.11	The percent difference in the simulator's frequency as a function of the applied magnetic field. Two peaks appears, one due to the magnetic field reaching the critical point and the other due to the effective magnetic field reaching the critical point $\frac{h_x}{2J} = 1$	32
3.12	The spectral function for varying number of sites. As the number of sites increases th effective magnetic field decreases eventually turning negative.	33
3.13	Percent difference in the simulator's frequency as a function of the number of sites. As the sites increase the effective magnetic field decreases, eventually flipping direct. This causes the measured and actual frequencies to match perfectly at $N = 1500$. Beyond that point additional sites causes the difference to increase monotonically.	33
3.15	The spectral function for varying cavity bandwidth κ . As the bandwidth increases the cavity-simulator peaks are suppressed.	35
3.16	Despite having a high magnetic field reliable detection can still be achieved because the large bandwidth suppresses the side band peaks. The second order peaks seen in (a) can also be distinguished in the (b). Parameters are $\frac{\kappa}{J} = 5 \times 10^{-1}$, $\frac{h_x}{2J} = 1.5$	36
4.1	These figures reproduce the results found in [1]. The values are $\frac{\omega_c}{2\pi} = 6.336GHz$, $\frac{\lambda}{2\pi} = 4.57GHz$, and $\frac{\omega_z}{2\pi} = 0.505GHz$. The temperature is at 20mK.	44
4.2	Spectra for the cavity qubit system calculated with the boson-boson assumption on a log scale. The parameters are the same as in figure 4.1.	44

4.3	Spectra for the cavity qubit system calculated in the rotating wave approximation on a log scale. The parameters are the same as in figure 4.1.	45
4.4	Log of the spectra for a cavity-qubit system where the cavity is modeled with 3-modes. Parameters are found in [2].	45
4.5	The numerical results for deep strong coupling. The values are $\frac{\omega_c}{2\pi} = 3.142GHz$, $\frac{\lambda}{2\pi} = 0.5GHz$. The temperature is at 20mK.	54
4.6	Log of the spectral function calculated with the second order diagrams, parameters are the same as in figure 4.5.	55
4.7	The four fourth order diagrams each has a counting factor of 1.	56
4.8	Log of the spectral function with the fourth order diagrams, parameters are the same as in figure 4.5.	57
4.9	Log of the spectral function in the effective interaction approximation, the parameters are the same as in figure 4.9.	58
4.10	The log of the spectral function is shown using numerical diagonalization in (a), second order diagrams in (b), fourth order diagrams in (c), and the effective interaction approximation in (d). The parameters are the same as before with the exception of the coupling strength. Here the coupling is 10 times smaller $\frac{\lambda}{2\pi} = 0.05GHz$	59
C.1	Descriptions of an adjacency list, adapted from Introduction to Algorithms [3]. (a) An example of a undirected graph (b) The adjacency list for the graph in (a)	87
C.2	Depth first transversal example. Taken from Introduction to Algorithms [3].	88

Chapter 1

Introduction

In the scientist's quest to understand the world we make use of computer simulations to gain knowledge and insight into complex physical situations. The advancement of available processing power and the development numerical techniques to study physical systems has been integral to the development of modern science. However, despite the steady advancement of computing power many interesting physical systems in our world remain out of computational reach. This is in part due to the fact that in quantum mechanical systems the number of operations needed to characterize the system scales exponentially with its size. While classical simulation methods like Monte Carlo, variational approaches, density-functional theory and more have been used to study quantum systems they all have limits regarding either the size of the system, the strength of the correlations, the dimensionality, or the type of system they apply best to. These limitations put the numerical study of systems like strongly correlated superconductors, frustrated or disordered systems, and large systems out of reach.

To get around this we can use a quantum simulator. Lloyd defines a simulation as follows, "Simulation is a process by which one system is made to mimic another." [4]. This was first proposed by Feynman in [5]. Quantum simulators can be broken down into two types, analog and digital. Quantum computers are a type of digital quantum simulator, sometimes called universal quantum simulators as they can in theory efficiently simulate any local quantum system [4]. It is largely impractical to

build a universal analog quantum simulator [6].

A good quantum simulator should fulfill several key ingredients which are spelled out in [7];

1. Uses a quantum system
2. Can be initialized to some state
3. The Hamiltonian can be engineered
4. Detection, ideally single shot
5. Can be verified in some situations

As a concrete example of this we look at the work done in [8]. Mei *et. al.* proposed an analog simulator for studying the properties of Holstein polarons. The Holstein polaron model is an interesting and relevant model in many-body physics that cannot be solved analytically, only numerically. Mei studied a chain of unit cells made up of transmons coupled to a superconducting resonator driven by a microwave source. They found that under the right conditions the Hamiltonian of this system approximated the Hamiltonian of the Holstein polaron. Additionally by changing the driving parameters all the physically relevant regimes could be reached. Detection was achieved by coupling the system to an ancilla qubit and measuring the stark shift. Thus this setup meets the criteria listed above.

This has been an active area of experimental research in recent times [9–11]. Trapped ions, which typically have good coherence times, have been used to construct such analog simulators. Trapped ions in 2D arrays as large as 11 by 11 with high fidelity and low loss have been observed in trapped ion systems [10]. In that experiment Strontium-88 atoms were arranged in a 2D array using optical trapping and imaged using a single shot atom resolving detection scheme. Additionally a 51 atom linear chain of Rubidium atoms have been created using optical tweezers [11]. In this case distances between the atoms and thus the couplings were tunable. Different symmetry regimes could be reached by pairing atoms together leading to modified Rabi-oscillations. These were detected using fluorescence imaging.

Quantum simulation is also useful for strongly correlated systems which can be resistant to both analytical and numerical methods of evaluation. There are proposals and experiments for systems like the Fermi-Hubbard model [12], strongly coupled spin-boson model [13], and superconductors [14] using trapped ions, driven cavities and qubits, and arrays of quantum dots respectively.

1.1 Detection

Since it is necessary to measure such a quantum simulator it is also necessary to couple the simulator to a detector. By coupling a detector to the simulator the time evolution of the simulator follows the total Hamiltonian of the system. Therefore the presence of the coupling to a detector alters how the simulated system will behave. Put more simply, the act of measurement changes the system. Sometimes this is in non-trivial or negligible ways.

A paper by Clerk studied a cavity coupled to a mechanical mode [15]. The analysis is done with the standard input-output relations [16]. The back-action on the quadrature of interest is pushed to another quadrature by oscillating the coupling at the mechanical resonance frequency. This ensures that the time averaged value of one of the quadratures is zero which limits what can be known about that quadrature and allows back-action free detection of the other quadrature. However, in the presence of noise there can be coupling between the quadratures which may then affect detection.

Schwenk *et. al.* explored the possibility of reconstructing the ideal results of a detector using time order Green's functions (correlators). By examining the difference between the measured quantities and what would be expected given reliable detection a correction term could be added to the correlator. This difference was tied to the applicability of the so-called Wick's theorem to the coupling operators of the simulator [17].

The effect of the back-action on a simulator was also considered by Du [18], in this case a transverse field Ising model was coupled to a cavity. Second order perturbation theory was applied to estimate the effect of the cavity back-action on the system. A shift in the frequency of the eigenmodes was found to increase linearly with system

size.

Transmission measurements can be used to detect the spectral function of the simulator [19] or monitor the dynamics [20]. Blais found that the transmission spectra of a cavity is split when coupled to a qubit. To a second order approximation in the coupling the effect is to pull the cavity frequency by $\pm\lambda^2/\kappa\Delta$ Where λ, κ, Δ are the coupling strength, bandwidth, and the detuning.

The research here most closely follows Tian's previous work in [21] where the conditions under which the back-action can be circumvented are explored using a diagrammatic Green's function (correlator) approach. The setup is the same as ours where a cavity is coupled to a system that acts as a simulator. The systems examined in that work were the Quantum Harmonic Oscillator (QHO) and the electron gas. As in [17] it was found that the validity of Wick's theorem to the coupling operator played a role in reliable detection.

1.2 Outline

The rest of the dissertation is as follows. In chapter 2 we discuss the theory underlying the rest of the dissertation. The entirety of the work is done in the second quantized formalism. The perturbation theory is developed in enough detail to give the reader a solid understanding of the rest of the material. Several key points and issues are pointed out here and will be expanded upon further in subsequent chapters. The perturbation theory discussed here was developed by Feynman, Dyson, Gell-Mann, and others. It has been widely used in condensed matter systems as well as in particle physics. As a way of keeping track of the term in the perturbation series, Feynman diagrams are developed and discussed. A warning is in order at this point. The impulse to attach physical meaning to the Feynman diagrams should be resisted. Rather we should see them as helping to track and give intuition about the many terms involved in a mathematical theory.

In chapter 3 we examine the reliable detection of a quantum simulator. We review the general properties and concepts required for quantum simulations and discuss how the techniques discussed in chapter 2 are applied to our case. The transverse field

Ising model is discussed and examined under this approach looking for if and when reliable detection can be achieved using a cavity mode as a detector. We find that the cavity frequency must be at least four times the maximum frequency of the simulator for a low-bandwidth cavity to avoid interference with the higher order effects from the cavity-simulator coupling. As the bandwidth of the cavity increases these effects are suppressed. Additionally the cavity-simulator coupling leads to an effective magnetic field which must be considered in examining the reliability of the system. The possible generality of these effects are discussed briefly.

Finally in chapter 4 we discuss using the diagrammatic techniques to find the spectrum of a cavity-qubit system. As far as we are aware this is a completely novel approach to this system. Qubits are the building blocks of digital quantum computing and have been well studied over the years [1,2,22–27]. There is an interest in reaching stronger cavity-qubit couplings to examine interesting and novel effects as well to increase the speed of gate operations in quantum computing [28]. Several approaches are used which can explain the primary spectral lines seen in experiments. However, to correctly apply the perturbation theory we must undergo a transformation from spin operators to Majorana operators. This leads to a more complicated coupling scheme. This is investigated using different types of infinite order approximations which only partially explain the spectra. It is likely necessary to use a self-consistent solution to achieve results that are in line with experiments. We discuss this possibility but a detailed description is beyond the scope of this work.

Chapter 2

Diagrammatic Perturbation Theory

2.1 Fundamentals

The fundamental aspects of this work are based on the Feynman diagram technique and the perturbation series that comes from it. We pick out some important aspects of the theory which will be relevant in further discussion. Comprehensive explanations for the theory can be found in [29–31] and many more. Our starting point in describing the theory is to consider a Hamiltonian which is broken into two parts, an unperturbed or "free theory" part and an interaction part $\hat{H} = \hat{H}_0 + \hat{H}_I$. We also consider the unitary operator $\hat{U}(t, t_0)$ whose effect on a state is to translate a state vector in time. We start in the interaction picture as is standard which gives a differential equation (eqn (2.1)) that describes the evolution operator and whose effect on a state vector is $|\psi(t)\rangle_I = U(t, t_0)|\psi(t_0)\rangle_I$ where the states are defined in the interaction picture. This operator can also be written as $\hat{U}(t, t_0) = e^{i\hat{H}_0 t/\hbar} e^{-i\hat{H}(t-t_0)/\hbar} e^{-i\hat{H}_0 t_0/\hbar}$.

$$i\hbar \frac{\delta}{\delta t} \hat{U}(t, t_0) = \hat{H}_I(t) \hat{U}(t, t_0) \quad (2.1)$$

This differential equation can be solved by iteration. Unfortunately there is no guarantee that the solution will converge. Upon iterating several times we are left

with equation (2.2).

$$\hat{U}(t, t_0) = 1 + \left(\frac{-i}{\hbar}\right) \int_{t_0}^t dt' \hat{H}_I(t') + \left(\frac{-i}{\hbar}\right)^2 \int_{t_0}^t dt' \int_{t_0}^{t'} dt'' \hat{H}_I(t') \hat{H}_I(t'') + \dots \quad (2.2)$$

We focus on the third term on the right hand side. We split the integral in to two parts.

$$\int_{t_0}^t dt' \int_{t_0}^{t'} dt'' \hat{H}_I(t') \hat{H}_I(t'') = \frac{1}{2} \int_{t_0}^t dt' \int_{t_0}^{t'} dt'' \hat{H}_I(t') \hat{H}_I(t'') + \frac{1}{2} \int_{t_0}^t dt' \int_{t_0}^{t'} dt'' \hat{H}_I(t'') \hat{H}_I(t') \quad (2.3)$$

Then we change the order of integration and rename the dummy variables

$$\int_{t_0}^t dt' \int_{t_0}^{t'} dt'' \hat{H}_I(t') \hat{H}_I(t'') = \int_{t_0}^t dt'' \int_{t_0}^{t''} dt' \hat{H}_I(t') \hat{H}_I(t'') = \int_{t_0}^t dt' \int_{t'}^t dt'' \hat{H}_I(t'') \hat{H}_I(t') \quad (2.4)$$

We then put this results into the previous equation which gives us the result we see in equation (2.5).

$$\frac{1}{2} \int_{t_0}^t dt' \int_{t_0}^{t'} dt'' \hat{H}_I(t') \hat{H}_I(t'') + \frac{1}{2} \int_{t_0}^t dt' \int_{t'}^t dt'' \hat{H}_I(t'') \hat{H}_I(t') \quad (2.5)$$

This result can be simplified using a time ordering operator with the definition of time order being $T[A(t_f)B(t_i)] = \Theta(t_f - t_i)A(t_f)B(t_i) + \Theta(t_i - t_f)B(t_i)A(t_f)$. Here A and B are arbitrary operators. This matches the time ordering that will be defined for bosonic operators. In most situations where this formalism is applied the interaction part of the Hamiltonian is boson-like. However, we will encounter a case where the interaction involves anticommuting variables. This leads to an inconsistency in the definition of the correlator. This will be addressed in Chapter 4.

$$\frac{1}{2} \int_{t_0}^t dt' \int_{t_0}^{t'} dt'' T [\hat{H}_I(t') \hat{H}_I(t'')] \quad (2.6)$$

We can repeat this process to all orders and use the time ordered exponential notation to simplify the result.

$$U(t, t_0) = T e^{-i \int_{t_0}^t dt' H_I(t')} \quad (2.7)$$

From this point the theory develops by considering the interaction to adiabatically "turn on". This allows us to consider the state of the interacting system as having evolved slowly from the state of the non-interacting system. The Gell-Mann and Low theorem shows that such a state will be an eigenstate of the full Hamiltonian as long as there is some overlap between the interacting and non-interacting systems. This assumption is violated in certain systems such as when there is a phase change [32]. For our purposes we will consider the theorem to be valid.

We move now to discuss the central mathematical object in this work, the 2-point Green's function. This is sometimes referred to as the propagator, the kernel, a correlator, or the response function depending on the exact context and use. Specifically we are examining the time ordered 2-point Green's function which, to avoid confusion, we will refer to as correlators. These correlators contains much of the interesting information about the system such as the energy spectrum and all the single particle operator averages [29–31]. Extraction of the many body correlator is thus desirable. The imaginary part of the retarded correlator can be measured via the spectral function [19]. From this the real part of the correlator can be calculated using the Kramers-Kronig relation [29]. The spectral function, which is minus twice the imaginary part of the correlator, can be thought of as a generalized density of states. Hence it is useful for examining the system and will be a primary focus of much of this work.

We turn to the mathematical definition of these correlators. For the photon operators we have the real time correlator:

$$-i \left\langle T \left(\hat{A}(t_f) \hat{A}(t_i) \right) \right\rangle = -i \frac{1}{\mathcal{Z}} \text{Tr} \left(\hat{\rho} T \left(\hat{A}(t_f) \hat{A}(t_i) \right) \right) \quad (2.8)$$

The operator $\hat{A}(t)$ is $a(t) + a^\dagger(t)$ where $a^\dagger(a)$ are the creation (annihilation) operators. The density matrix $\rho = e^{-\beta \hat{H}}$ (β is the inverse of $k_b T$ and \hat{H} is the Hamiltonian) and the partition function \mathcal{Z} serves to normalize the correlator. T is the time ordering operator which places operators with greater time to the left. For fermions each

exchange introduces a minus sign. The retarded correlator is similar:

$$-i\Theta(t_f - t_i) \left\langle \left[\hat{A}(t_f), \hat{A}(t_i) \right] \right\rangle \quad (2.9)$$

Where $\theta(t)$ is the Heaviside step function which is 1 for positive t and 0 for negative t . The average is now over the commutator of the operators for bosons. For fermion operators we would use the anticommutator. It turns out it is much more convenient to deal with equilibrium finite temperature systems with what's called a Matsubara correlator.

$$\mathcal{D}(\tau_f, \tau_i) = - \left\langle T_\tau \left(\hat{A}(\tau_f) \hat{A}(\tau_i) \right) \right\rangle \quad (2.10)$$

What this does is it takes time to be imaginary $\tau = it$ which allows the same diagrammatic expansion used in the zero temperature formalism to be used in the finite temperature formalism. Additionally the Fourier transform of the correlator maps to an imaginary discrete frequency $i\omega_n$ instead of the normal real continuous frequencies. The Matsubara correlator connects to the retarded correlator in a relatively simple fashion. It can be shown (see [29]) that the retarded correlator can be obtained by taking $i\omega_n \rightarrow \omega + i\delta$ where δ is some small positive quantity. Thus if one obtains the Matsubara correlator then the retarded correlator and the experimentally accessible spectral function are obtained as well.

Another effect of taking time to be imaginary is that the Fourier Transform needs to also be defined. Equation 2.11 provides a definition of this. The frequencies in the Fourier transform are discrete frequencies which are even (odd) for bosons (fermions). They take the form $\frac{\pi 2n}{\hbar\beta}$ or $\frac{\pi(2n+1)}{\hbar\beta}$ where n is an integer. Other than that, the mechanics and properties of the Fourier transforms are essentially the same.

$$\mathcal{G}(i\omega_n) = \int_0^{\hbar\beta} \mathcal{G}(\tau) e^{i\omega_n \tau} d\tau \quad (2.11)$$

$$\mathcal{G}(\tau) = \frac{1}{\hbar\beta} \sum_{i\omega_n} e^{-i\omega_n \tau} \mathcal{G}(i\omega_n) \quad (2.12)$$

As previously done we start with a system with a Hamiltonian that can be written as $H = H_0 + H_I$ where H_0 is a Hamiltonian with known eigenstates and H_I is a perturbation. If we move into the interaction picture the unitary time evolution operator becomes:

$$\hat{U}(\tau, \tau_0) = \sum_{n=0}^{\infty} \left(\frac{-1}{\hbar} \right)^n \frac{1}{n!} \int_{\tau_0}^{\tau} d\tau_1 \dots d\tau_n T(H_I(\tau_1) \dots H_I(\tau_n)) \quad (2.13)$$

Where equation 2.13 is the same as equation 2.7 only expanded. Now the reason why we use an imaginary time can be made clear. In the Matsubara framework the unitary time evolution operators is $\hat{U}(\tau, \tau_0) = e^{\hat{H}_0\tau/\hbar} e^{-\hat{H}(\tau-\tau_0)/\hbar} e^{-\hat{H}_0\tau_0/\hbar}$. This implies that we can rewrite the density operator so that $\rho = e^{-\beta(H_0+H_I)} = e^{-\beta H_0} U(\hbar\beta, 0) = \rho_0 U(\hbar\beta, 0)$. In this case $\hbar\beta$ acts as an imaginary time and can be treated accordingly. Thus by using the imaginary time formalism the density matrix $e^{-\beta H}$ can be seen as a type of unitary time evolution with imaginary time $\hbar\beta$. With the added assumption that the perturbation was “turned on” adiabatically sometime infinitely far back and “turned off” infinitely far forward in time then we can rewrite the Matsubara correlator as:

$$\mathcal{D}(\tau_f, \tau_i) = - \sum_{n=0}^{\infty} \left(\frac{-1}{\hbar} \right)^n \frac{1}{n!} \int_0^{\hbar\beta} d\tau_1 \dots d\tau_n \left\langle T_{\tau} \left(H_I(\tau_1) \dots H_I(\tau_n) \hat{A}(\tau_f) \hat{A}(\tau_i) \right) \right\rangle_0 \quad (2.14)$$

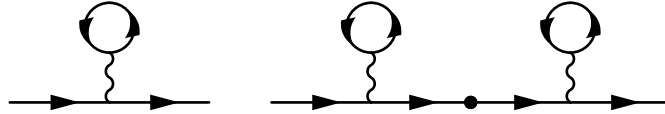
To analyze this to some order n we must find the value of an $n+2$ -point correlator. Wick’s theorem offers an easy way to do so [29, 30, 33, 34]. Wick’s theorem can be seen as a consequence of the Gaussian nature of Hamiltonian in a path integral formulation [35] or as a consequence of the algebraic structure of the operators (see Appendix A). Either way Wick’s theorem states that given a quadratic unperturbed Hamiltonian, correlators of linear operators can be expanded into all the permutations of 2-point correlators. For example a 4-point correlator would be reduced to 2-point correlators as follows:

$$\begin{aligned} \langle T \left(\hat{A}_4 \hat{A}_3 \hat{A}_2 \hat{A}_1 \right) \rangle_0 &= \langle T_\tau \left(\hat{A}_4 \hat{A}_3 \right) \rangle_0 \langle T_\tau \left(\hat{A}_2 \hat{A}_1 \right) \rangle_0 + \langle T \left(\hat{A}_4 \hat{A}_2 \right) \rangle_0 \langle T_\tau \left(\hat{A}_3 \hat{A}_1 \right) \rangle_0 \\ &\quad + \langle T_\tau \left(\hat{A}_4 \hat{A}_1 \right) \rangle_0 \langle T_\tau \left(\hat{A}_3 \hat{A}_2 \right) \rangle_0 \end{aligned} \tag{2.15}$$

Each of these terms can be visualized as a Feynman diagram and can be roughly described as the propagation of some particle between times. For example the first term on the right describes the propagation of a photon from τ_1 to τ_2 and τ_3 to τ_4 . This would be represented by drawing a line between the points 1,2 and 3,4.

2.2 Dyson's equation

In the process of expanding out different n-point correlators one will find that lower order diagrams appear in higher orders. For example in electron-electron interactions the bubble diagram repeats itself. The lowest order and next lowest are seen below. All diagrams were created with the FeynMP latex extension [36]:



(2.16)

The straight lines represent the free fermion correlator \mathcal{G}^0 and the squiggly line represents the electron-electron interaction. The first diagram is considered proper while the second is improper because it is composed of proper diagrams (at the dot) [29]. We must remember that the diagrams represent terms in an equation. The repeating of a diagram is a multiplication so that an expansion of the full fermion correlator \mathcal{G} with those diagrams would be $\mathcal{G}(\omega) = \mathcal{G}^0(\omega)(1 + R + R^2 + \dots)\mathcal{G}^0(\omega)$ where R represents the contribution from the bubble part of this particular diagram. This forms a geometric series which can be summed to infinite order. Thus we are able to have infinite order perturbation series based on the class or type of diagrams we

choose to include. By summing the proper diagrams to infinite order we find the Dyson equation:

$$\mathcal{G}(\omega) = \frac{\mathcal{G}_0(\omega)}{1 - \mathcal{G}_0(\omega)\Sigma(\omega)} \quad (2.17)$$

Here Σ is the sum of whichever proper diagrams you choose to include. For example if you only include only the lowest order proper diagrams this gives you the a forward scattering type approximation [29].

2.3 Cavity-Bath Coupling

To show the technique explicitly we follow the paper from Tian which overviews how this can be used to look at the coupling between the cavity and a bath [21]. The Hamiltonian is $\hat{H} = \hat{H}_c + \hat{H}_B + \hat{H}_{c-B}$ where $\hat{H}_c = \hbar\omega_c a^\dagger a$, $\hat{H}_B = \hbar \sum_i b_i^\dagger b_i$ are the unperturbed cavity and bath Hamiltonian, and $\hat{H}_{c-B} = A\hbar \sum_i c_i (b_i + b_i^\dagger)$ is the cavity-bath interaction. We consider the coupling between the bath and cavity to be a perturbation. In the interaction picture the correlator takes the form:

$$\mathcal{D}_R(\tau_f, \tau_i) = - \sum_{n=0}^{\infty} \left(-\frac{1}{\hbar}\right)^n \frac{1}{n!} \int_0^{\beta\hbar} d\tau_1 \cdots d\tau_n Tr \left[\hat{\rho}_0 \hat{T}_\tau \left[\hat{H}_{c-B}(\tau_1) \cdots \hat{H}_{c-B}(\tau_n) \hat{A}(\tau_f) \hat{A}(\tau_i) \right] \right] \quad (2.18)$$

Where $\rho_0 = e^{-\beta\hat{H}_0}$. Each term in the perturbative expansion can be represented diagrammatically, keeping only the diagrams that connect to the end points. Once all the calculations are done then the diagrams can be converted back into equations.

$$\overline{\overline{\text{---}}} = \overline{\text{---}} \rightarrow \overline{\text{---}} \text{---} \overline{\text{---}} \rightarrow \overline{\text{---}} \quad (2.19)$$

The above diagram is the only 2nd order diagram. There are $n!$ ways of drawing such a diagram which exactly cancels with the $\frac{1}{n!}$ term in the expansion. Thus we say the diagram has a counting factor of 1. We can examine the 4th order process and find a similar diagram. This also has a counting factor of 1. Higher ordered processes

are built up the same way.

$$\overline{\overline{\mathcal{D}_{RB}^4(\omega_n)}} = \overline{\overline{\mathcal{D}_{RB0}(\omega_n)}} \xrightarrow{|c_i|^2 \mathcal{D}_i(\omega_n)} \overline{\overline{\mathcal{D}_{RB0}(\omega_n)}} \xrightarrow{|c_i|^2 \mathcal{D}_i(\omega_n)} \overline{\overline{\mathcal{D}_{RB0}(\omega_n)}} \quad (2.20)$$

Note the simple recursive structure inherent in the diagrams. This is due to there being only a single proper diagram. This gives a simple recursive structure that is reminiscent of a power series. We can thus sum to infinite order using the Dyson equation to obtain the full cavity-bath photon correlator.

$$\overline{\overline{\mathcal{D}_{RB}(\omega_n)}} = \overline{\overline{\mathcal{D}_{RB0}(\omega_n)}} + \overline{\overline{\mathcal{D}_{RB0}(\omega_n)}} |c_i|^2 \mathcal{D}_i(\omega_n) \overline{\overline{\mathcal{D}_{RB}(\omega_n)}} \quad (2.21)$$

This new correlator will be used to in examining the cavity-simulator photon correlator. This allows us to include the cavity decay from the bath.

In frequency space the unperturbed photon correlator, $\mathcal{D}_{RB0}(\omega_n)$, has the well known solution [30]:

$$\mathcal{D}_{RB0}(\omega_n) = \frac{2\omega_c}{(i\omega_n)^2 - \omega_c^2} \quad (2.22)$$

Similarly we can define a correlator for the bath similar to the photon correlator. It has the solution:

$$\mathcal{D}_i(\omega_n) = \frac{2\omega_i}{(i\omega_n)^2 - \omega_i^2} \quad (2.23)$$

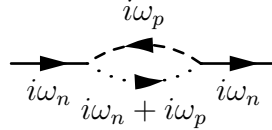
Tian in [21] found that by approximating the noise spectrum as flat (κ) the detector-bath correlator becomes:

$$\mathcal{D}_{RB} = \frac{2\omega_c}{(i\omega_n)^2 - \omega_c^2 + 2i\omega_c\kappa} \quad (2.24)$$

It is important to note that this is only valid for the retarded correlator. If one takes the advanced correlator then the sign in front of kappa must change.

2.4 Matsubara Summations

For more complicated interactions there can be diagrams with branches. The detailed theory can be worked out from the above work but a visual demonstration is more enlightening. In the diagram below we see the correlators split from one line into two. The frequency flows from the left line into the branch where it is split. Some of the frequency flows through the top branch and some flows through the bottom. The way we determine how much flows in each direction is by summing over all the possible values of $i\omega_p$ and requiring that frequency is conserved.



(2.25)

The result of these branches is to give equations like equation (2.26) where the infinite sum needs to be solved. The exact form of the correlators is unimportant at the moment. We simply need two assumptions. The first is that the correlators $\mathcal{G}(z)$ has poles z_i and residues b_i . The second assumption is that the poles do not lay on the imaginary axis.

$$\sum_{i\omega_p} \mathcal{G}(i\omega_p)\mathcal{G}(i\omega_n + i\omega_p) = -\frac{\hbar\beta}{2\pi i} \oint dz n_F(\hbar z)\mathcal{G}(z)\mathcal{G}(i\omega_n + z) \quad (2.26)$$

The trick here is to realize that the fermion (boson) frequencies are the same as the poles of the Fermi-Dirac (Bose-Einstein) distribution. Thus we can rewrite the sum as a contour integral where the contour encloses the poles of the distribution. We can then deform the contour to wrap around the poles of the correlators. We can then let the contour grow in size so that contribution from the curves on the outside go to zero (figure 2.1). If the correlator in question has a $\frac{1}{\omega}$ dependence then care needs to be taken to ensure proper convergence properties. In our cases all of our summations have summands with a $\frac{1}{\omega^2}$ dependence or greater.

We can now use residue theory to calculate the value of the integral. If we assume,

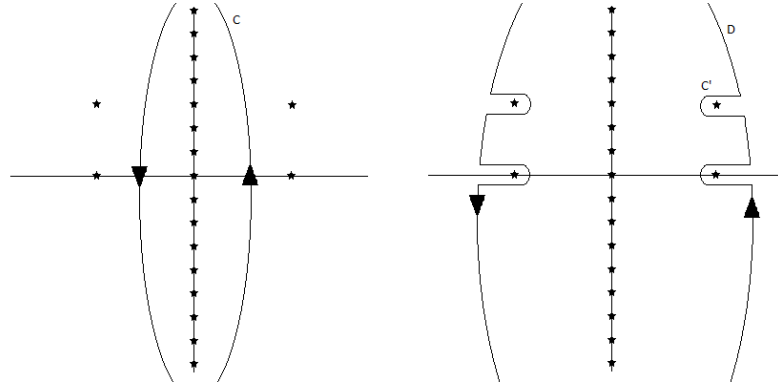


Figure 2.1: Poles at $i\omega_n$, z_i , and $i\omega_n - z_i$

for simplicity, that \mathcal{G} has a single pole at z_0 with unit residue then we can solve equation 2.26.

$$\hbar\beta (n_F(\hbar z_0)\mathcal{G}(i\omega_n - z_0) + n_F(\hbar(z_0 - i\omega_n))\mathcal{G}(z_0 - i\omega_n)) \quad (2.27)$$

The second term can be simplified by realizing that $n_F(x - i\omega_n) = -n_B(x)$ due to the discrete nature of the Matsubara frequencies. This gives the theoretical tools that will be used throughout this work.

Chapter 3

Transverse Field Ising Model

3.1 Introduction

As we build quantum simulators we are confronted with several questions regarding the construction of such systems. Quantum simulators need to be useful, controlled, and capable of reliable detection.

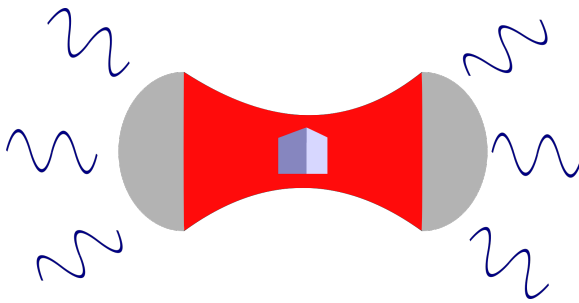


Figure 3.1: Quantum system (blue box) within a cavity. The photon's spectral function is measured in the hope of determining the quantum system's correlator

The detection aspect as pointed out in [7] is the subject of our current exploration. We consider a cavity coupled to a system of interest which we call the simulator (Figure 3.1). When a detector is coupled to the simulator there will be a finite back-action on the simulator. This will affect the many-body correlator and the evolution of the system in a non-trivial manner. This leads to an important question. Can we reliably readout information from a quantum simulator?

We follow our previous work done in [21] by examining the many-body correlators. Typically one ignores such a back-action though there have been attempts to circumvent this either through squeezing [15] or

reconstruction [17].

We examine a system of interest that will be our quantum simulator with a Hamiltonian H_s . We are interested in obtaining the many-body correlator associated with the system. This gives us many of the important properties of the system [29]. In order to readout the relevant information from the simulator we couple it to a cavity through which measurements can be made, i.e. the transmission spectra [19].

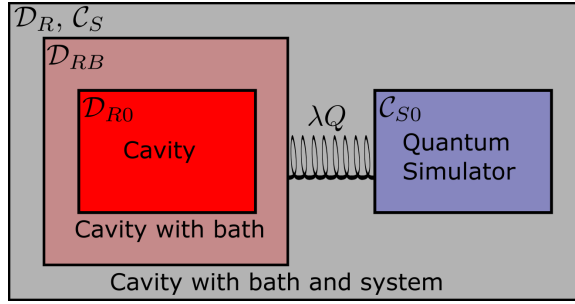


Figure 3.2: Schematic of the coupling between the cavity, simulator, and the bath

However, by adding the cavity the quantum simulator’s correlator is perturbed. A naive measurement will measure this altered correlator. The question we seek to answer is whether or not the quantum simulator’s unperturbed correlator can be extracted. Additionally the cavity is coupled to a bath which has the effect of broadening the cavity resonance. This can be seen schematically in figure 3.2.

Table 3.1: Correlators paired with their respective Hamiltonians

$H_c + H_b + H_s$	\mathcal{D}_{RB0}
$H_c + H_b + H_s + H_{b-c}$	\mathcal{D}_{RB}
$H_c + H_b + H_{b-c} + H_s + H_I$	\mathcal{D}_R

We define the correlators in table 3.1 for different Hamiltonians. The correlator \mathcal{D} maintains the form $-\langle T_\tau [\hat{A}(\tau)\hat{A}(\tau')] \rangle$. H_c , H_b , and H_s refer to the cavity, bath, and simulator’s Hamiltonian, H_{b-c} and H_I refer to the bath-cavity coupling and the cavity-simulator interaction.

3.2 Theory of the Transverse Field Ising Model

3.2.1 Model

The Transverse Field Ising Model (TFIM) or Quantum Ising Model is well studied model in physics that has implications in quantum simulation as well as combinatorics and machine learning [11, 37–41]. In its more general form it is a lattice of spins that couple to the nearest neighbors via spin-coupling. Additionally a magnetic field is applied in the transverse direction which differentiates the quantum case from the classical case. This is shown schematically in figure 3.3.

In our case we are looking at the one-dimensional case. This is exactly solvable through transformations that will be discussed in the next section. This allows us to compare the values that would be calculated in a transmission type experiment with the known values. A discussion of how our results generalize to the more complicated cases is discussed at the end of this chapter.



Figure 3.3: Transverse Field Ising Model. The blue squares represent a qubit with each line representing a state. The wavy line represents the interaction between qubits and the magnetic field is in the transverse direction.

3.2.2 Transformations

In its current form the TFIM is unwieldy to work with. Therefore we will apply a number of transformations to bring the Hamiltonian to a more desirable form. At the moment our focus will be on the unperturbed TFIM (equation 3.1). An outline of the transformations will be given here, for more details see Appendix B,

$$-\hbar J \sum_n^N \sigma_n^z \sigma_{n+1}^z - \hbar \frac{h_x}{2} \sum_n^N \sigma_n^x \quad (3.1)$$

The first difficulty we encounter is in the ambiguous commutation laws for the raising and lowering spin operators. Spins operators corresponding to a single site

follow an anti-commutation law $\{\sigma^-, \sigma^+\} = 1$ but spin operators on differing sites follow a commutation law $[\sigma^-, \sigma^+] = 0$. One way to deal with this is to include correction to the sign of the correlation function that depends on the structure of the Green's function [34]. We take a different approach and apply the well known Jordan-Wigner transformation.

The Jordan-Wigner transformation provides a way of mapping spins to spinless fermions by introducing a non-localized phase factor.

$$\sigma_n^- \rightarrow c_n e^{-i\pi \sum_j^{n-1} c_j^\dagger c_j} \quad (3.2)$$

The phase factor in eqn. 3.2 will give either a +1 or a -1 depending on the number of spin up states preceding the spin. This is what serves to fix the commutation relations so that $\{c_n, c_m^\dagger\} = \delta_{n,m}$ where n and m refer to the sites n and m respectively. With the proper commutation relations in we can transform our Hamiltonian (eqn 3.3). In doing so we drop the periodic term which is negligible for large N.

$$H = -\hbar J \sum_n^N (c_n^\dagger c_{n+1} + c_n^\dagger c_{n+1}^\dagger + h.c.) - \hbar \frac{h_x}{2} \sum_n^N (1 - 2c_n^\dagger c_n) \quad (3.3)$$

In it's current form the Hamiltonian is still unwieldy, additionally there are terms like $c_n^\dagger c_{n+1}^\dagger$ which do not conserve particle number. To remedy this we apply a Fourier transform and a Bogolubov transformation [42]. The Fourier transform simplifies the indices because the wave number is conserved. The Bogolubov transformation is seen in eqn. 3.4 ensures that particle number is conserved.

$$c_k = u_k \gamma_k + i v_k \gamma_{-k}^\dagger \quad (3.4)$$

The operator γ_k is the Bogolubov quasi-particle which has frequency :

$$\omega_k = 2J \sqrt{1 + \left(\frac{h_x}{2J}\right)^2 - \frac{h_x}{J} \cos(k)} \quad (3.5)$$

The coefficients in front of the gamma ensure that the proper commutation relations are followed $u_k = \cos(\theta_k); v_k = \sin(\theta_k)$. After these transformations are complete we

are left with the Hamiltonian seen in equation 3.6. A more detailed account of all these transformations is given in Appendix B.

$$H_{TFIM,0} = \sum_k \hbar \omega_k \gamma_k^\dagger \gamma_k \quad (3.6)$$

3.3 Perturbative expansion

With the unperturbed Hamiltonian transformed into a form that is easily worked with we can examine the effect the detector has. To do this we first transform the detector-simulator coupling in the same way. The coupling is assumed to be in the form $A \sum_i \sigma_i^x$ [18]. The results of these transformations leads to eqn. 3.7. The c_k operators are linear summations of the γ operators as seen in equation (3.4).

$$H_I = \lambda A \sum_k (1 - 2c_k^\dagger c_k) \quad (3.7)$$

$$H = \omega_c a^\dagger a + \sum_k \omega_k \gamma_k^\dagger \gamma_k + \lambda A \sum_k (1 - 2c_k^\dagger c_k) \quad (3.8)$$

There is one more transformation that we want to make. If we proceed from this we will find ourselves with extra fourth order diagrams that are not useful to the analysis. In order to simplify this we apply the unitary displacement operator $e^{\alpha a^\dagger - \alpha^* a}$ to the Hamiltonian. This has the effect of shifting the boson operators as in equation (3.9).

$$a \rightarrow a - \alpha \quad (3.9)$$

With the appropriate choice of α we can eliminate the constant term in the simulator's coupling operator. This reduces the number of diagrams we will have to consider going forward. An additional effect of this transformation is the modification of the magnetic field that the spins feel (eqn. 3.10). A detailed explanation of this shifting operation can be found in Appendix B. The final Hamiltonian is given in equation (3.11).

$$h'_x = h_x - \frac{4N\lambda^2}{\omega_c} \quad (3.10)$$

$$H = \omega_c a^\dagger a + \sum_k \omega_k \gamma_k^\dagger \gamma_k - 2\lambda A \sum_k c_k^\dagger c_k \quad (3.11)$$

At this point we are almost ready to expand our correlator to find the effect of the detector. However, instead of the typical correlator $-\langle T_\tau [A(\tau_f)A(\tau_i)] \rangle$ we consider a correlator where the operators are shifted $\hat{A} \rightarrow \hat{A} - \langle \hat{A} \rangle$. This removes diagrams which are connected to the initial point and the final point but are not fully connected [29]. The correlators can now be expanded exactly as is described in chapter 2.

We define the unperturbed correlators in equations (3.12) and (3.13) and their Fourier transform as equations (3.14) and (3.15).

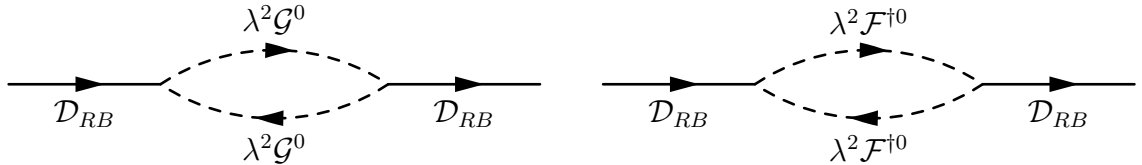
$$\mathcal{G}_k^0(\tau_1 - \tau_2) = -\left\langle T \left[c_k(\tau_1) c_k^\dagger(\tau_2) \right] \right\rangle_0 \quad (3.12)$$

$$\mathcal{F}_k^{\dagger 0}(\tau_1 - \tau_2) = -\left\langle T \left[c_k^\dagger(\tau_1) c_k^\dagger(\tau_2) \right] \right\rangle_0 \quad (3.13)$$

$$\mathcal{G}_k^0(\omega_n) = \left(\frac{|u_k|^2}{i\omega_n - \omega_k} + \frac{|v_k|^2}{i\omega_n + \omega_k} \right) \quad (3.14)$$

$$\mathcal{F}_k^{\dagger 0}(\omega_n) = i \left(\frac{u_k^\dagger v_k^\dagger}{i\omega_n - \omega_k} - \frac{u_k^\dagger v_k^\dagger}{i\omega_n + \omega_k} \right) \quad (3.15)$$

Now we can expand out the correlator and find the different order diagrams. The lowest non-zero correction is the second order. The diagrams can be seen below. They are similar in their shape but have different correlators associated with each line.



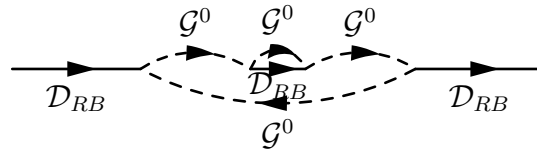
In principle the simulator's correlators can be separated from the photon correlators. The self-energy part of the diagrams relies solely on the simulator's correlators.

If these were they only diagrams then we could retrieve information about the simulator without any back-action from the cavity. If we represent the self-energy part with Σ^* then we see that measuring the cavity with and without the simulator coupled is sufficient to readout information about the simulator (eqn. (3.16)).

$$\Sigma^* = \left(\frac{1}{\mathcal{D}^0} - \frac{1}{\mathcal{D}} \right) \tag{3.16}$$

As we move to higher orders the situation changes. We will look at the fourth order corrections next which is the next non-zero correction to the photon correlator. A naive approach to expanding out the correlator to fourth order will yield over 2000 diagrams. However, shifting the photon operators reduces that number to 1575 with 32 unique diagrams. By examining the shifted correlator partially connected diagrams are removed and the number of unique diagrams drops down to 22. Sorting through these diagrams in a rigorous and thorough method is time consuming and prone to error. As a way of finding, sorting, and eventually calculating with the fourth order diagrams we designed and built a program to assist in carrying out the analysis.

The program consists of three modules, one which draws the Feynman Diagrams, one which translates the Feynman diagrams into equations and then carry's out the Matsubara summations symbolically, and an underlying module which defines the structures and operations necessary for the program to function. This was written in Python and a detailed explanation is provided in Appendix C as well as parts of the code itself. An example of one such diagrams is seen in equation (3.17).

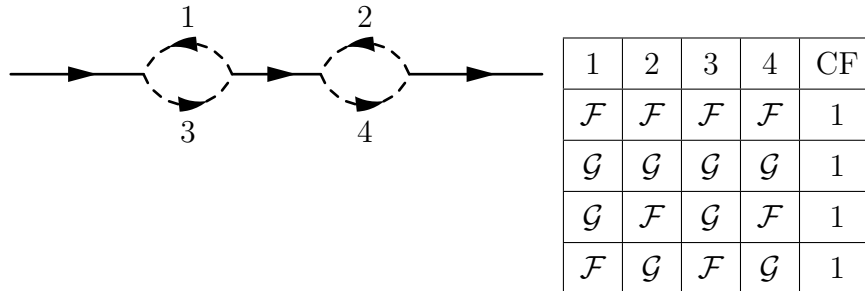


(3.17)

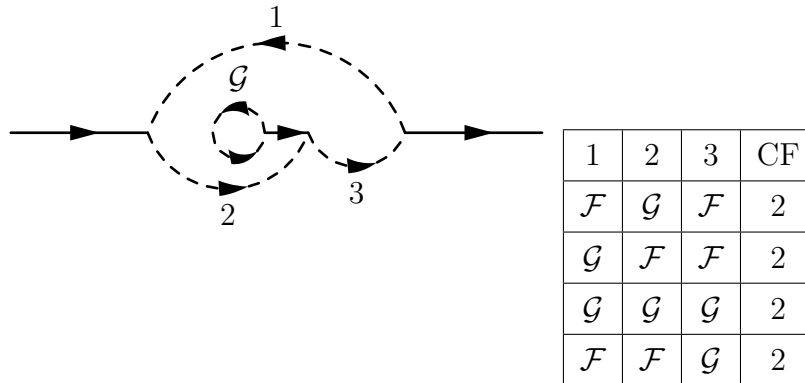
To simplify the visualization we can view the structure of the diagrams. Each diagram has the interaction lines labeled one through four. The table next to the diagram shows the possible sets of correlators that the interaction line can be. Additionally it shows the counting factor for each diagram. There are only 4 possible

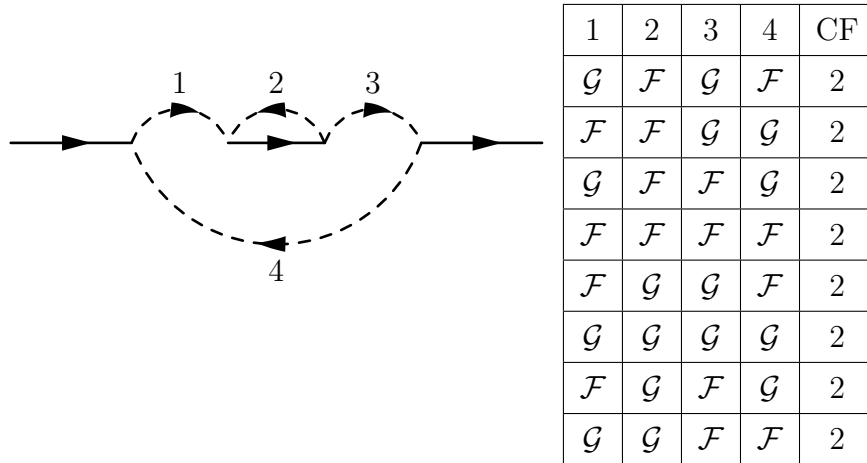
fully connected structures. The first structure is an improper diagram. It is simply the second order diagrams repeated. For the rest, the structure of the diagrams is more complicated. These complications mean that the system's unperturbed correlator cannot be extracted. Since the structure of the diagrams is determined by the form of the interaction Hamiltonian we can link this failure to the form of system's coupling operator. The quadratic nature of this operator allows the system's correlators to fold back onto itself to create increasingly more complicated interactions that don't permit reliable detection.

Our first class of diagrams is the loop diagrams that are due to the second order seen below. These diagrams are consider improper because they can be written as the product of second order diagrams. Thus they are already considered in the infinite order approximation discussed earlier. Therefore to avoid over counting they are not added in the proper self energy term.

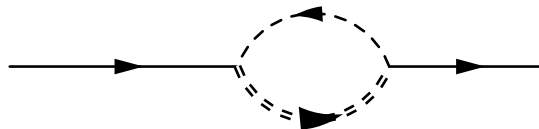


The next two classes of diagrams are considered to be effective interaction diagrams or polarization diagrams. These are diagrams where the higher order contribution can be removed by "cutting" interaction lines and replacing the contribution with an effective interaction.

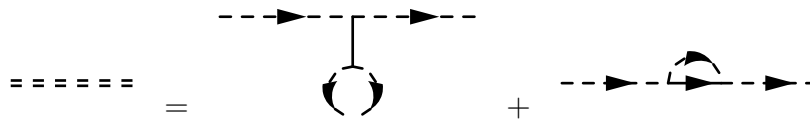




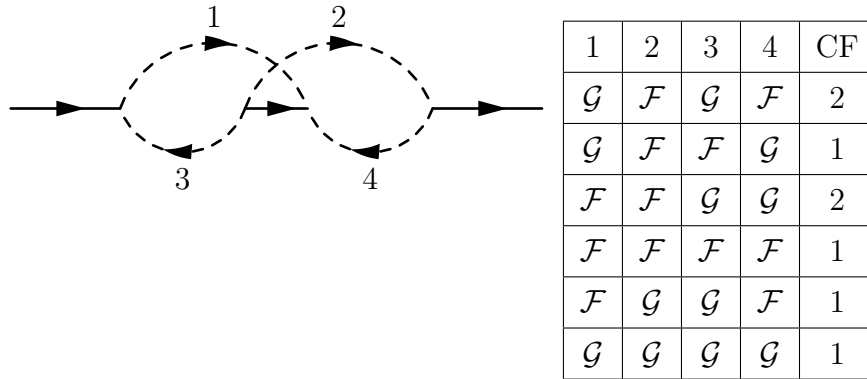
In the case of the effective interactions the counting factor of 2 represents the fact that the effective interaction can be in either the upper branch or the lower branch of the diagram. Going through this process of "cutting" out the higher order contributions and replacing it with an effective interaction gives the diagram seen below.



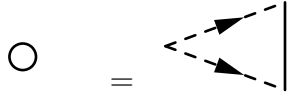
The effective interaction is the sum of the effective parts and is defined below.



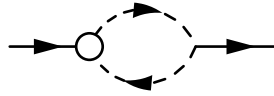
The last diagram is a vertex correction. This is the part which contributes most strongly to the cavity-simulator coupling.



By defining the vertex correction as:



The last diagram can be simplified to the following:



3.4 Approximate detection

We can use this approach to see how much the interaction with the cavity changes the system's correlator. Our first approach will be done by calculating the system's correlator to infinite order using the second order diagrams and then calculating the correlator to infinite order using the second and fourth order diagrams. Then the spectral function $\rho = -2\text{Imag}(\mathcal{D}_{ret})$ will be calculated for both and compared. The spectral function (also known as the spectral density or spectral weight) gives the transition energies of the system. It can be thought of as a generalized density of states. The diagrams were drawn and calculated using a home built program to facilitate analysis. The diagrams were used to ensure conservation of frequency across the correlators to perform the Fourier transform symbolically. A program was built on top of the Sympy package to perform the Matsubara sums symbolically and to get an analytic result for the Dyson equation using second and fourth order diagrams. The fourth order spectral function ρ drop below 0 around the peaks. This is likely

due to the process of analytically continuing the Matsubara correlator. However, by examining the absolute value of the spectral function we can still learn about the system due to the location of the peaks which give the transition energies.

Unless otherwise stated the parameters are as follows, $N = 20, J = 2\pi GHz, \frac{\omega_c}{J} = 12, \frac{\lambda}{J} = 0.04, \frac{h_x}{2J} = 0.02, \frac{\kappa}{J} = 10^{-4}, T = 0.1K$. We first examine

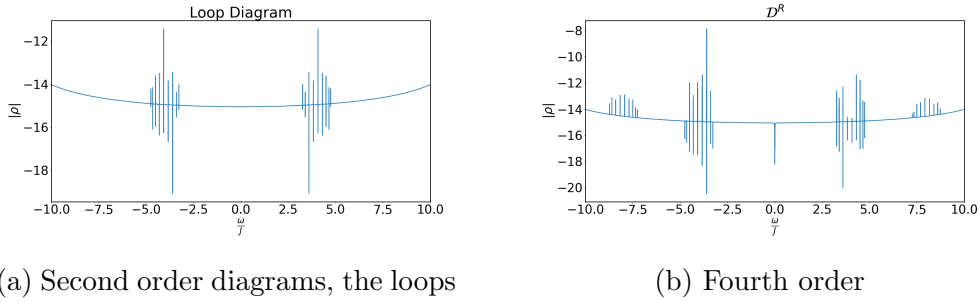
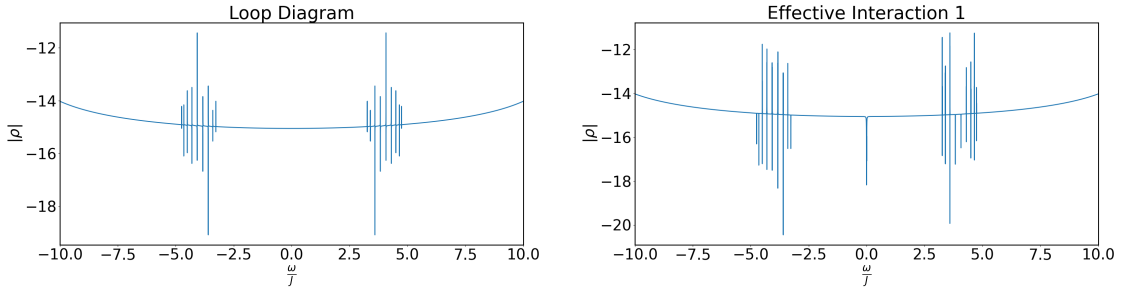


Figure 3.4: spectral density for $N=20$.

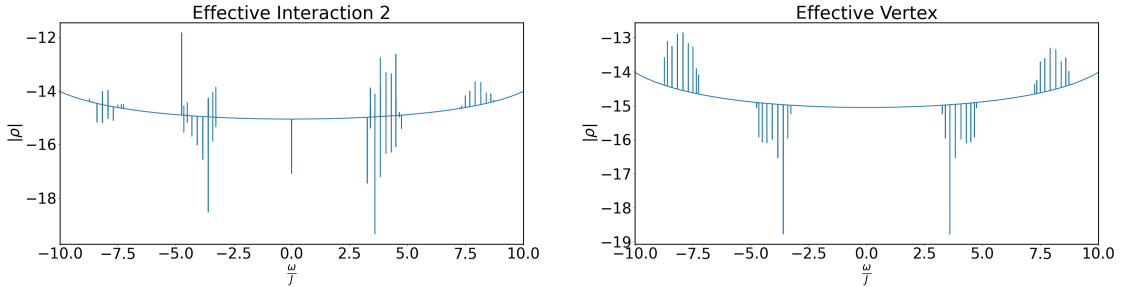
While the results are similar there are some noticeable difference. Firstly the inside peaks in the second order are due to the simulator and occur at $\pm 2\omega_k$. In the fourth order these peaks are largely present but some undergo a suppression and more than half undergo a splitting. In addition there are cavity-simulator peaks on the outside in the fourth order approximation. These are due to the system-cavity interaction and are found at $\pm(\omega_c - 2\omega_k)$.

The effect of the spectral function due to the four classes of diagrams is shown in figure 3.5. The first effective interaction modifies the peak heights but has little effect other than that. If we examine the diagram we can see why this is the case. Because the intermediate photon line connects to a simulator line that closes upon itself, the photon line carries no frequency. Thus the photon line and the closed simulator line both factor out of the diagram.

The second effective interaction similarly affects the main simulator peaks but it also causes cavity-simulator peaks to appear. Recall that each line on the diagram is itself a correlator. The effective interaction therefore is a correlator which will have poles shifted by an amount ω_c due to the Matsubara summation over the internal photon correlator.



(a) Spectral function using the lowest order diagrams (b) Spectral function with the first effective interaction



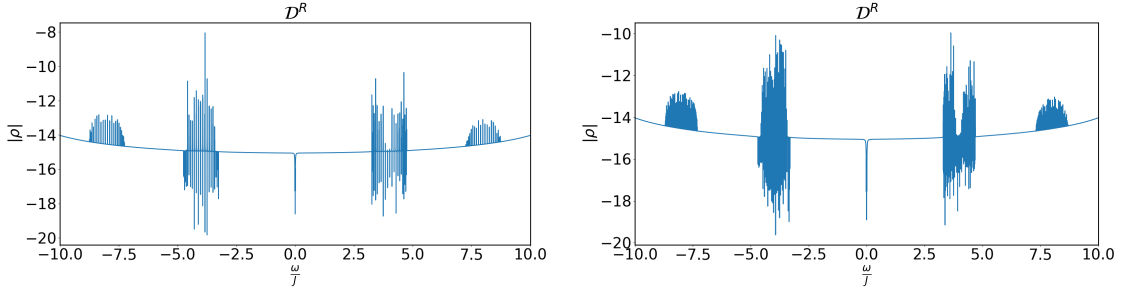
(c) Spectral function with the second effective interaction (d) Spectral function with the effective vertex

Figure 3.5: The spectral function is shown for cases with different classes of diagrams included.

The two effective interactions lead to a suppression of certain peaks due to partial cancellation. This suggests that examining the spectrum of the cavity directly may be problematic. The vertex part contributes the most strongly to the cavity-simulator peaks. This is due to the vertex part affecting both branches of the diagram.

We can examine the effect that a differing number of sites will have. As the number of sites, N , increases then more and more peaks relating to $2\omega_k$ appear. Looking at two additional cases below where $N = 50$ and $N = 100$ we can see the number of peaks increase. The number of peaks that are suppressed increases as well but not at the same rate as the total number of peaks.

The splittings for each individual peak undergo stays constant with respect to the number of sites. The average splitting was around $0.0012J$ for all values of N . Pursuing this line of analysis provides insight into the cavity-simulator system

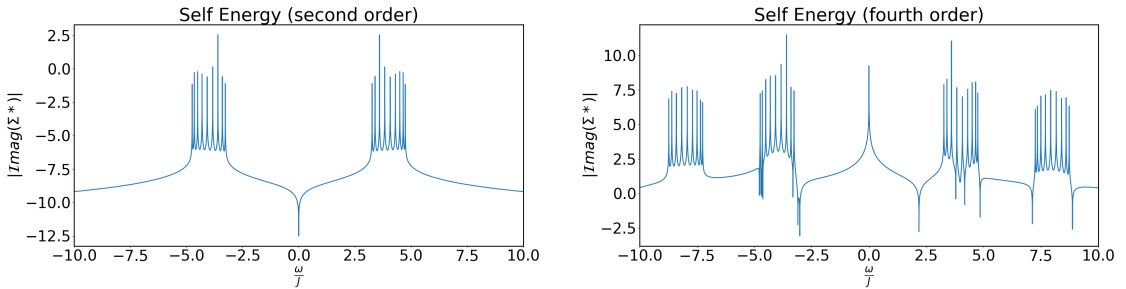


(a) Fourth order spectral function with $N = 50$ (b) Fourth order spectral function with $N = 100$

Figure 3.6: Spectral density for different number of sites. As the number of sites increases the number of lines in the spectrum also increase. However, the splittings caused by the interaction with the cavity does not change. Parameters are $\frac{\omega_c}{J} = 12$, $\frac{\lambda}{J} = 0.04$, $\frac{h_x}{2J} = 0.02$, $\frac{\kappa}{J} = 10^{-4}$, $T = 0.1K$

but makes analysis of the simulator itself difficult. We can completely ignore the difficulties associated with the splittings and peak suppression by examining the self energy as in equation 3.16.

The second order self energy term is not directly affected by the cavity and the cavity frequency does not directly come into the spectrum. The peaks seen in figure 3.7 (a) can be found at twice the frequency of the simulator. Measuring these peaks can then give information about the simulator itself.



(a) Second order retarded self energy (b) Fourth order retarded self energy

Figure 3.7: Using equation 3.16 we can calculate the self energy. This eliminates the splittings in the peaks caused by the cavity allowing for less ambiguous results. The parameters are the same as in figure 3.6 with $N = 20$.

In the fourth order self energy the effects of the cavity can be seen in the cavity-simulator peaks which occur at $\pm(\omega_c - 2\omega_k)$. As long as the two sets of peaks remain far apart then we can unambiguously distinguish the peaks due to the simulator and those due to the cavity-simulator system. This occurs as long as the cavity frequency meets the condition in equation 3.18. This condition can be seen by recognizing that the cavity-simulator peaks need to be outside of the simulator peaks which occur at $2\omega_k$. Since the cavity-simulator peaks occur at $\omega_c - 2\omega_k$ the cavity frequency will need to be more than four times the simulator's frequency.

$$\omega_c > 4\omega_k \quad (3.18)$$

To find the effect of the different parameters on detection we must consider two aspects. The first is the unambiguous distinction between the simulator peaks and the cavity-simulator peaks as discussed above. The second is in the reliable extraction of the simulator's parameters. We can attempt this by fitting the data to the form of the second order expression which only depends on the simulator. The second order self energy term is written as:

$$\frac{4\lambda^2}{\hbar\beta} \sum_k \sum_{i\omega_1} \mathcal{G}_k^0(i\omega_1) \mathcal{G}_k^0(i\omega_n + i\omega_1) - \mathcal{F}_k^{\dagger 0}(i\omega_1) \mathcal{F}_k^{\dagger 0}(i\omega_n + i\omega_1) \quad (3.19)$$

After completing the Matsubara summations using the technique discussed in chapter 2 we get the solution:

$$8\lambda^2 \sum_k (1 - 2n_F(\omega_k)) \left(\frac{4\omega_k u_k^2 v_k^2}{(i\omega_n)^2 - (2\omega_k)^2} \right) \quad (3.20)$$

This has two issues, firstly the height of the peak is difficult to properly extract both numerically and experimentally. Secondly, the effects of the cavity are more pronounced in the peaks of the self energy. A combination of peak finding and inspection would work better to extract out the simulator's frequencies in practice. Here we simply use the calculated values since the difference between that and a peak finding algorithm are small and depends solely on the numerical resolution. We must also consider the effect of the effective magnetic field on the simulator's

frequency. Since the magnetic field is shifted by an amount $-\frac{4N\lambda^2}{\omega_c}$ the measured value of simulator's frequency will also be altered. To examine the effect of the cavity on the simulator's frequency we look at the average of the absolute value of the percent difference eqn.(3.21)

$$pd = 100 \left| \frac{\omega_k - \omega'_k}{\frac{\omega_k + \omega'_k}{2}} \right| \quad (3.21)$$

We start by examining the effect of the cavity frequency. As the frequency increases the cavity-simulator peaks are pulled farther away. For small cavity frequencies the cavity-simulator peaks cannot be easily distinguished from the simulator's peaks as shown in figure 3.8.

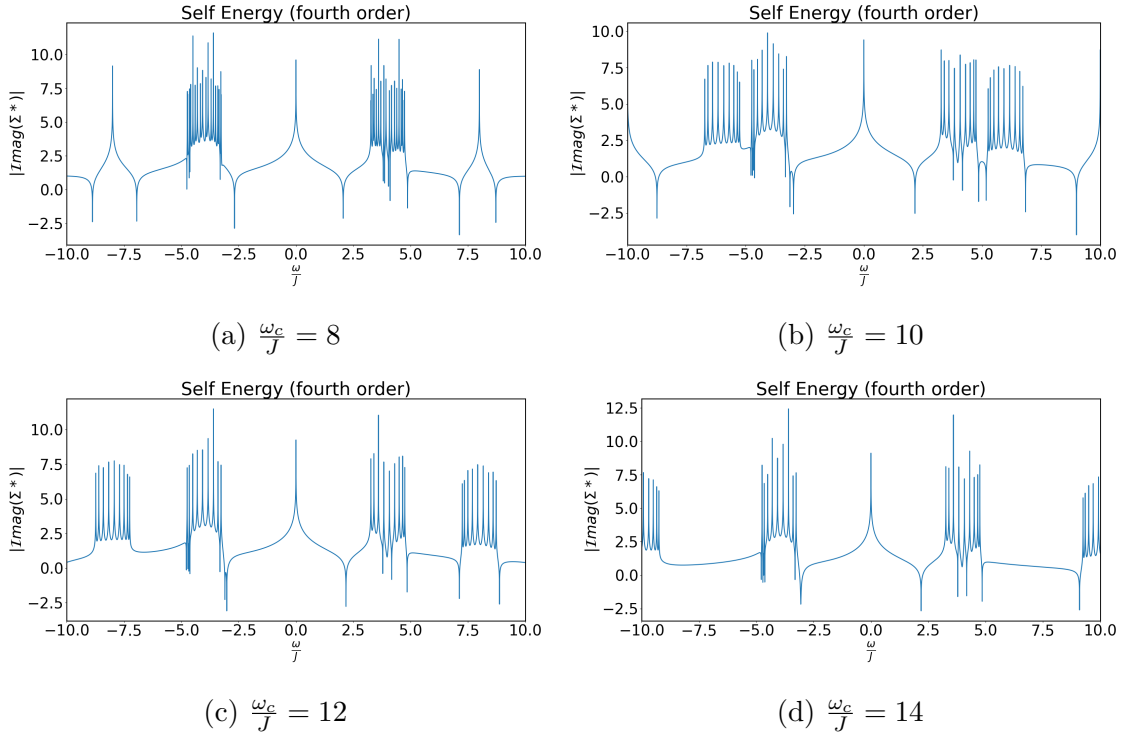


Figure 3.8: The spectral function for varying values of cavity frequency. As the cavity frequency increase the cavity-simulator peaks pull away from the simulator peaks.

In the case where $\frac{\omega_c}{J} = 8$ the cavity-simulator peaks are mingled with the simulator peaks and cannot be distinguished. As the cavity frequency increases the

percent difference between the unperturbed and perturbed simulator frequency decreases roughly as $\frac{1}{\omega_c}$. Thus having a high cavity frequency is desirable for both accurate measurement of the peaks and distinguishability between the peaks due to the simulator and those due to the cavity-simulator interaction.

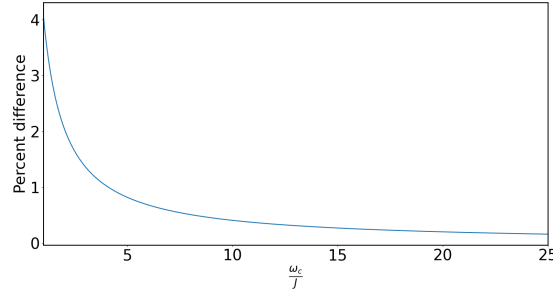


Figure 3.9: The percent difference in the simulator's frequency as a function of cavity frequency

We can see a similar behavior as we increase the magnetic field, h_x . As h_x increases the frequency of the simulator also increases, eventually causing the cavity-simulator peaks to mix with the simulator's peaks. At $\frac{h_x}{2J} = 0.5$ The maximum simulator frequency is just under one fourth of the cavity frequency. The cavity-simulator peaks are still distinguishable but just barely.

The effect of the accuracy of the location of the peaks is effected in a more pronounced fashion. Since the correction to the magnetic field is fairly small for the given parameters, less than 3% there is little change to the measured frequencies. As h_x approaches the critical point and the effective critical point the percent difference between the simulator's frequency and the measured frequency peaks.

An important aspect to examine in this discussion is the number of sites. The appeal of quantum simulators is tackle problems a classical simulator simply cannot handle. Smaller quantum systems can usually be simulated. As the number of sites increases the number of peaks also increases. For a set effective magnetic field the peaks stay within the same range getting closer and closer together. However, the number of sites alters the effective magnetic field causing a change in the measured simulator frequencies. This effect can be seen in figure 3.12 by examining the space in

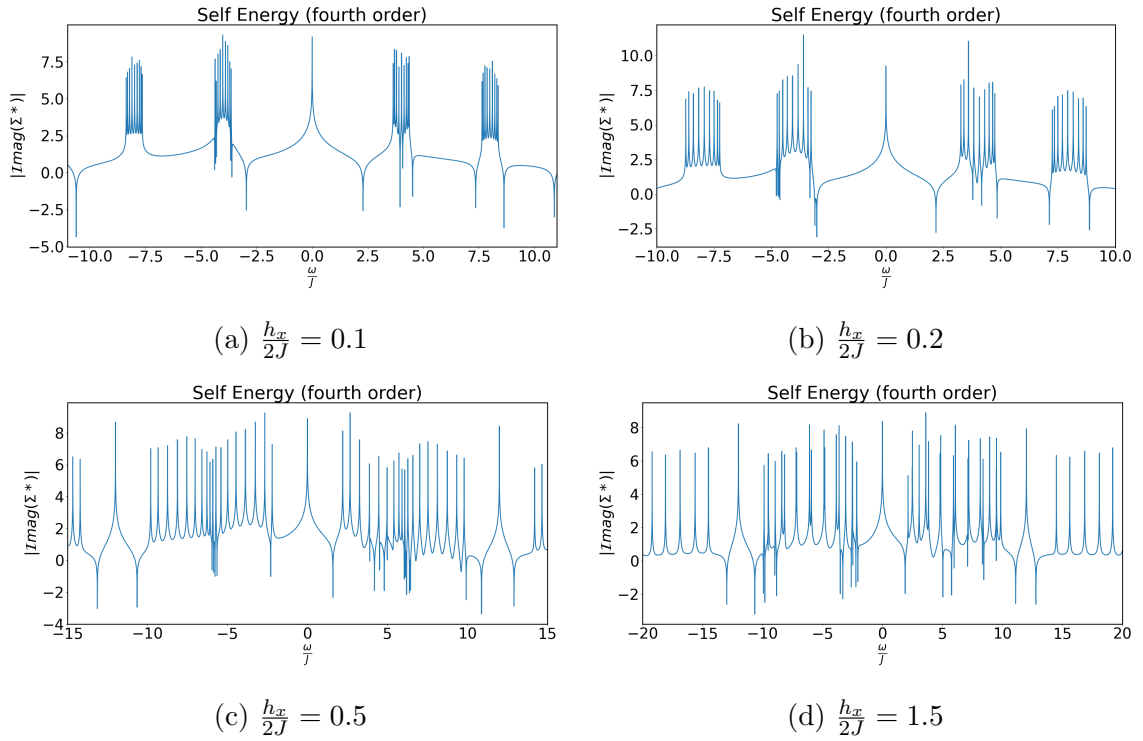


Figure 3.10: The spectral function for varying values of magnetic fields. As the magnetic field increases the frequency of the simulator spreads out, eventually causing the simulator peaks to overlap with the cavity-simulator peaks.

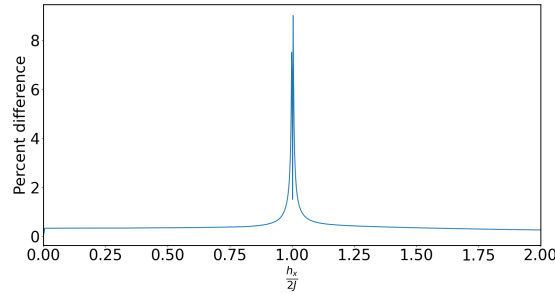


Figure 3.11: The percent difference in the simulator’s frequency as a function of the applied magnetic field. Two peaks appears, one due to the magnetic field reaching the critical point and the other due to the effective magnetic field reaching the critical point $\frac{h_x}{2J} = 1$.

which the peaks are residing. For $N = 1000$ we see that all of the peaks are squished together. However, as the number of sites continues to increase the spacing opens up.

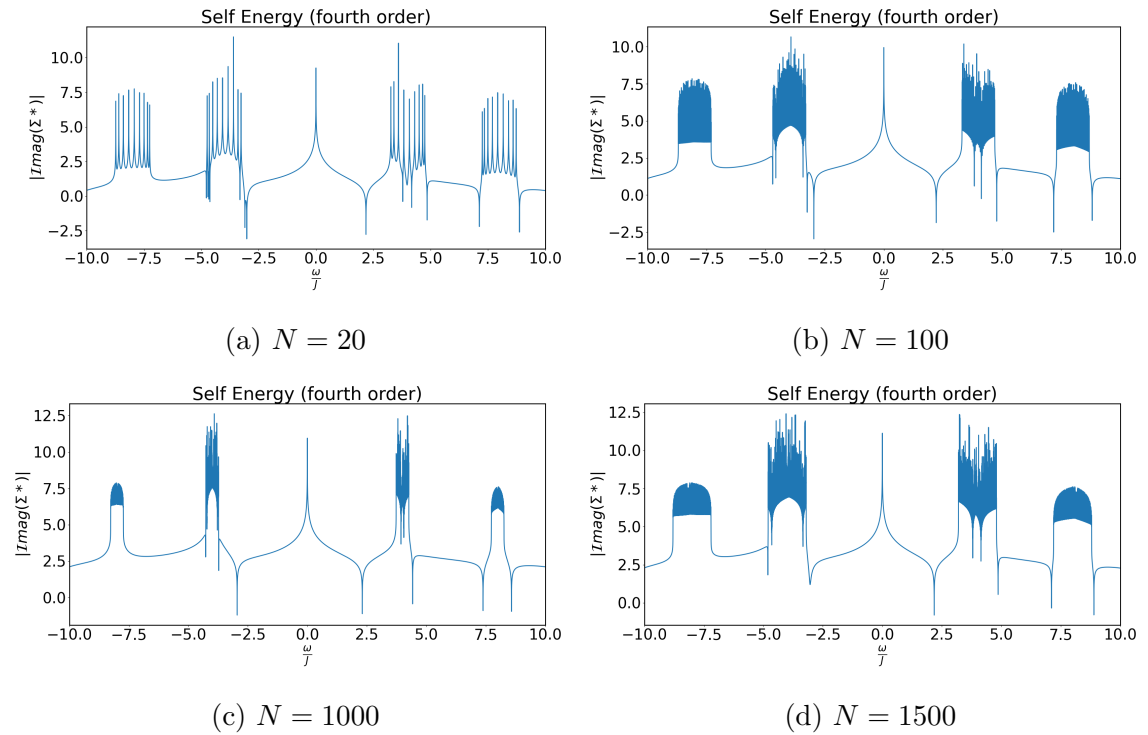


Figure 3.12: The spectral function for varying number of sites. As the number of sites increases the effective magnetic field decreases eventually turning negative.

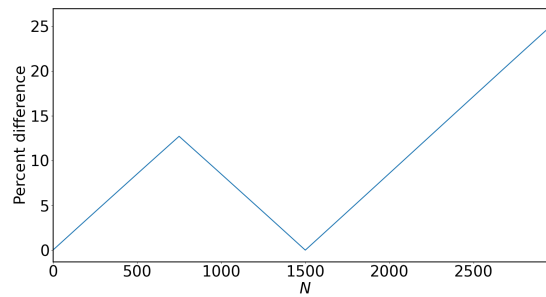


Figure 3.13: Percent difference in the simulator's frequency as a function of the number of sites. As the sites increase the effective magnetic field decreases, eventually flipping direct. This causes the measured and actual frequencies to match perfectly at $N = 1500$. Beyond that point additional sites causes the difference to increase monotonically.

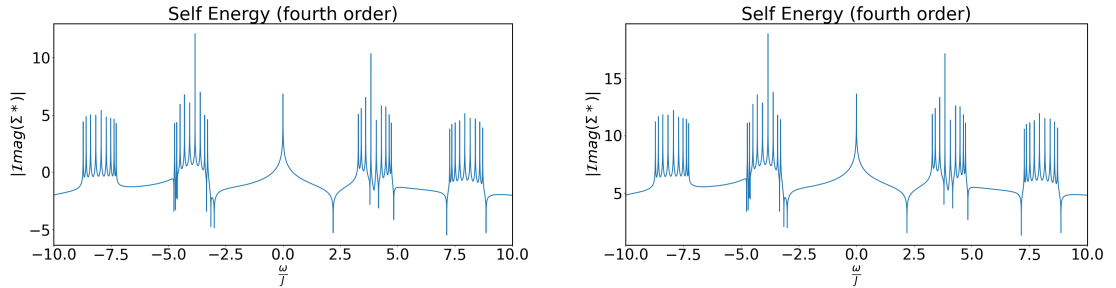
The reason for this is that as the correction to the magnetic field grows eventually the field is canceled. For these parameters this occurs when $N = 750$. This can be

used as an advantage. As the correction to the magnetic field continues to increase eventually the effective magnetic field will change direction. If the parameters are tuned just right the effective magnetic field can be tuned so that it is the same magnitude but differing in direction to the decoupled case. In this case there is no difference in the expected and the measured simulator frequency. This is the cases in figure 3.12 (d). This effect is encapsulated in figure 3.13 where the percent difference is seen to increase, then decrease to zero before increasing again. Thus truly reliable detection can be achieved. This occurs under the following condition:

$$h'_x = -h_x = h_x - \frac{4N\lambda^2}{\omega_c} \quad (3.22)$$

$$h_x = \frac{2N\lambda^2}{\omega_c} \quad (3.23)$$

Pushing this further we can examine the effect of a higher coupling $\lambda = 0.5J$ if the magnetic field is adjusted so that the effective magnetic field stays the same.



(a) Retarded self energy term, $\frac{\lambda}{J} = 0.01 \frac{h_x}{2J} = 0.1897$ (b) Retarded self energy term, $\frac{\lambda}{J} = 0.5 \frac{h_x}{2J} = 1.022$

In this case the different sets of peaks can be easily differentiated. While the height of the peaks are significantly affected the location remains the same. This shows that the effect of the coupling can be circumvented with respect to detection so long as the other parameters can be tuned.

In general this shows an interesting case where by fine-tuning the parameters accurate measurement of the simulator's peaks can be achieved. From a practical standpoint this may be problematic as the tuning of these parameters may not be

easily achieved. The details of this depend on the specific system. For any array of trapped ions it may be possible to change the number of sites a superconducting architecture may not have that luxury.

Finally turning to the effect of the bandwidth on the detection of the simulator peak we find an interesting use. As the bandwidth increases the height of the cavity-simulator peaks is suppressed and the peaks become more spread out as seen in figure 3.15. This is due to the fact that the effect of the bandwidth is to alter the poles of the cavity's correlator giving it a non-zero imaginary part. The imaginary part gives the full-width-half-max of the peak. Since the poles from the simulator do not depend on the cavity directly they are not affected by this. Thus the cavity-simulator peaks can be independently controlled.

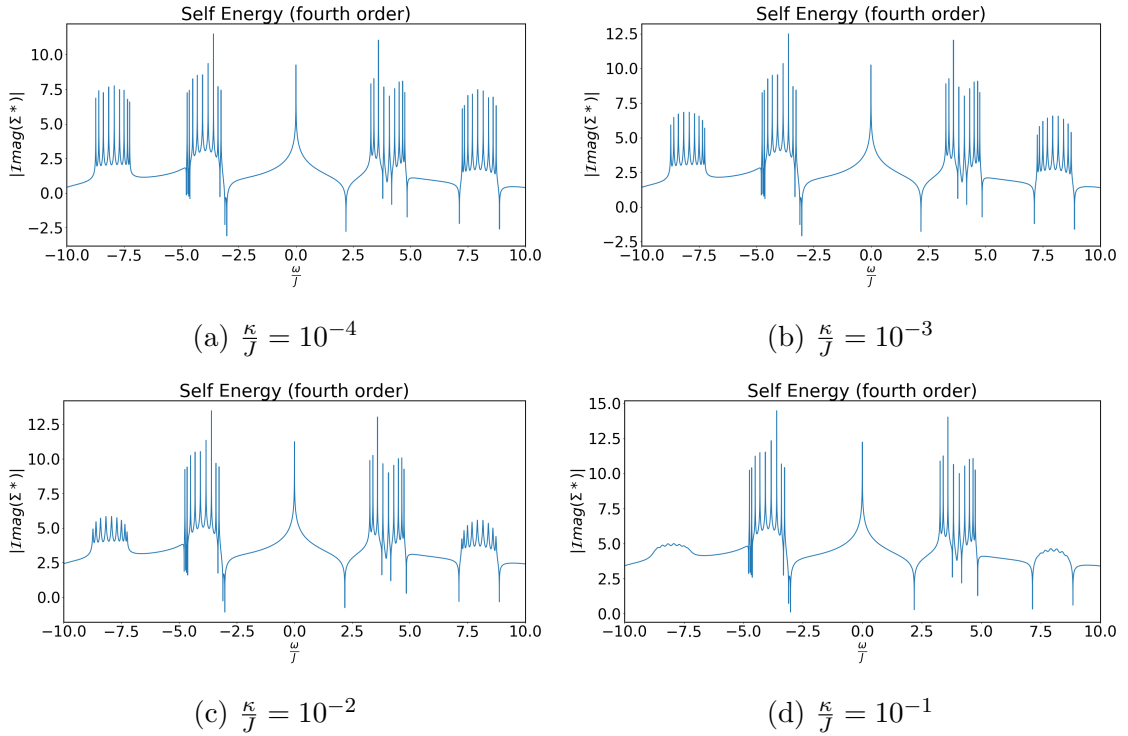
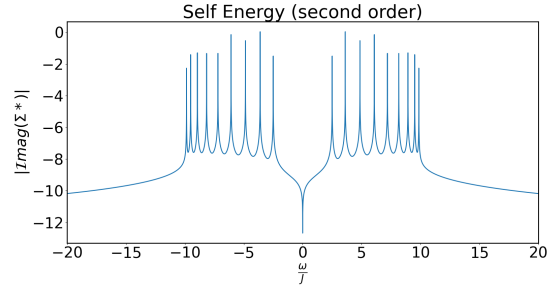


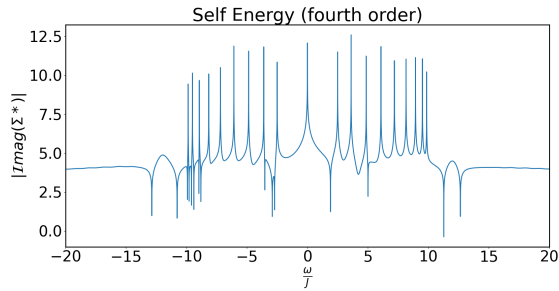
Figure 3.15: The spectral function for varying cavity bandwidth κ . As the bandwidth increases the cavity-simulator peaks are suppressed.

Using this fact we can examine stronger magnetic fields so long as the band width is sufficiently high. In the graphs below we look at the case where the magnetic field

is 3 times J and the band width is $0.5J$. This greatly suppresses the cavity-simulator peaks and allows for the simulator's peaks to be read out. Even though the energy of the simulator does not meet the condition in eqn. (3.18) with a max value of $5J$, the cavity-simulator peaks have been suppressed to such a large extent that the simulator peaks are easily distinguished.



(a) The self energy calculated directly by only considering the loop diagrams.



(b) The self energy calculated considering all the diagrams.

Figure 3.16: Despite having a high magnetic field reliable detection can still be achieved because the large bandwidth suppresses the side band peaks. The second order peaks seen in (a) can also be distinguished in the (b). Parameters are $\frac{\kappa}{J} = 5 \times 10^{-1}$, $\frac{h_x}{2J} = 1.5$

As the higher order terms broadened and are suppressed the peaks from the cavity are broadened as well. Therefore care should be taken to keep the broadening of the cavity peaks from overwhelming the simulator's peaks. Thus increasing the bandwidth of the cavity acts to soften the condition that the cavity frequency should be far from the simulator's frequency.

Previous work done by Du and Tian estimated the back action from this set up [18]. Their estimate used finite second order perturbation theory to estimate the shift $\delta\omega_k$ in the eigenmodes to be $|\delta\omega_k| \approx 2\frac{\lambda^2 N}{\omega_c}$. By examining the effect of the cavity shift on the simulator we can find a similar shift. The value of the simulator's frequency $\omega_k = 2J \left(1 + \left(\frac{h_x}{2J}\right)^2 - \frac{h_x}{J} \cos(k)\right)^{\frac{1}{2}}$ is changed due to the effective magnetic field giving a frequency ω'_k . Defining $\delta\omega_k = \omega_k - \omega'_k$ squaring the results, making a binomial expansion, simplifying and then examining the leading term gives a result of $|\delta\omega_k| \approx 4\frac{N\lambda^2}{\omega_c}$ which is in good agreement with Du's result.

An important question is how our results extend to other systems, in particular systems with an unknown eigenvalue spectrum. This is the key interest of these types of adiabatic quantum simulators. While specific conclusions regarding and numerical answers regarding the reliability of the results are likely not valid this work provides some insight into approaches and parameter regimes that future work may want to consider. Our conclusion that the simulator's coupling operator quadratic nature leads to issues with detection remains valid for all systems. This is because the coupling determines the structure of the Feynman diagrams and a quadratic simulator coupling operator will lead to diagrams intertwined with the photon lines. Additionally no matter the type of simulator or the simulator's coupling operator as long as the detector coupling is linear the first non-zero contribution in the perturbation series will solely depend on the simulator terms. Higher order terms will include photon lines which carry Matsubara frequencies that must be summed over. It is because of this feature that the cavity-simulator peaks depend on the bandwidth. Thus for all foreseeable systems we would expect a higher bandwidth to suppress the cavity-simulator peaks while leaving the pure simulator peaks untouched.

For a similar reason we also expect that the cavity frequency will need to be greater than the maximum simulator cavity by $2nMAX(\omega_s)$ where n is the number order of the simulator coupling operator and ω_s is the simulator's frequency. This is because each additional term in the coupling operator generates a new line and thus another Matsubara summation is required. One of the effects of summing over the Matsubara frequencies is to alter the poles of the resulting correlator by adding (subtracting) the original pole. This is only true in the case where the simulator's

correlators have two each and opposite poles.

In a more intuitive fashion, we consider a vertex of the Feynman diagram with two simulator lines. The simulator's correlators have poles at $\pm\omega_s$. A value of $i\omega_n = 2\omega_s$ can flow through the diagram with a $+\omega_s$ flowing out and a $-\omega_s$ flowing in to the vertex. Thus in that diagram $2\omega_s$ is a pole.

In the case of vertex with three simulator lines entering or exiting a value of $3\omega_s$ can pass through the diagram in a similar fashion. Thus $3\omega_s$ is a pole. Therefore to avoid overlap between the simulator peaks and the cavity-simulator peaks we would expect that the cavity frequency would need to be at least $6\omega_s$.

3.5 Conclusions

We analyzed the Transverse Field Ising Model with nearest neighbor interaction coupled to a cavity readout. We found that the form of the coupling prevents reliable readout due to the quadratic nature of the simulator's coupling term. We found that despite being able to achieve exact detection that reliable detection can still be achieved. This is done by considering two features. Firstly the distinguishability of the simulator's peaks and secondly the accuracy of the simulator's peaks. In the first case the simulator's peaks can be mixed with the simulator-cavity peaks due to higher order corrections. These simulator-cavity peaks are located at $\pm(\omega_c - 2\omega_k)$ thus to distinguish the peaks the cavity frequency must be far enough away from the simulator's frequency, $\omega_c > 4\omega_k$. This condition can be mitigated in the case of higher bandwidth since the simulator-cavity peaks are sensitive to the bandwidth and will be suppressed where the simulator peaks are not.

The accuracy of the peaks is determined by the effective magnetic field which is due to properties of the simulator, the cavity, and the coupling between them. Instead of reading out results for an applied magnetic field of h_x the actual results are for an effective magnetic field $h_x - 4\frac{\lambda^2 N}{\omega_c}$. For small coupling, small sites, and large cavity frequency this effect may be negligible; however, as this correction increases the effect becomes more pronounced until the effective field is flipped, eventually the effective field is the same as the applied field only in the opposite direction. In a symmetric

system such as ours this will give the same result. After this point the differences between the ideal result and the actual result increases roughly linearly. If there is full experimental control of the system's parameters then this effect can be mitigated by changing the parameters. For example in a trapped ion system the number of captured atoms may be variable. However, in a superconducting system the number of sites is fixed. Thus the ability to mitigate this particular effect is strongly depend on the type of system and experimental controls that are possible.

Finally though our results are for specific case of th Transverse Field Ising model coupled to a cavity several of the most important results generalize to other systems with a linear detector coupling operator. The effect of bandwidth suppression on the cavity-simulator peaks will likely hold for most systems because the higher order terms involve Mastubara summations over the detector's correlations. Additionally if the simulator's correlators have peaks that come in positive-negative pairs then we can be fairly certain that the detectors frequency will need to be greater than $2n\omega_s$ to ensure distinguishability between the simulator and the cavity-simulator peaks. Thus this works offers insight in to the reliable detection of analog quantum simulators.

Chapter 4

Qubit

4.1 Introduction

Qubits have been proposed for use in universal quantum computers [4] with various systems used in their experimental realization [43–45]. Cavity-qubit systems in the strong coupling regime (where cavity-qubit coupling is greater than the decay rates) have been studied in-depth within the Rotating Wave approximation (RWA) [46,47]. The RWA involves removing terms like $a\sigma^-$ and $a^\dagger\sigma^+$ and is only valid for coupling strengths significantly less than the cavity frequency [48]. These are often called the counter rotating terms. The regime is properly understood by the Jaynes-Cumming model. Because the counter-rotating terms are removed only states which trade off excitations between the photon mode and the qubit are allowed. This leads to a subspace in the Hilbert space that is described by the eigenvectors $|\uparrow, n\rangle$ and $|\downarrow, n+1\rangle$. In this regime anti-bunching in the photon statistics is observed which is indicative of the quantum nature of the photon. The interaction between the qubit and the cavity also leads to a shift in the cavity spectrum which is dependent on the state of the qubit [46]. This can be used for readout of the qubit state.

Work has been done to expand results to stronger and stronger couplings [22,23,26,49,50]. There is an interest in possible new physics and interesting effects including a break down of the Purcell effect, entangled ground states, non-classical light, ultra fast quantum gates, and more [28,51–55]. Kohler used an approach that involved

finding the density matrix without the rotating ave approximation and then used that in a linear response theory to find the susceptibility and transmission spectra. Additionally experimental setups have reached into "Deep-strong-coupling" where the coupling strength approaches the cavity frequency [1,26]. In our work we do not make the RWA and thus we are free to explore the physics beyond this approximation.

4.2 First approach

We start with the Hamiltonian for the cavity-qubit system.

$$H = \hbar\omega_c a^\dagger a + \hbar\frac{\omega_z}{2}\sigma^z + \hbar\lambda A\sigma^x \quad (4.1)$$

Where $A = a + a^\dagger$. If we wish to expand this using Feynman diagrams we run into a problem discussed in chapter 2. The interaction term is made up of anti-commuting operators. During the derivation of the perturbation theory there is an implicit assumption that interaction Hamiltonian behaves as a boson with respect to the commutation laws. This leads to a inconsistency between the time ordering operator in definition used for the correlators and the expansion of the time-order exponential.

Three approaches were taken to handle this issue. The first treat the spin operators with a boson like time ordering. The second approach is to ignore the inconsistency and proceed. This is the most naive approach but there are some interesting insights that develop from these two approaches. Finally we use a Majorana transformation to fix the interaction term. In most of these cases we set the bandwidth to be zero for ease of comparison.

Firstly we consider a slightly more general definition of the time ordering operator seen in equation (4.2). I have shown that non-relativistic perturbation theory can be self consistently built for any real value of θ . Here we will just assign it the value of 2π so that the time ordering is similar to that of bosons. I will refer to the value $e^{i\theta}$ as the exchange factor. More generally we can have any value for θ so long as the total exchange factor for the interaction term is 1. For example using an exchange

factor of i and $-i$ for the photon and spin exchange factor will work though solving the Matsubara sums requires a different weighting function.

$$T_\tau[\sigma^x(\tau_f)\sigma^x(\tau_i)] = \Theta(\tau_f - \tau_i)\sigma_f^x\sigma_i^x + e^{i\theta}\Theta(\tau_f - \tau_i)\sigma_f^x\sigma_i^x \quad (4.2)$$

The sign change from the time ordering is what causes discrete frequencies and determines the type. In this case the spins will have boson like frequencies which are even. There are two advantages to this. The first is that the diagrams that are produced are simple. The second is that the main spectral lines are captured by this theory. However, we will see that transitions from higher energy states are not captured and the theory can not be altered to address this.

The correlator for the unperturbed qubit is defined by equation (4.3) and its Fourier transform is seen in equation (4.4). This equation only applies for the current exchange factor of $+1$.

$$\mathcal{S}^0(\tau_f, \tau_i) = -\langle T_\tau[\sigma^x(\tau_f)\sigma^x(\tau_i)] \rangle_0 \quad (4.3)$$

$$\mathcal{S}^0(i\omega_n) = \frac{2\omega_z}{(i\omega_n)^2 - \omega_z^2} \quad (4.4)$$

We have shown that Wick's theorem applies to the raising and lowering spin operators for a single qubit. The derivation is very similar to that of Fetter [29] and is worked out in Appendix A. Because the spin operator is just a linear combination of raising and lowering operators each vertex in the diagram has only one photon line and one spin line connected. This leads to one simple diagram.

$$\begin{array}{c} \text{---} \blacktriangleright \text{---} \text{---} \text{---} \text{---} \text{---} \blacktriangleright \text{---} \\ \mathcal{D}_{RB} \quad \lambda^2 \mathcal{S}^0 \quad \mathcal{D}_{RB} \end{array} \quad (4.5)$$

Going to higher and higher orders we see this is the only proper diagram. This can also be seen by recognizing that the only way to add a diagram is to insert two vertices, a single photon line, and a single spin line. We can place this diagram into the self energy part of Dyson's equation to find the full correlator.

$$\overline{\overline{\mathcal{D}_R}} = \overline{\mathcal{D}_{RB}} + \overline{\mathcal{D}_{RB}} \overline{\lambda^2 \mathcal{S}^0} \overline{\mathcal{D}_R} \quad (4.6)$$

This diagram is then converted into an equation which we can use to calculate results (eqn(4.7))

$$\mathcal{D}_R(\omega_n) = \frac{\mathcal{D}_{RB}(\omega_n)}{1 - \lambda^2 \mathcal{D}_{RB}(\omega_n) \mathcal{S}^0(\omega_n)} \quad (4.7)$$

To analyze our results we compare our theoretical predictions to numerical diagonalization with values based on an experiment by Yoshihara. This paper looks at the transmission spectra for a superconducting qubit-oscillator system in the deep coupling regime. In the experimental papers we compare our results to the Hamiltonian used to describe the cavity-qubit system has an additional term $\frac{\varepsilon}{2} \sigma^x$ due to an applied external flux. This external flux is a free parameter by which the system can be studied. Additionally we rotate the Hamiltonian so that their coupling matches ours. This flux term is handled by diagonalizing the qubit part of the free Hamiltonian and rewriting the coupling. This leads to an extra term in the coupling which is unimportant to this part of the results as it leads to disconnected diagrams.

The spectra is obtained using a weak probe to scan across the transmission range. In figure 4.1 (a) we can see the relevant eigenvalue transitions that occur in the experiments and the spectral function calculate from the eigenvalues using the Lehmann representation of the Green's function. These graphs are obtained by numerically solving for the eigenvalues using a 16 photon basis. The results are seen in figure 4.1 which agree well with the experiment. The ratio of the coupling strength to the cavity frequency is 0.72 which is considered to be in the deep strong coupling regime.

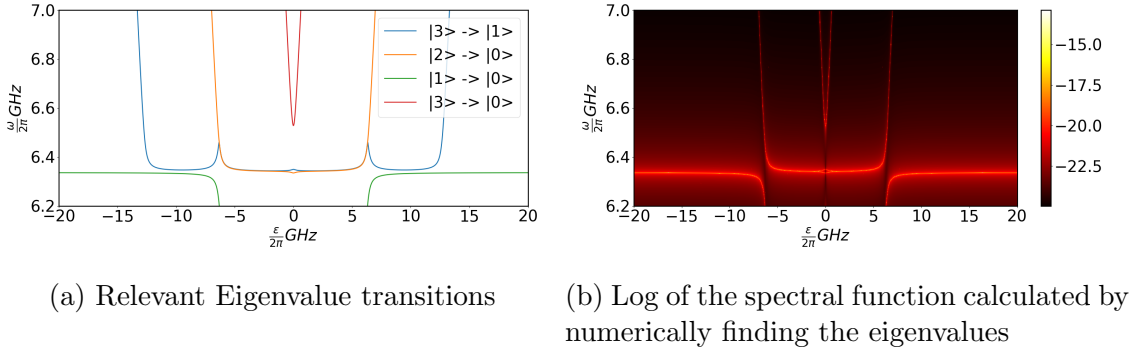


Figure 4.1: These figures reproduce the results found in [1]. The values are $\frac{\omega_c}{2\pi} = 6.336GHz$, $\frac{\lambda}{2\pi} = 4.57GHz$, and $\frac{\omega_z}{2\pi} = 0.505GHz$. The temperature is at 20mK.

We can now compare our approach. Using the Green’s function we calculated via perturbation theory we plot the results for the spectral function in figure 4.2. The main transmission line is close corresponding to the lowest excited state transitioning to the ground state. The second excitation to ground state transition is partially captured but the behavior around $\epsilon = 0$ is completely incorrect. Additionally none of the other transitions are seen. Because of the lack of Matsubara summations, there is also no temperature dependence. Thus while there is some agreement with the primary transmission peaks this approach lacks the ability to determine higher state transitions.

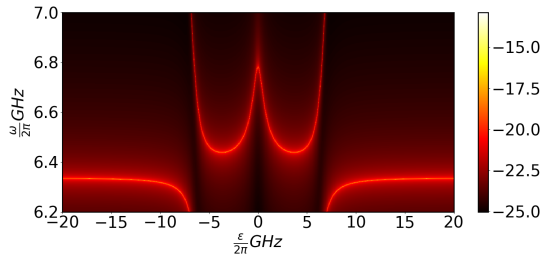


Figure 4.2: Spectra for the cavity qubit system calculated with the boson-boson assumption on a log scale. The parameters are the same as in figure 4.1.

We can also compare this within the rotating wave approximation (figure 4.3). In this case the second excited state transition is even farther from the experimental spectra but the two are otherwise fairly close. This is suggestive that our approach is missing certain key aspects. However, there is an advantage to this approach if you are only concerned with the primary peaks and are at lower coupling.

It can be easily extended to include a multi-mode cavity like was done by Chen in [2]. This was an experiment involving a superconducting flux qubit coupled to a coplanar waveguide resonator. In modeling the system the three lowest modes of the resonator were considered, each with a different coupling strength. The maximum coupling was approximately 10% of the resonance frequency of the mode. This puts the system in to the ultra-strong coupling regime and the Jayne’s Cumming model is expected to not be valid.

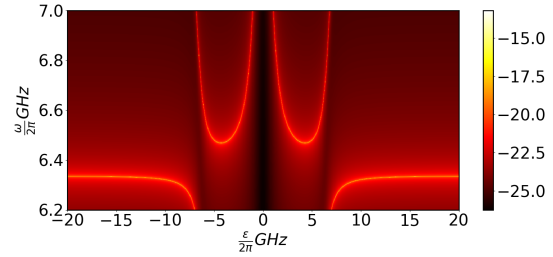


Figure 4.3: Spectra for the cavity qubit system calculated in the rotating wave approximation on a log scale. The parameters are the same as in figure 4.1.

The normal photon correlators are upgraded to by an $N \times N$ matrix where N is the number of modes. For the three mode cavity system in [2] these results capture the

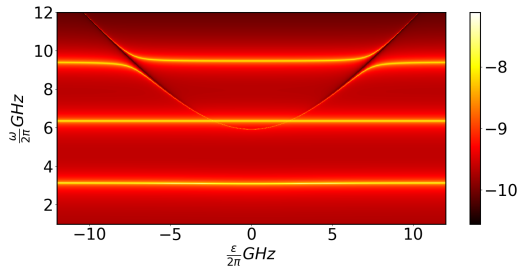


Figure 4.4: Log of the spectra for a cavity-qubit system where the cavity is modeled with 3-modes. Parameters are found in [2].

main lines. However, the side bands due to higher energy transitions are not caught by this method and the avoided crossing behavior between the is not quite right. In this case the rotating wave approximation only makes a very small difference. This indicates two things. The first is the our approach is not properly taking into account the counter-rotating terms. The second is this approach as a whole has no chance

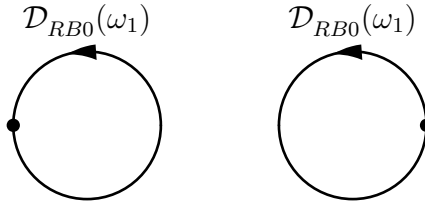
of capturing the higher order transitions we expect in systems like these.

4.2.1 4-point correlator

A final advantage to this approach is that it allows for an easy extension to higher point correlators. The 4-point correlator is defined in equation 4.8

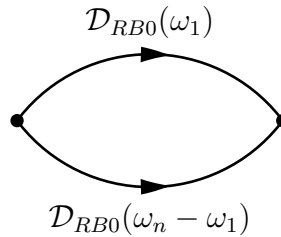
$$\mathcal{D}_{4R}(t + \tau, t) = - \left\langle T_\tau \left(\hat{A}(t + \tau) \hat{A}(t + \tau) \hat{A}(t) \hat{A}(t) \right) \right\rangle \quad (4.8)$$

This is related to the variance of the photons. This particular value is useful in discussing the statistics of light and finding the quantum nature of light [48]. The same technique as used before can be applied here. For the 0th order term there are two diagrams. The first is a bubble diagram



$$(4.9)$$

The second is a propagating diagram. This diagram has an extra factor of two which can be attributed to the number of different ways two lines can be connected between 2 points.

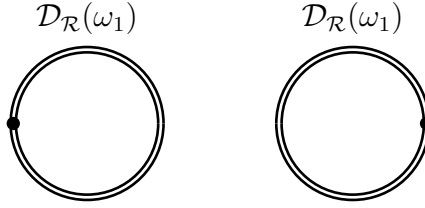


$$(4.10)$$

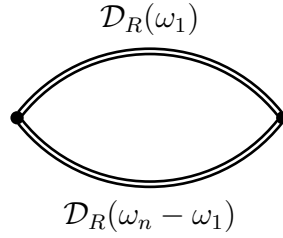
Each higher order diagram adds $\frac{n}{2}$ interaction lines and photon lines to the 0th order diagram. All of the propagating diagrams have a factor of 2. After expanding the

correlator out it can be shown that the non-interacting lines in the 0-order diagrams can be replaced by the fully interacting 2-point lines.

This can be represented diagrammatically as follows:



(4.11)



(4.12)

Or in equations:

$$\mathcal{D}_{4R}^{Bubbles} = \left(\sum_{\omega_1} \mathcal{D}_{2R}(\omega_1) \right)^2 \tag{4.13}$$

$$\mathcal{D}_{4R}^{Propagating} = 2 \sum_{\omega_1} \mathcal{D}_{2R}(\omega_1) \mathcal{D}_{2R}(\omega_n - \omega_1) \tag{4.14}$$

4.2.2 Higher order correlators

To better characterize the statistics of the system higher-order correlators are required [48]. The sum of higher-order correlators was examined in [56]. This was done within the RWA and in the time domain to find the characteristic function. By extending the arguments of the previous section the diagrams for the N-order photon correlator

can be determined by examining the unperturbed diagrams. If we consider a diagram with p propagating lines then there will be $p!$ ways if adding in a single interaction. If we consider adding two interactions but keeping them on the same line there is again $p!$ ways of doing this. Thus the arguments used above to show that the zero-order lines can be replaced with the full 2-point correlator lines will hold.

Drawing the 0th order diagrams for the N-operator correlator proves to be fairly simple. The topology of the 0th order diagrams follows a simple recursive rule (The Mastubara sums and frequencies have been suppressed at the moment for ease of notation):

$$\mathcal{D}^{(N)} \leftarrow B^2 \mathcal{D}^{(N-2)} + \mathcal{D}_{fullypropagating}^{(N)} \quad (4.15)$$

This is not an equality but rather just shows how you can produce the next level of diagrams. The counting factor for each diagram is not so simple. Equation 4.15 says the the N-order diagrams are drawn by looking at the (N-2)-order diagrams, multiplying by two bubbles and then adding in a diagram unique to the N-order diagram, namely the diagram in which all lines are propagating from the initial point to the final. For the 6-order diagrams you would start with the single propagating line. After multiplying by the two bubble terms you end up with a diagram that looks like a dumbbell. Finally the unique term for the 3-photon diagrams is three propagating lines. .

As mentioned previously the counting factor for each diagram does not have a neat recursive form. It can be found by considering how the lines can connect for a given topology. For any given N-order diagram there are n lines, p propagating lines, and b bubble lines. The number of bubble lines will always be even and $n = p + b$. The equation for the counting factor can be seen as follows. There are n operators for the initial time and n operators for the final time. For p propagating lines there are n choose p ways of picking the initial operator and n choose p ways of picking the final time. There are then $p!$ ways of permuting this. For the bubble terms each operator is paired with one of the same time. Since each paring takes up two possible operators of the same time there will be $(b - 1)!!$ ways of pairing for each time. Thus we arrive at the equation for the counting factor for and arbitrary N-order photon

correlator.

$$r_{n,p,b} = \binom{n}{p}^2 ((b-1)!)^2 p! \quad (4.16)$$

Within that line of thought it would be interesting to see if one could sum types of diagrams. For example, by grouping all the N-operator diagrams with p propagating lines you get a sum that looks like:

$$(r_{p,0} + r_{p,2}B^2 + \cdots + r_{p,b}B^b) \mathcal{D}_p \quad (4.17)$$

Unfortunately attempting to sum this is unsuccessful as the sum diverges. This isn't necessarily an issue since such a function is not guaranteed to exist.

4.3 Second Approach

Another approach is to ignore the inconsistency in the time-ordering and to treat give the qubit exchange factor a value of -1 like one would expect of anti-commuting operators. This creates an interesting situation where the photon correlator carries in a bosonic frequency which is discrete and even into a vertex and the qubit correlator carries out a fermionic frequency which is discrete and odd. Therefore there cannot be conservation of frequency as is typically demanded.

In the diagrammatic approach conservation of frequency is derived from taking the Fourier transform of the time-space correlators. Each vertex leads to terms like $\int_0^{\hbar\beta} e^{(i\omega_n - i\omega_m + i\omega_r)\tau} d\tau$ where the frequency terms are discrete and equal to either $\frac{2\pi n}{\hbar\beta}$ or $\frac{2\pi n+1}{\hbar\beta}$. Since the interaction term in the Hamiltonian is typically bosonic the frequencies in the exponential will be add in such a way that the overall term is even. If the overall term in the exponential is even then the integral is proportional to the Kronecker delta which ensures that the discrete frequency flowing into and out of the vertex is conserved.

In this case each vertex has a single boson frequency and a single fermion frequency flowing into or from it. Thus the term in the exponential cannot be even and frequency is not conserved. The integral is carried out and the Matsubara summation due to the Fourier transform still remain to be solved. Because of this approach there are two

changed. The first is that the qubit correlator differs from equation 4.4.

$$\mathcal{S}^0(i\omega_n) = \frac{2i\omega_n}{(i\omega_n)^2 - \omega_z^2} \quad (4.18)$$

The second difference is that equation 4.7 does not describe the full photon correlator, we have to include summations over the Matsubara frequencies for each vertex. Thus each higher order of the diagram has more and more summations. However, we the diagram solution in equation 4.6 can be used to get an answer which can be explicitly checked.

$$\mathcal{D}(i\omega_n) = \mathcal{D}^0(i\omega_n) + \frac{4\lambda^2}{\hbar^2\beta^2} \sum_{i\omega_1, i\omega_2} \frac{\mathcal{D}(i\omega_1)\mathcal{S}^0(i\omega_2)\mathcal{D}(i\omega_n)}{(i\omega_1 - i\omega_2)(i\omega_2 - i\omega_n)} \quad (4.19)$$

In the current form equation 4.19 is not easily solved. Since we do not know the poles of the full photon correlator *a priori* the only move forward is to do the second summation first. Thus we first consider just the terms with the second Matsubara frequency. It should be noted that the first frequency is a bosonic frequency and the second is a fermionic frequency.

$$\sum_{i\omega_2} \frac{\mathcal{S}^0(i\omega_2)}{(i\omega_1 - i\omega_2)(i\omega_2 - i\omega_n)} \quad (4.20)$$

Immediately we see an issue as the summand in equation 4.20 has poles along the imaginary axis. There are two ways to deal with this. The first is to add a small positive real part to the poles which then go to zero. The second is to deform the contour so that poles due to the bosonic frequency are just avoided. Since the fermionic frequency that we are summing over is always odd and the bosonic terms are always even they never overlap. However, this leads to another issues. Since we will have to carry out one more summation, which is over bosonic frequencies, after this one we want to ensure that there are no poles which are on the imaginary axis and are even. The best way to do this is to add a small real part to the frequency $i\omega_n$ which will go to zero at the end of the calculation.

Carrying out the summation in equation 4.20 and simplifying gives the following

result.

$$-\hbar\beta \left(\frac{n_F(\delta)\mathcal{S}^0(i\omega_n + \delta) - \frac{1}{2}\mathcal{S}^0(i\omega_1)}{i\omega_n + \delta - i\omega_1} + \frac{\mathcal{S}^0(i\omega_n + \delta)\mathcal{S}^0(i\omega_1)}{4(i\omega_n + \delta)i\omega_1} \right. \\ \left. \left((i\omega_n + \delta)i\omega_1 + \omega_z^2 - \frac{n_F(\omega_z)}{n_B(\omega_z)}(i\omega_n + \delta + i\omega_1)\omega_z \right) \right) \quad (4.21)$$

It turns out that the first term and the second term are identical once δ is taken to be zero. The effect of δ only changes the first term and only changes the result of the pole at $i\omega_n$. Therefore we can partially take care of the $i\omega_1$ summation and we can simplify our results.

$$\mathcal{D}(i\omega_n) = \mathcal{D}^0(i\omega_n) + \lambda^2 \mathcal{D}(i\omega_n) \mathcal{S}^0(i\omega_n) \mathcal{D}^0(i\omega_n) \\ + \frac{\lambda^2}{\hbar\beta} \left(\sum_{i\omega_1} \frac{n_F(\omega_z)}{n_B(\omega_z)} \frac{\mathcal{S}^0(i\omega_n)\mathcal{S}^0(i\omega_1)}{i\omega_n i\omega_1} \omega_z (i\omega_n + i\omega_1) \mathcal{D}(i\omega_n) \right) \quad (4.22)$$

If we subtract the second term, factor out \mathcal{D} on the left hand side and divide by the factor in front then we end up with a term very similar to equation 4.7. This is possible to solve but we first make some substitutions, $i\omega_n \rightarrow x, i\omega_1 \rightarrow y, \mathcal{D} \rightarrow \phi$. Additionally the term similar to equation 4.7 will be denoted $f(x)$, all the constants in the summation will be α and finally all the other terms in the sum will be grouped into a kernel, $K(x, y)$. With these substitutions our equations simplifies to:

$$\phi(x) = f(x) + \alpha \sum_y \phi(y) K(x, y) \quad (4.23)$$

This is reminiscent of a Fredholm equation of the second kind with a sum instead of an integral. In this case the kernel is separable so these can be recast as a set of linear equations with coefficients that are solvable integrals [57]. The solution is rather lengthy and closely follows the standard approach to solving the problem only with summations instead of integrals. However, those summations eventually are recast as integrals when solving them. This can also be extended for the case of a multi-mode cavity. However, the process is quite lengthy and not particularly enlightening.

At the end of this lengthy process we find that the results are not significantly different from our previous results. However, we believe this offers an interesting path forward. Since this particular approach ignores the consistency issue between the time evolution expansion and the time ordering definition it is likely insufficient. Another approach might be to allow the bosonic term to have a time ordering exchange factor of i and the spin to have one of $-i$. This requires a similar approach as was done above since the frequencies associated with these exchange factors are quarters and third quarters of $\pi/\hbar\beta$ and thus frequency will not be conserved in the vertices. That being stated, the likelihood of such an approach working seems low and will require quite a bit more work to flesh out.

4.4 Majorana Approach

Our last approach to circumvent the issue of the consistency between the expansion of the time evolution operator and the time ordering issue is to use a Majorana representation of spin. Majorana fermions are fermions which are their own anti-particle. They are expected to be useful in building fault-tolerance quantum computing [58]. Our interest here is to use the Majorana representation of spin operators to recast the Hamiltonian in a more desirable form. This embeds the spin operators in a higher dimensional Hilbert space. This approach has been used to find a solution in the Kitaev honeycomb model, which is a honeycomb lattice of spins and other condensed matter systems [59–61]. There are three main ways of representing spin with Majorana fermions, we choose the $SO(3)$ representation which involves casting the spin operators in terms of three Majorana fermions [59].

$$\sigma^x = -i\eta^y\eta^z \tag{4.24}$$

$$\sigma^y = -i\eta^z\eta^x \tag{4.25}$$

$$\sigma^z = -i\eta^x\eta^y \tag{4.26}$$

Where η^j are the Majorana operators which satisfy the Clifford algebra $\{\eta^i, \eta^j\} =$

$2\delta_{i,j}$. They are also their own antiparticle $\eta^{i\dagger} = \eta^i$. Because of this we can represent the Majorana particles as a sum of regular fermion operators [60].

$$\eta^x = f + f^\dagger \quad (4.27)$$

$$\eta^y = i(f - f^\dagger) \quad (4.28)$$

$$\eta^z = g + g^\dagger \quad (4.29)$$

We can transform the Hamiltonian now:

$$H = \hbar\omega_c a^\dagger a - \hbar\omega_z f^\dagger f + \hbar\lambda A(f^\dagger - f)(g^\dagger + g) \quad (4.30)$$

Or

$$H = \hbar\omega_c a^\dagger a + \hbar\omega_f f^\dagger f + \hbar\omega_g g^\dagger g + \hbar\lambda AFG \quad (4.31)$$

Where $\omega_f = -\omega_z$ and $G = g^\dagger + g$, $F = f^\dagger - f$. We let ω_g go to zero at the end. Its existence will allow us to use Wick's theorem. We now define our Green's functions in the typical way, \mathcal{F} for the f fermions and \mathcal{G} for the g fermions.

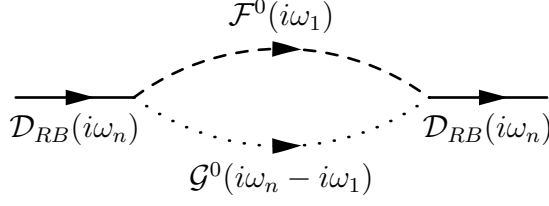
$$\mathcal{D}(i\omega_n) = \frac{2\omega_c}{(i\omega_n)^2 - \omega_c^2} \quad (4.32)$$

$$\mathcal{F}(i\omega_n) = \frac{-2i\omega_n}{(i\omega_n)^2 - \omega_f^2} \quad (4.33)$$

$$\mathcal{G}(i\omega_n) = \frac{2i\omega_n}{(i\omega_n)^2 - \omega_g^2} \quad (4.34)$$

4.4.1 Second Order

At this point normal perturbation theory can be carried out. The lowest order diagram is shown below.



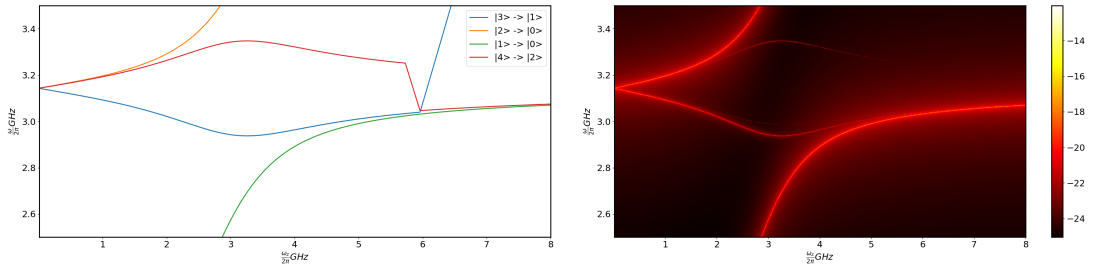
The self energy term is

$$\frac{\lambda^2}{\hbar\beta} \sum_p \mathcal{F}^0(i\omega_p) \mathcal{G}^0(i\omega_n - i\omega_p) \quad (4.35)$$

Which can be evaluated by the typical Matsubara technique of equating the sum to an integral and solving via the residue theorem. This function has poles at $\pm\omega_f$ and $i\omega_n \pm \omega_g$ so that the ω_g limit must be taken at the end of the calculation. This gives a self energy of

$$\frac{2\omega_z}{(i\omega_n)^2 - \omega_z^2} \frac{n_{FD}(\omega_z)}{n_{BE}(\omega_z)} \quad (4.36)$$

For comparison we use a similar approach as before. We numerically diagonalize the Hamiltonian for the case of 16 photons and then compare. We pick values comparable to experiments. The results of this can be seen in figure 4.5. The main lines we will focus on are the transitions from the two lowest states to the ground state and the two level transitions from the third and fourth state. The results are given in terms of the spectral density $\rho = | -2\text{Imag}(\mathcal{D}^{ret}(\omega)) |$.



(a) Relevant eigenvalue transitions

(b) Log of the spectral function calculated by numerically finding the eigenvalues

Figure 4.5: The numerical results for deep strong coupling. The values are $\frac{\omega_c}{2\pi} = 3.142\text{GHz}$, $\frac{\lambda}{2\pi} = 0.5\text{GHz}$. The temperature is at 20mK.

When compared to the exact diagonalization results this provides an excellent match to the main line (figure 4.6).

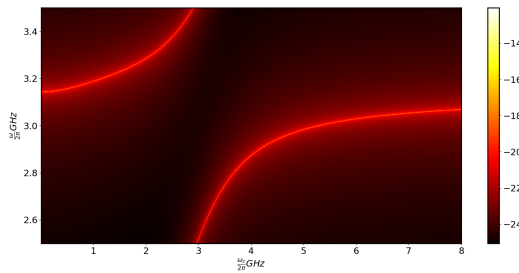
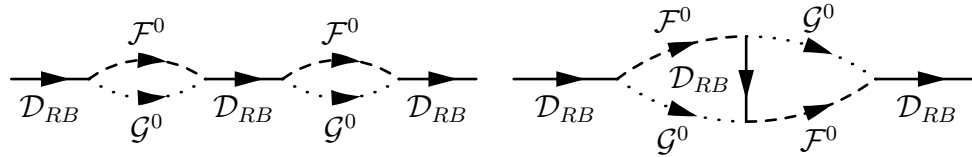


Figure 4.6: Log of the spectral function calculated with the second order diagrams, parameters are the same as in figure 4.5.

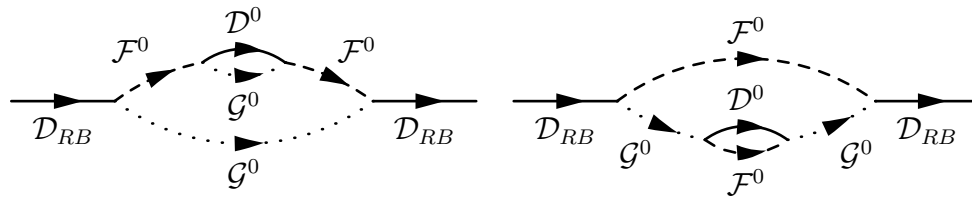
This is the most promising approach so far as the lack of agreement has a simple and natural explanation. In this case we are only dealing with the lowest order diagrams. Including more diagrams is a natural way to increase the accuracy. Additionally higher order diagrams have the chance to take into account the higher excitation transitions that we saw in the numerical results. Thus a natural next step is to examine the higher order diagrams produced by this result.

4.4.2 Fourth Order



(a) This is the only non-proper diagram. It is simply a repetition of the single second order diagram.

(b) This is an effective vertex diagram. It can be redrawn as the second order diagram with a frequency dependent vertex.



(c) This is an interaction diagram. The top interaction is changed by the smaller diagram within it.

(d) This is also an interaction diagram. It is similar to the above one only effecting the bottom branch instead of the top.

Figure 4.7: The four fourth order diagrams each has a counting factor of 1.

For fourth order diagrams there are four new diagrams, three of them are proper diagrams. The three proper diagrams are made up of two that create an effective interaction and one that makes up a vertex correction. Including these diagrams into the Dyson equation gives us a higher order approximation. There are two issues that immediately arise. The first is similar to the issue we saw in the previous section regarding analytical continuation.

Namely the spectral function dips below zero for certain values. We sidestep this in the same way by looking at the absolute value of the spectral function. The second issue is that the results seen in figure 4.8 are less of a match to the numerical results. This could require a more sophisticated approximation scheme. We first proceed under that assumption and examine an effective interaction approximation.

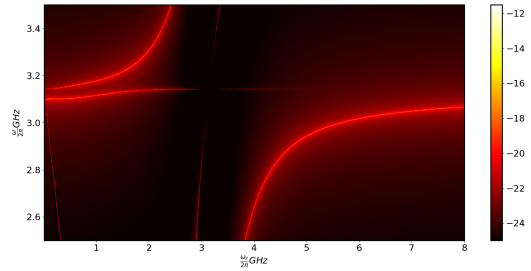


Figure 4.8: Log of the spectral function with the fourth order diagrams, parameters are the same as in figure 4.5.

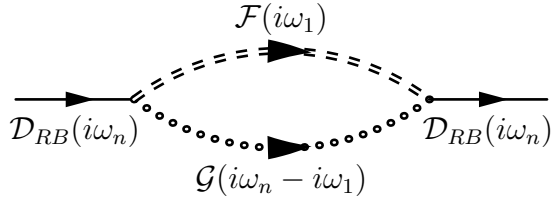
4.4.3 Effective Interaction

Another possible approximation is the effective interaction [31]. In the interaction diagrams seen above there is a loop within the interaction lines. At each higher order there will be an additional loop. The key to realizing this approximation is to understand that this can be summed in a similar way as the self energy terms. Therefore we can consider an interaction line which is the sum of all effective interaction diagrams in the form of figure 4.7:

$$\begin{aligned}
 \text{=====} \blacktriangleright \text{=====} &= \text{-----} \blacktriangleright \text{-----} + \text{-----} \blacktriangleright \text{-----} \text{---} \text{---} \text{---} \blacktriangleright \text{-----} + \text{-----} \blacktriangleright \text{-----} \text{---} \text{---} \text{---} \text{---} \text{---} \blacktriangleright \text{-----} + \dots \\
 & \hspace{20em} (4.37)
 \end{aligned}$$

Equation 4.37 shows how this is done with one interaction line, the same applies to

the other. This can be summed exactly as in previous sections where the self energy is summed to infinite order. This gives the following diagram which is the same as the second order diagram only with effective interactions.



Solving the Matsubara sums for this type of diagram follows the exact same process as before. The only difference is that the poles of the interaction lines are not trivial to calculate. The poles of these functions are the roots of a polynomial function. The \mathcal{F} line has roots which can be solved by hand but the \mathcal{G} line must be solved numerically for each relevant value. Once those values are in hand then the Matsubara sum can be carried out. The primary concern in this approach is if there are non-simple poles. A simple check during the numerical calculations shows that this is not the case.

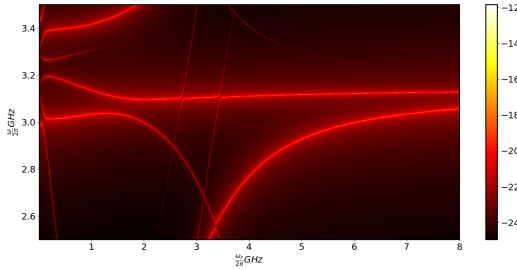
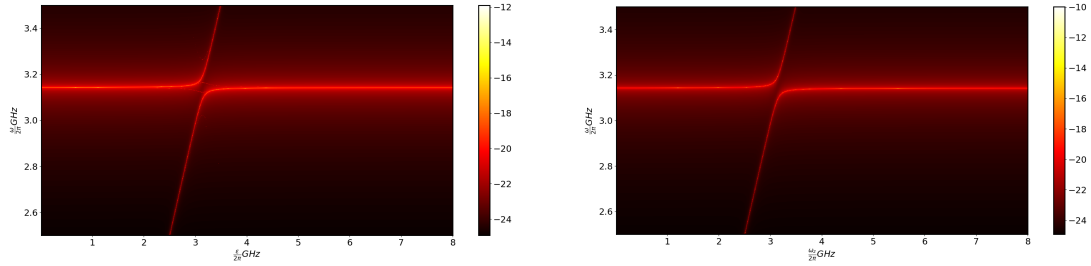


Figure 4.9: Log of the spectral function in the effective interaction approximation, the parameters are the same as in figure 4.9.

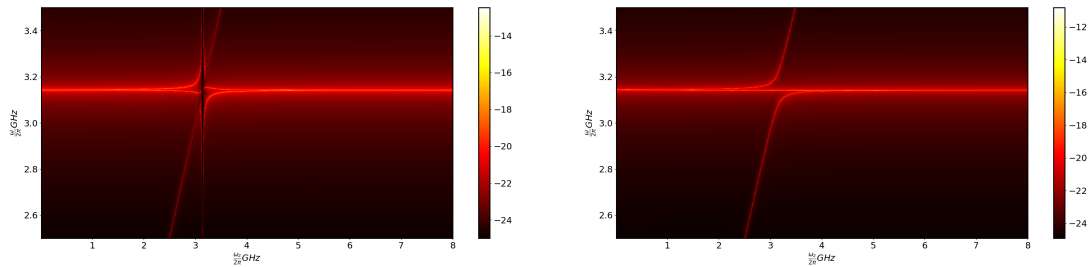
After going through this process we can see the results in figure 4.9. We notice that our results have gotten worse indicating some fundamental issue with our approach. A possible approach would be to continue to search for possible approximations that give good results. For example the vertex diagram could be expanded into a ladder-like approximation in a similar way as was done with the effective interaction. However, in an attempt to shed some insight onto this matter we examine these different

approximations in a lower coupling. The results of this are seen in figure 4.10. At lower coupling the second order diagrams still give a good approximation to the main spectral lines. This time however, the higher order approximations maintain good

agreement with the main spectral lines and the higher order transitions are closer to what is seen in the numerical results. This seems to indicate that the problem may lay with the perturbation approach itself. There is one possible remedy to this which we will explore in the next section.



(a) Spectral function derived from numerical diagonalization (b) Spectral function derived using the second order diagrams



(c) Spectral function derived using the fourth order diagrams (d) Spectral function derived within the effective interaction approximation

Figure 4.10: The log of the spectral function is shown using numerical diagonalization in (a), second order diagrams in (b), fourth order diagrams in (c), and the effective interaction approximation in (d). The parameters are the same as before with the exception of the coupling strength. Here the coupling is 10 times smaller $\frac{\lambda}{2\pi} = 0.05GHz$.

4.4.4 Self Consistent Solution

Finally we examine the possibility of solving the correlators via a self consistent solution. For a self consistent solution we proceed to find the diagrams as normal. Once we have choose the set of diagrams we wish to include in the Dyson equation we replace all of the internal unperturbed correlators with full correlators. This then

is solved self consistently by guessing a solution, calculating the result, and then plugging the solution back into the original equation again. This proceeds until the correlator converges. The reasoning for using a self consistent approach is as follows. As the coupling increases the counter rotating contributions become more important. But in any non-self consistent approach the counter rotating terms are not fully accounted for because we are expanding in terms of the unperturbed correlators. In the unperturbed correlators terms like $a^\dagger a^\dagger$ are zero. This is similar to the case found when a system under goes a phases change [32]. The diagrams and equations for the lowest order are below.

$$(4.38)$$

$$(4.39)$$

$$(4.40)$$

$$\mathcal{D}(i\omega_n) = \frac{\mathcal{D}^0(i\omega_n)}{1 - \frac{\lambda^2}{\hbar\beta} \sum_p \mathcal{F}(i\omega_p) \mathcal{G}(i\omega_n - i\omega_p)} \quad (4.41)$$

$$\mathcal{F}(i\omega_n) = \frac{\mathcal{F}^0(i\omega_n)}{1 - \frac{\lambda^2}{\hbar\beta} \sum_p \mathcal{G}(i\omega_p) \mathcal{D}(i\omega_n - i\omega_p)} \quad (4.42)$$

$$\mathcal{G}(i\omega_n) = \frac{\mathcal{G}^0(i\omega_n)}{1 - \frac{\lambda^2}{\hbar\beta} \sum_p \mathcal{F}(i\omega_p) \mathcal{D}(i\omega_n - i\omega_p)} \quad (4.43)$$

The issue with a self consistent approach is that it must be done numerically. This poses a problem since the analytical continuation will then have to be done numerically as well. We attempted solving this by assuming correlator can be expanded by the Mittag-Leffler expansion hoping we could get around the issue of numerical analytical continuation. The assumptions are that the function is analytic at $\omega = 0$ and that all the poles are discrete and simple.

$$\mathcal{D}(\omega) = \mathcal{D}(0) + \sum_i b_i \left(\frac{1}{\omega - z_i} - \frac{1}{z_i} \right) \quad (4.44)$$

Where z_i, b_i are the poles and residues of the function. This allows one to find the residue and roots of the Matsubara Green's function and then analytically continue the function using the Mittag-Leffler expansion and adding a small imaginary part to the pole. However, this approach did not converge to an appropriate solution. Thus the next step is to numerically solve the Matsubara correlators directly and then use a numerical analytical continuation technique. The specifics of this are outside the scope of this work but we will briefly outline one approach though there are other methods as well [62–65].

The most promising approach for our situation is discussed in a paper by Han [66]. The idea behind Padé analytic continuation is to fit the numerical data calculated from the Matsubara correlator in imaginary frequency space to a rational function of degree r .

$$\frac{\sum_{n=0}^{r-1} p_n z^n}{\sum_{n=0}^r q_n z^n} \quad (4.45)$$

Once the parameters p_n and q_n are found the fitted function can be analytically continued by taking z to $\omega + i\delta$. The value of r must be large enough that it covers all the relevant poles of the function. However, if it is too large the fitting procedure can introduce errors. Additionally Padé analytic continuation suffers from sensitivity to numerical noise. Han *et. al.* examined the influence of decimal precision on numerically analytically continuing a correlator compared to the known result. They found that in order to achieve a good quality match with the known value the Padé approach required 82 decimal places of precision. This is generally impractical to

achieve. To combat this, they found that by expanding the distribution functions into a Padé representation and then using those frequencies instead of the Matsubara frequencies they could get much better results with less computational time. They called this technique the Padé decomposition [66]. This approach should be able to be applied to our situation in later work.

4.5 Conclusions

We examined different approaches for calculating the spectrum of a cavity-qubit system in the ultra strong and deep strong coupling regime. In doing so we found that naive approaches to using Feynman diagrams and perturbation theory led to good agreement with the primary spectral lines found in experiments but lacked the power to explain the higher excitation transitions. This stems from a failure of the theory to account for Hamiltonians with fermi-like interactions.

We developed a technique to recast the standard cavity-qubit Hamiltonian in terms of Majorana fermions by using a Majorana spin representation. This embeds the system in a higher dimensional Hilbert space giving a more complicated form to the Hamiltonian. The positive effect of this is to alter the interaction term such that it is boson-like and thus the standard Feynman perturbation theory can be safely applied. In doing so the various techniques used in diagrammatic approaches can be applied and different infinite order approximations can be found.

Using this Majorana representation we found that our results still didn't match with numerical calculations. By examining the effect of the coupling on this difference we concluded that the most reasonable course of action would be to apply a self-consistent approach to ensure that the counter-rotating terms that increase in importance with coupling are properly accounted for. This requires numerically finding the Matsubara correlators and then numerically analytically continuing those correlators to find the spectral function. We discussed a possible approach to this method using the Padé decomposition method.

This work provides a new perspective on a fundamental system in quantum information and cavity QED. The full influence of this perspective is yet to be seen

but has the potential to provide results to higher coupling regimes. Additionally this work provides a pathway to other such systems in cavity QED where the coupling operator does not follow a boson-like commutation rule.

Chapter 5

Conclusion

In this work we examined the reliability of detecting the Transverse Field Ising Model coupled to a cavity detector. We used a diagrammatic expansion technique to examine when the simulator's correlator can be reliably retrieved by the detector. We saw that because of the simulator's quadratic coupling operator perfectly reliable detection could not be achieved. In Tian's previous work we saw that when the simulator's coupling operator is linear then Wick's theorem applies to the operator itself and perfectly reliable detection can be achieved. The quadratic coupling creates diagrams with structures such that only a perturbed correlator can be retrieved.

By examining the self energy of the total system some of the detector's effects can be mitigated. Additionally we examined the parameters and conditions under which approximately reliable detection can be achieved. We found that a large cavity frequency, low coupling strength, small simulator size, and high detector bandwidth are beneficial for ensuring good detection. The effect of these is to reduce the shift on the simulator's measured eigenvalues as well as preventing overlap between peaks in the spectra that are due to the simulator and those that are due to the detector-simulator interaction.

We believe that the concepts elucidated in this work are useful in the design and analysis of quantum simulators beyond that of our model. Further work can be done to build on these results by examining additional models as well as exploring potential experimental mitigation of the eigenvalue shift from the detector. As quantum

simulators grow in size and importance the effects of the detector will become more prominent and thus must be taken into account.

We also studied the spectra of the well known Rabi model, a cavity coupled to a qubit. We used a diagrammatic technique to find the spectral function in different approximations. Because the technique allows for infinite order perturbation theory we hoped to accurately calculate the transmission spectra for ultra-strong and deep-strong coupling. This was done by calculating the Matsubara correlator. While our results are not in agreement with experiments we found that by recasting the spin operators in terms of Majorana operators we could apply the typically infinite order perturbation theory which opens up a plethora of known theoretical techniques.

Further work can be done in examining the various possible approximations as well as extending our results to allow for self consistent solutions. In cases where the Matsubara correlator cannot be solved analytically the Padé decomposition may allow for numerically analytically continuing the correlator to find the spectral function. This work can be built on further by applying the same transformations to the situation where there is an additional external magnetic coupling in the qubit.

Appendix A

Wicks Theorem for Spin

To start the proof of Wick's theorem for spin operators we start off with a Hamiltonian of the form of eqn (A.1). The unperturbed part of the Hamiltonian is quadratic in the raising and lowering operators. The proof follows closely the proof in [29].

$$H = H_0 + H_I \tag{A.1}$$

$$H_0 = \frac{\hbar\omega_x}{2}\sigma^z \tag{A.2}$$

$$H_I = \lambda A\sigma^x \tag{A.3}$$

We first define the contraction to be as follows:

$$\overline{AB} = \langle T_\tau[AB] \rangle_0 \tag{A.4}$$

Let A and B be spin operators then the contraction is the Spin Green's function $\mathcal{S}^0(\tau, \tau') = \langle T[\sigma^x(\tau)\sigma^x(\tau')] \rangle_0$ Where the 0 subscript just refers to the use of the unperturbed Hamiltonian. The claim we want to prove is that: $\langle T_\tau[ABC\dots Z] \rangle_0 =$ sum of all full contracted permutations:

Since changing the time order on the left brings about a total sign change on the right as well we can consider just the term: $\langle ABC\dots Z \rangle_0$ so long as $\tau_A > \tau_B\dots$ and drop the time ordering term.

Rewriting the field operators in terms of raising and lowering operators

$$A = \sum_j \chi_j(\tau) \sigma_j^{\alpha_a} \quad (\text{A.5})$$

Where $\sigma_j^{\alpha_a}$ is either a raising or lowering operator depending on α_a . The term $\langle ABC\dots Z \rangle_0$ can then be rewritten as

$$\langle ABC\dots Z \rangle_0 = \sum_a \sum_b \dots \sum_z \chi_a \chi_b \dots \chi_z \text{Tr}[\rho_0 \sigma_a^{\alpha_a} \sigma_b^{\alpha_b} \dots \sigma_z^{\alpha_z}] \quad (\text{A.6})$$

We are assuming that the unperturbed states are such that an equal number of raising and lower operators are needed to be non-zero. Commuting the first term over to the right many times:

$$\begin{aligned} \text{Tr}[\rho_0 \sigma_a^{\alpha_a} \sigma_b^{\alpha_b} \dots \sigma_z^{\alpha_z}] &= \text{Tr}[\rho_0 [\sigma_a^{\alpha_a}, \sigma_b^{\alpha_b}]_+ \sigma_c^{\alpha_c} \dots \sigma_z^{\alpha_z}] - \\ \text{Tr}[\rho_0 \sigma_a^{\alpha_a} [\sigma_b^{\alpha_b}, \sigma_c^{\alpha_c}]_+ \dots \sigma_z^{\alpha_z}] \dots &- \text{Tr}[\rho_0 \sigma_b^{\alpha_b} \sigma_c^{\alpha_c} \dots \sigma_z^{\alpha_z} \sigma_a^{\alpha_a}] \end{aligned} \quad (\text{A.7})$$

$$\begin{aligned} \text{Tr}[\rho_0 \sigma_a^{\alpha_a} \sigma_b^{\alpha_b} \dots \sigma_z^{\alpha_z}] &= [\sigma_a^{\alpha_a}, \sigma_b^{\alpha_b}]_+ \text{Tr}[\rho_0 \sigma_c^{\alpha_c} \dots \sigma_z^{\alpha_z}] - \\ [\sigma_b^{\alpha_b}, \sigma_c^{\alpha_c}]_+ \text{Tr}[\rho_0 \sigma_a^{\alpha_a} \dots \sigma_z^{\alpha_z}] \dots &- \text{Tr}[\rho_0 \sigma_b^{\alpha_b} \sigma_c^{\alpha_c} \dots \sigma_z^{\alpha_z} \sigma_a^{\alpha_a}] \end{aligned} \quad (\text{A.8})$$

The last term can be rewritten using the cyclic properties of the trace:

$$\text{Tr}[\rho_0 \sigma_b^{\alpha_b} \sigma_c^{\alpha_c} \dots \sigma_z^{\alpha_z} \sigma_a^{\alpha_a}] = \text{Tr}[\sigma_a \rho_0 \sigma_b^{\alpha_b} \sigma_c^{\alpha_c} \dots \sigma_z^{\alpha_z}] \quad (\text{A.9})$$

Finally using the Baker-Hausdorff theorem we can show that:

$$\sigma_a^{\alpha_a} \rho_0 = \rho_0 \sigma_a^{\alpha_a} e^{\alpha_a \beta \omega_z} \quad (\text{A.10})$$

$$\sigma^\alpha \rho_0 = \sigma^\alpha e^{-\beta H_0} = e^{-\beta H_0} e^{\beta H_0} \sigma^\alpha e^{-\beta H_0} \quad (\text{A.11})$$

Defining $\sigma^\alpha(\beta) = e^{\beta H_0} \sigma^\alpha e^{-\beta H_0}$ we can find easily take the derivative of the function with respect to β .

$$\dot{\sigma}^{\alpha(\beta)} = e^{\beta H_0} [H_0, \sigma^\alpha] e^{-\beta H_0} = \alpha \hbar \omega_z \sigma^\alpha(\beta) \quad (\text{A.12})$$

Gives

$$\sigma^\alpha \rho_0 = \rho_0 \sigma^\alpha \quad (\text{A.13})$$

Where α_a is equal to 1(-1) for the raising (lowering) operator. Plugging this in, pulling the last term over to the left and dividing we are left with:

$$\begin{aligned} Tr[\rho_0 \sigma_a^{\alpha_a} \sigma_b^{\alpha_b} \dots \sigma_z^{\alpha_z}] &= \frac{1}{1 + e^{\alpha_a \beta \omega_z}} [\sigma_a^{\alpha_a}, \sigma_b^{\alpha_b}]_+ Tr[\rho_0 \sigma_c^{\alpha_c} \dots \sigma_z^{\alpha_z}] + \\ &\frac{1}{1 + e^{\alpha_a \beta \omega_z}} [\sigma_b^{\alpha_b}, \sigma_c^{\alpha_c}]_+ Tr[\rho_0 \sigma_a^{\alpha_a} \dots \sigma_z^{\alpha_z}] \dots \end{aligned} \quad (\text{A.14})$$

The coefficients of the form $\frac{[\sigma_a^{\alpha_a} \sigma_b^{\alpha_b}]_+}{1 + e^{\alpha_a \beta \omega_z}}$ are equal to $\langle T_\tau[\sigma_a^{\alpha_a} \sigma_b^{\alpha_b}] \rangle$. So that each term is a contraction. Combining equations A.6 and A.14, repeating for each term in the brackets, and then placing the time ordering back in proves Wick's theorem.

Appendix B

TFIM Transforms

Starting with the Hamiltonian:

$$- \hbar J \sum_n^N \sigma_n^z \sigma_{n+1}^z - \hbar \frac{h_x}{2} \sum_n^N \sigma_n^x \quad (\text{B.1})$$

We first recast the spin operators as fermions to correct commutations relations on differing sites. Currently $\{\sigma_i^-, \sigma_i^+\} = 1$ and $[\sigma_i^-, \sigma_j^+] = 0$ for different i and j . To fix this we add in a phase to the operators.

$$\sigma_n^+ \rightarrow c_n^\dagger e^{i\pi \sum_j^{n-1} c_j^\dagger c_j} \quad (\text{B.2})$$

$$\sigma_n^- \rightarrow c_n e^{-i\pi \sum_j^{n-1} c_j^\dagger c_j} \quad (\text{B.3})$$

$$\sigma_n^z = 2\sigma_n^+ \sigma_n^- - 1 \rightarrow 2c_n^\dagger c_n - 1 \quad (\text{B.4})$$

$$\sigma_n^x = \sigma_n^+ + \sigma_n^- \rightarrow c_n^\dagger e^{i\pi \sum_j^{n-1} c_j^\dagger c_j} + c_n e^{-i\pi \sum_j^{n-1} c_j^\dagger c_j} \quad (\text{B.5})$$

It would be nice if the first term in the Hamiltonian involved the x-components instead of the z components because then all of the phases might cancel. So we rotate the entire system $\frac{\pi}{2}$ around the y-axis which takes $\sigma_n^z \rightarrow \sigma_n^x$ and $\sigma_n^x \rightarrow -\sigma_n^z$. Thus our new Hamiltonian:

$$\begin{aligned}
& -\hbar J \sum_n^N \left(c_n^\dagger e^{i\pi \sum_j^{n-1} c_j^\dagger c_j} + c_n e^{-i\pi \sum_j^{n-1} c_j^\dagger c_j} \right) \left(c_{n+1}^\dagger e^{i\pi \sum_j^n c_j^\dagger c_j} + c_{n+1} e^{-i\pi \sum_j^n c_j^\dagger c_j} \right) \\
& -\hbar \frac{h_x}{2} \sum_n^N (1 - 2c_n^\dagger c_n)
\end{aligned} \tag{B.6}$$

It is important to notice that the first terms commute with the creation/destruction operators in the second term because they are necessarily not included within the summation. However, if periodic boundary conditions are used then there is an extra term due to the c_{N+1}, c_{N+1}^\dagger terms not anticommuting with the c_1, c_1^\dagger terms.

$$\sum_n^N c_n^\dagger e^{i\pi c_n^\dagger c_n} c_{n+1}^\dagger + c_n e^{i\pi c_n^\dagger c_n} c_{n+1}^\dagger + c_n^\dagger e^{-i\pi c_n^\dagger c_n} c_{n+1} + c_n e^{-i\pi c_n^\dagger c_n} c_{n+1} + PT \tag{B.7}$$

With Periodic Terms (PT):

$$PT = -2\hbar J (c_N^\dagger e^{i\pi c_N^\dagger c_N} c_1^\dagger + c_N e^{i\pi c_N^\dagger c_N} c_1^\dagger + c_N^\dagger e^{i\pi c_N^\dagger c_N} c_1 + c_N e^{i\pi c_N^\dagger c_N} c_1) \tag{B.8}$$

Since $c_n^\dagger c_n$ is either 0 or 1 the exponentials are equal to $1 - 2c_n^\dagger c_n$.

$$\sum_n^N c_n^\dagger (1 - 2c_n^\dagger c_n) c_{n+1}^\dagger + c_n (1 - 2c_n^\dagger c_n) c_{n+1}^\dagger + c_n^\dagger (1 - 2c_n^\dagger c_n) c_{n+1} + c_n (1 - 2c_n^\dagger c_n) c_{n+1} + PT \tag{B.9}$$

Realizing that $c_n^\dagger c_n^\dagger$ and $c_n c_n$ are zero the first and third terms can be reduced to $c_n^\dagger c_{n+1}^\dagger$ and $c_n^\dagger c_{n+1}$. The second and fourth term reduce to $c_{n+1}^\dagger c_n$ and $c_{n+1} c_n$ which can be seen by using the anti-commutation rules $\{c_n, c_n^\dagger\} = 1$. We neglect the periodic terms so the Hamiltonian is:

$$H = -\hbar J \sum_n^N (c_n^\dagger c_{n+1}^\dagger + c_n^\dagger c_{n+1}^\dagger + c_{n+1}^\dagger c_n + c_{n+1} c_n) - \hbar \frac{h_x}{2} \sum_n^N (1 - 2c_n^\dagger c_n) \tag{B.10}$$

$$H = -\hbar J \sum_n^N (c_n^\dagger c_{n+1} + c_n^\dagger c_{n+1}^\dagger + h.c. - \hbar \frac{h_x}{2} \sum_n^N (1 - 2c_n^\dagger c_n)) \quad (\text{B.11})$$

The periodic term similarly reduces.

$$PT = -2\hbar J (c_N^\dagger c_1^\dagger + c_N^\dagger c_1 + h.c.) \quad (\text{B.12})$$

In order to simplify the subscripts it is useful move into k -space using the transform $c_n = \frac{1}{\sqrt{N}} \sum_k c_k e^{ikn}$ where $k = 2\pi m_k/N$, $m_k = -N/2 + 1 \dots N/2$. Writing out the important terms:

$$\sum_n c_n^\dagger c_n = \frac{1}{N} \sum_{nkk'} c_k^\dagger c_{k'} e^{-i(k-k')n} = \sum_k c_k^\dagger c_k \quad (\text{B.13})$$

$$\sum_n c_n^\dagger c_{n+1} = \frac{1}{N} \sum_{nkk'} c_k^\dagger c_{k'} e^{-i(k-k')n} e^{ik'} = \sum_k c_k^\dagger c_k e^{ik} \quad (\text{B.14})$$

$$\sum_n c_n^\dagger c_{n+1}^\dagger = \frac{1}{N} \sum_{nkk'} c_k^\dagger c_{k'}^\dagger e^{-i(k+k')n} e^{-ik'} = \sum_k c_k^\dagger c_{-k}^\dagger e^{ik} \quad (\text{B.15})$$

$$\sum_n c_{n+1}^\dagger c_n = \sum_k c_k^\dagger c_k e^{-ik} \quad (\text{B.16})$$

$$\sum_n c_{n+1} c_n = \sum_k c_{-k} c_k e^{-ik} \quad (\text{B.17})$$

Putting all of this into the Hamiltonian:

$$H = -\hbar J \sum_k (c_k^\dagger c_k e^{ik} + c_k^\dagger c_{-k}^\dagger e^{ik} + c_k^\dagger c_k e^{-ik} + c_{-k} c_k e^{-ik}) + \hbar h_x \sum_k c_k^\dagger c_k - \frac{\hbar h_x}{2} N \quad (\text{B.18})$$

$$H = \sum_k \hbar (h_x - 2J \cos(k)) c_k^\dagger c_k - \hbar J c_k^\dagger c_{-k}^\dagger e^{ik} - \hbar J c_{-k} c_k e^{-ik} \quad (\text{B.19})$$

Where the constant term was dropped. Now one more transformation is required to complete this analysis. Currently the Hamiltonian does not conserve particle number. To remedy this we make a Bogoliubov transformation $c_k = u_k \gamma_k + i v_k \gamma_{-k}^\dagger$. Working

out each term:

$$c_k^\dagger c_k = |u_k|^2 \gamma_k^\dagger \gamma_k + |v_k|^2 \gamma_{-k}^\dagger \gamma_{-k} + i(u_k^\dagger v_k \gamma_k^\dagger \gamma_{-k}^\dagger - v_k^\dagger u_k \gamma_{-k}^\dagger \gamma_k) \quad (\text{B.20})$$

$$c_k^\dagger c_{-k}^\dagger = u_k^\dagger u_{-k}^\dagger \gamma_k^\dagger \gamma_{-k}^\dagger - v_k^\dagger v_{-k}^\dagger \gamma_{-k}^\dagger \gamma_k^\dagger - i(u_k^\dagger v_{-k}^\dagger \gamma_k^\dagger \gamma_k + v_k^\dagger u_{-k}^\dagger \gamma_{-k}^\dagger \gamma_{-k}) \quad (\text{B.21})$$

$$c_{-k} c_k = u_{-k} u_k \gamma_{-k} \gamma_k - v_{-k} v_k \gamma_k^\dagger \gamma_{-k}^\dagger + i(u_{-k} v_k \gamma_{-k} \gamma_{-k}^\dagger + v_{-k} u_k \gamma_k^\dagger \gamma_k) \quad (\text{B.22})$$

Therefore the Hamiltonian is:

$$H = \sum_k \hbar(h_x - 2J \cos(k)) (|u_k|^2 \gamma_k^\dagger \gamma_k + |v_k|^2 \gamma_{-k}^\dagger \gamma_{-k}^\dagger + i(u_k^\dagger v_k \gamma_k^\dagger \gamma_{-k}^\dagger - v_k^\dagger u_k \gamma_{-k}^\dagger \gamma_k)) \quad (\text{B.23})$$

$$- \hbar J (u_k^\dagger u_{-k}^\dagger \gamma_k^\dagger \gamma_{-k}^\dagger - v_k^\dagger v_{-k}^\dagger \gamma_{-k}^\dagger \gamma_k^\dagger - i(u_k^\dagger v_{-k}^\dagger \gamma_k^\dagger \gamma_k + v_k^\dagger u_{-k}^\dagger \gamma_{-k}^\dagger \gamma_{-k})) e^{ik} \quad (\text{B.24})$$

$$- \hbar J (u_{-k} u_k \gamma_{-k} \gamma_k - v_{-k} v_k \gamma_k^\dagger \gamma_{-k}^\dagger + i(u_{-k} v_k \gamma_{-k} \gamma_{-k}^\dagger + v_{-k} u_k \gamma_k^\dagger \gamma_k)) e^{-ik} \quad (\text{B.25})$$

Rewriting the above by grouping the γ terms:

$$\begin{aligned} H = & \sum_k \gamma_k^\dagger \gamma_k \left(\hbar(h_x - 2J \cos(k)) |u_k|^2 + i\hbar J (u_k^\dagger v_{-k}^\dagger e^{ik} - v_{-k} u_k e^{-ik}) \right) + \\ & \gamma_{-k} \gamma_{-k}^\dagger \left(\hbar(h_x - 2J \cos(k)) |v_k|^2 + i\hbar J (v_k^\dagger u_{-k}^\dagger e^{ik} - u_{-k} v_k e^{-ik}) \right) + \\ & \gamma_k^\dagger \gamma_{-k}^\dagger \left(i\hbar(h_x - 2J \cos(k)) u_k^\dagger v_k + \hbar J (v_{-k} v_k e^{-ik} - u_k^\dagger u_{-k}^\dagger e^{ik}) \right) + \\ & \gamma_{-k} \gamma_k \left(-i\hbar(h_x - 2J \cos(k)) v_k^\dagger u_k + \hbar J (v_k^\dagger v_{-k}^\dagger e^{-ik} - u_{-k} u_k e^{-ik}) \right) \end{aligned} \quad (\text{B.26})$$

To determine the values of the coefficients first we examine the commutations relations and require the new operators to obey the same anticommutation relations. Additionally to preserve particle number we require the $\gamma^\dagger \gamma^\dagger$ and $\gamma \gamma$ terms to disappear. From the commutations relations:

$$\{c_k, c_k^\dagger\} = 1 = \{u_k \gamma_k + i v_k \gamma_{-k}^\dagger, u_k^\dagger \gamma_k^\dagger + i v_k^\dagger \gamma_{-k}\} \quad (\text{B.27})$$

$$= \{u_k \gamma_k, u_k^\dagger \gamma_k^\dagger + i v_k^\dagger \gamma_{-k}\} + \{i v_k \gamma_{-k}^\dagger, u_k^\dagger \gamma_k^\dagger + i v_k^\dagger \gamma_{-k}\} \quad (\text{B.28})$$

$$= \{u_k \gamma_k, u_k^\dagger \gamma_k^\dagger\} + \{u_k \gamma_k, i v_k^\dagger \gamma_{-k}\} + \{i v_k \gamma_{-k}^\dagger, u_k^\dagger \gamma_k^\dagger\} + \{i v_k \gamma_{-k}^\dagger, i v_k^\dagger \gamma_{-k}\} \quad (\text{B.29})$$

$$= |u_k|^2 + |v_k|^2 = 1 \quad (\text{B.30})$$

We can parameterize the coefficients as $u_k = \cos(\theta_k)$; $v_k = \sin(\theta_k)$. Examining the $\gamma^\dagger \gamma^\dagger$ terms and requiring that it equals zero:

$$i\hbar(h_x - 2J \cos(k)) \cos(\theta_k) \sin(\theta_k) + \hbar J (\sin(\theta_{-k}) \sin(\theta_k) e^{-ik} - \cos(\theta_k) \cos(\theta_{-k}) e^{ik}) = 0 \quad (\text{B.31})$$

$$\frac{i(h_x - 2J \cos(k))}{J} = \frac{\cos(\theta_k) \cos(\theta_{-k}) e^{ik} - \sin(\theta_{-k}) \sin(\theta_k) e^{-ik}}{\cos(\theta_k) \sin(\theta_k)} \quad (\text{B.32})$$

Taking just the imaginary parts:

$$\frac{(h_x - 2J \cos(k))}{J \sin(k)} = \frac{\cos(\theta_k) \cos(\theta_{-k}) + \sin(\theta_{-k}) \sin(\theta_k)}{\cos(\theta_k) \sin(\theta_k)} \quad (\text{B.33})$$

Examining the case where k goes to minus k we see the left hand side becomes negative. In order for the right hand side to match the bottom term must become negative. This occurs when $\theta_k = -\theta_{-k}$ Thus:

$$\frac{(h_x - 2J \cos(k))}{J \sin(k)} = \frac{\cos^2(\theta_k) - \sin^2(\theta_k)}{\cos(\theta_k) \sin(\theta_k)} \quad (\text{B.34})$$

Using the basic trig identities:

$$\frac{(h_x - 2J \cos(k))}{J \sin(k)} = \frac{2 \cos(2\theta_k)}{\sin(2\theta_k)} \quad (\text{B.35})$$

$$\tan(2\theta_k) = \frac{2J \sin(k)}{(h_x - 2J \cos(k))} \quad (\text{B.36})$$

Moving back to the Hamiltonian:

$$H = \hbar \sum_k \gamma_k^\dagger \gamma_k \left((h_x - 2J \cos(k)) \cos^2(\theta_k) + iJ \sin(\theta_k) \cos(\theta_k) (e^{-ik} - e^{ik}) \right) + \gamma_{-k} \gamma_{-k}^\dagger \left((h_x - 2J \cos(k)) \sin^2(\theta_k) + iJ \sin(\theta_k) \cos(\theta_k) (e^{ik} - e^{-ik}) \right) \quad (\text{B.37})$$

For the second term we can replace the dummy variable k with $-k$

$$H = \hbar \sum_k \gamma_k^\dagger \gamma_k \left((h_x - 2J \cos(k)) \cos^2(\theta_k) + iJ \cos(\theta_k) \sin(\theta_k) (-e^{ik} + e^{-ik}) \right) + \gamma_k \gamma_k^\dagger \left((h_x - 2J \cos(k)) \sin^2(\theta_k) + iJ \cos(\theta_k) \sin(\theta_k) (-e^{-ik} + e^{ik}) \right) \quad (\text{B.38})$$

$$H = \hbar \sum_k \gamma_k^\dagger \gamma_k \left((h_x - 2J \cos(k)) \cos^2(\theta_k) + 2J \sin(k) \cos(\theta_k) \sin(\theta_k) \right) + \gamma_k \gamma_k^\dagger \left((h_x - 2J \cos(k)) \sin^2(\theta_k) - 2J \sin(k) \cos(\theta_k) \sin(\theta_k) \right) \quad (\text{B.39})$$

$$H = \hbar \sum_k \gamma_k^\dagger \gamma_k \left((h_x - 2J \cos(k)) (\cos^2(\theta_k) - \sin^2(\theta_k)) + 2J \sin(k) \cos(\theta_k) \sin(\theta_k) \right) - (h_x - 2J \cos(k)) \sin^2(\theta_k) - 2J \sin(k) \cos(\theta_k) \sin(\theta_k) \quad (\text{B.40})$$

$$H = \hbar \sum_k \gamma_k^\dagger \gamma_k \left((h_x - 2J \cos(k)) \cos(2\theta_k) + 2J \sin(k) \sin(2\theta_k) \right) + E_g \quad (\text{B.41})$$

Using the trig identity $A \cos(\theta) + B \sin(\theta) = C \sin(\theta + \phi)$ where $C^2 = A^2 + B^2$ and $\tan(\psi) = \frac{A}{B}$ we can rewrite the Hamiltonian. Also because in our case $\tan(\theta) = \frac{B}{A}$ the sum of the two angles is 90° .

$$H = \hbar \sum_k \gamma_k^\dagger \gamma_k \sqrt{(h_x - 2J \cos(k))^2 + 4J^2 \sin^2(k)} + E_g \quad (\text{B.42})$$

$$H = \hbar \sum_k \gamma_k^\dagger \gamma_k 2J \sqrt{1 + \left(\frac{h_x}{2J}\right)^2 - \frac{h_x}{J} \cos(k)} + E_g \quad (\text{B.43})$$

$$H = \sum_k \hbar \omega_k \gamma_k^\dagger \gamma_k + E_g \quad (\text{B.44})$$

where $\omega_k = 2J \sqrt{1 + \left(\frac{h_x}{2J}\right)^2 - \frac{h_x}{J} \cos(k)}$.

The ground state energy is (with the previously dropped constant term included):

$$E_g = - \sum_n^N \hbar \frac{h_x}{2} + \sum_k (h_x - 2J \cos(k)) \sin^2(\theta_k) - 2J \sin(k) \cos(\theta_k) \sin(\theta_k) \quad (\text{B.45})$$

B.1 Shifting Transformation

We can further simplify the cavity-TFIM Hamiltonian. Doing so also has an effect on the bath which can be neglected. This is explored in more detail in the section following this one. We start with a reminder of the Hamiltonian:

$$\hbar \omega_c a a^\dagger a + \hbar \lambda (a^\dagger + a) \sum_k (1 - 2c_k^\dagger c_k) \quad (\text{B.46})$$

Shifting the operators $a \rightarrow a - \alpha$ (this can be done more explicitly with a unitary transformation [48]) preserves the commutation relations so long as α is a c-number. It leads to the Hamiltonian

$$\hbar \omega_c a^\dagger a - \hbar \omega_c \alpha (a^\dagger + a) + |\alpha|^2 + \hbar \lambda (a^\dagger + a - 2\alpha) \sum_k (1 - 2c^\dagger c) \quad (\text{B.47})$$

Expanding out the terms and ignoring constants:

$$\hbar \omega_c a^\dagger a - \hbar \omega_c \alpha (a^\dagger + a) + N \hbar \lambda (a^\dagger + a) - 2 \hbar \lambda (a^\dagger + a) \sum_k c^\dagger c + 4 \hbar \lambda \alpha \sum_k c^\dagger c \quad (\text{B.48})$$

The middle two terms can be removed by a suitable choice of α ($\alpha = \frac{N\lambda}{\omega_c}$) but

that leaves us with

$$\hbar\omega_c a^\dagger a - 2\hbar\lambda(a^\dagger + a) \sum_k c^\dagger c + 4\hbar \frac{N\lambda^2}{\omega_c} \sum_k c^\dagger c \quad (\text{B.49})$$

Now examine the full Hamiltonian:

$$H_{TFIM0} + \hbar \frac{h_x}{2} \sum_k (1 - 2c_k^\dagger c_k) + \hbar\omega_c a^\dagger a - 2\hbar\lambda(a^\dagger + a) \sum_k c^\dagger c + 4\hbar \frac{N\lambda^2}{\omega_c} \sum_k c^\dagger c \quad (\text{B.50})$$

$$H_{TFIM0} + \hbar \frac{h_x}{2} \sum_k (1 - (2 - \frac{8N\lambda^2}{\omega_c h_x}) c_k^\dagger c_k) + \hbar\omega_c a^\dagger a - 2\hbar\lambda(a^\dagger + a) \sum_k c^\dagger c \quad (\text{B.51})$$

Constants:

$$\frac{N^2\lambda^2}{\omega_c^2} - 2\hbar \frac{N^2\lambda^2}{\omega_c} = \frac{N^2\lambda^2}{\omega_c^2} (1 - 2\hbar\omega_c) \quad (\text{B.52})$$

Adding in constants

$$H_{TFIM0} + \hbar \frac{h_x}{2} \sum_k (1 + \frac{2N\lambda^2}{\hbar h_x \omega_c^2} (1 - 2\hbar\omega_c) - 2(1 - \frac{4N\lambda^2}{\omega_c h_x}) c_k^\dagger c_k) + \hbar\omega_c a^\dagger a - 2\hbar\lambda(a^\dagger + a) c_k^\dagger c_k \quad (\text{B.53})$$

Simplify:

$$H_{TFIM0} + \hbar \frac{h_x}{2} \sum_k (1 - 2c_k^\dagger c_k - (\frac{4N\lambda^2}{\omega_c h_x} (1 - 2c_k^\dagger c_k)) + \frac{2N\lambda^2}{\hbar h_x \omega_c^2}) + \hbar\omega_c a^\dagger a - 2\hbar\lambda(a^\dagger + a) c_k^\dagger c_k \quad (\text{B.54})$$

or

$$H_{TFIM0} + \hbar \frac{h_x}{2} \sum_k (A_1 - 2A_2 c_k^\dagger c_k) + \hbar\omega_c a^\dagger a - 2\hbar\lambda(a^\dagger + a) \sum_k c^\dagger c \quad (\text{B.55})$$

$$A_1 = 1 + \frac{2N\lambda^2}{\hbar h_x \omega_c^2} (1 - 2\hbar\omega_c) = A_2 + \frac{2N\lambda^2}{\hbar h_x \omega_c^2} \quad (\text{B.56})$$

$$A_2 = 1 - \frac{4N\lambda^2}{\omega_c h_x} \quad (\text{B.57})$$

Dropping just the $|\alpha|^2$ term:

$$H_{TFIM0} + \hbar \frac{h'_x}{2} \sum_k (1 - 2c_k^\dagger c_k) + \hbar \omega_c a^\dagger a - 2\hbar \lambda (a^\dagger + a) \sum_k c_k^\dagger c_k \quad (\text{B.58})$$

with:

$$h'_x = h_x - \frac{4N\lambda^2}{\omega_c} \quad (\text{B.59})$$

B.2 Effect of the shift on the bath

The effect of shifting the cavity operators by α was neglected in the previous section. Here we fill in a few details. In shifting the cavity operator the term in the Hamiltonian responsible for the cavity-bath coupling also shifts. This requires a different value for the cavity shift α . This will also alter the effective magnetic field. The Hamiltonian takes the form:

$$H = H_{TFIM0} + H_{cavity0} + H_{bath0} + \lambda A \sum_k (1 - 2c_k^\dagger c_k) + A \sum_i c_i (b_i^\dagger + b_i) \quad (\text{B.60})$$

By shifting the cavity operators there will be an extra term created due to the cavity-bath coupling. This can be rectified by also shifting the bath operators. The process is very similar to what was done in the previous section. We shift the operators $a \rightarrow a - \alpha$ and $b_i \rightarrow b_i - \beta_i$, then we drop the constant terms, and finally we require that the extraneous terms cancel. This leads to a set of two equations to be solved simultaneously.

$$(a + a^\dagger)(-\alpha\omega_c + \lambda N - 2 \sum_i c_i \beta_i) = 0 \quad (\text{B.61})$$

$$\sum_i (b_i + b_i^\dagger)(-2\alpha c_i - \omega_i \beta_i) = 0 \quad (\text{B.62})$$

The first two terms in equation B.61 are due to the cavity-TFIM part of the Hamiltonian as before. Solving these equations with the realization that equation

B.62 must be zero for each mode individually gives:

$$\alpha = \frac{\lambda N}{\omega_c} \left(\frac{1}{1 - \frac{4}{\omega_c} \sum_i \frac{c_i^2}{\omega_i}} \right) \quad (\text{B.63})$$

The last term in the denominator is small as can be seen by the following argument. We consider the bandwidth to be $\kappa = J(\omega) = \pi \sum_i \frac{c_i^2}{\omega_i} \delta(\omega - \omega_i)$ [67, 68]. The term in the denominator will be significantly smaller than this as the coupling term is reduced by a factor of the frequency of the bath modes. $\frac{\kappa}{\pi} \gg \sum_i \frac{c_i^2}{\omega_i}$ This is further reduced by the cavity frequency. Thus the effect on α is small and the effect on the effective magnetic field is also small. Thus we neglect this correction for all parts of this work.

Appendix C

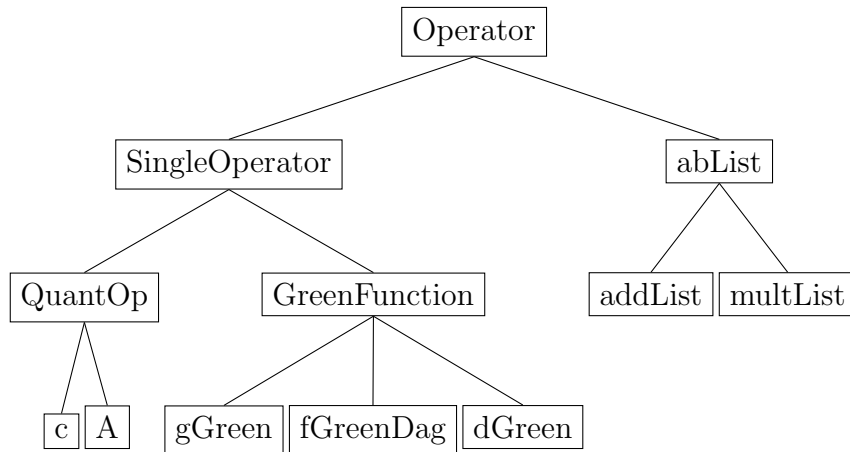
Code

I have include a description of the main program I designed and built to assist in this work. Code snippets are included for clarity. Work was done in Python 3 using the NumPy [69] and SymPy [70] libraries. Code is altered for length.

C.1 Operator

The Operator program defines the objects used in the preceding code. It is necessary because SymPy's symbolic programming does not always write out the equations in a way that is easy or efficient to parse. By building my own symbolic programming machinery I am able to simplify equations in a way which makes the most sense for representing diagrams and for carrying out the Matsubara summation.

The Operator module defines operators as an python object and allows for convenient definitions for addition, multiplication, and more. In particular it allows me to easily manipulate objects to develop the 4th order expansion of the Green's function and evaluate it analytically. The bare-bones layout of the Operator module is given below.



Operator Lays the foundation for the objects used. Used for type checking

SingleOperator A single operator. The basic algebraic rules are established here.

QuantOp Quantum Operator. Time is set as the argument, the contraction function is added, and QuantOp has a list of Green's functions to aid in the contraction.

c The operator as defined above. Its symbols, latex code, contractions, and more are further defined.

A The photon operator. Its symbols, latex code, contractions, and more are further defined.

GreenFunction Describes the Green's function. Carries all the information needed to draw the associated line.

gGreen \mathcal{G}

fGreenDag \mathcal{F}^\dagger

dGreen \mathcal{D}

abList Abstract list.

addList A list whose elements are considered to be summed.

multList A list whose elements are considered to be multiplied.

```

class Operator:
    def __init__(self):
        """Largely for checking instances"""
        self.inverse = False #To deal with inverses and divisions
        self.sign=1 # Sign is a property
    def new(self): #Operations return new objects
        newOp= self.__class__()
        newOp.sign=self.sign
        return newOp
    @staticmethod
    def getSign(thing):
        if isinstance(thing,Operator):
            return thing.sign
        num=float(thing)
        if num<0:
            return -1
        if num==0:
            return 0
        return 1

class SingleOperator(Operator):
    def __init__(self,char,sym,arg,subscripts,dagger,fermion):
        Operator.__init__(self)
        """Creates an operator. dagger and fermion are booleans.
        Currently arg cannot be complex"""
        self.arg=arg #arguments of the operator (time, momentum, etc)
        self.dagger=dagger #is it daggered?
        self.fermion=fermion #is it a fermion?
        self.char=char #for presentation and latex code
        self.sym=sym #subscripts should be added manually

        self.subscripts=[str(subs) for subs in subscripts] #list of subscripts
    def dag(self):

```



```

        """Returns a daggered version of the old operator"""
dagger=self.dagger!=True
        newOp=self.__class__(self.subscripts,self.time,dagger,self.char)
        newOp.sign=self.sign
        return newOp
    def __add__(self,other):
        """Creates an addList with the operator in it or appends it to
an addList"""
        if isinstance(other,Operator):
            other=other.new()
        newSelf=self.new()
        if isinstance(other,addList):
            if len(other)==0:
                return newSelf #empty addList acts as identity.
            else:
                other.append(newSelf)
                return other
        if other==0:
            return self #identity
        if other==self: #If they are the same thing but opposite sign
            return 0
        return addList(self,other)
    def __mul__(self,other):
        """Creates a multList with the operator in it or appends it to
a multList"""
        if isinstance(other,Operator):
            other=other.new()
        newSelf=self.new()
        if isinstance(other,abList): #can be an addList or a multList
#multiplication is defined by the particular abstract list
            return other.__rmul__(newSelf)
        else:
#mult list of single operator and single operator/numeric

```

```

        return multList(newSelf,other)
def __eq__(self,other): #Defines equality for single operators
    if isinstance(other,self.__class__):
        otherSign=Operator.getSign(other)
        if self.sign==otherSign: #check signs
            return self.latex==other.latex #check latex code
    elif isinstance(other, str):
        return self.sym==other #probably not necessary
    return False #else return false
def new(self): #define a new function for ease
    if isinstance(self.arg,Operator):
        newArg=self.arg.new() #get a new argument
    else:
        newArg=self.arg
    newSubs=[]
    for item in self.subscripts: #Why?
        newSubs.append(item)
    newOp=self.__class__(
self.char,self.sym,newArg,newSubs,self.dagger,self.fermion)
        newOp.sign=self.sign
    return newOp

#abList is a virtual class

class addList(abList): #contains elements that are being added
    def __init__(self,*args):
        abList.__init__(self,*args)
    def __add__(self,other):
        if isinstance(other,addList):
            newList=addList()
            for item in self: #get back the addList type
                newList.append(item)
            for item in other:

```

```

        negItem=-item #check to see if there are cancellations
        negFound=False
        i=0
        while i<len(newList) and not negFound:
            if negItem==newList[i]: #they cancel
                newList.pop(i)
                negFound=True
                i=i-1
            i+=1
        if not negFound:
            newList.append(item)
    return newList
else:
    if len(self)==0:
        return other
    other=addList(other)
    return self+other
def __mul__(self,other): #automatically distributes
    newList=addList()
    if isinstance(other,Operator):
        newOther=other.new()
    else:
        newOther=other
    for item in self:
        if isinstance(item,Operator):
            newItem=item.new()
        else:
            newItem=item
    newList=newList+newItem*newOther
    return newList

class multList(abList): #contains elements thar are bein multiplied
    def __init__(self,*args): #What if it's called with only one argument?

```

```

abList.__init__(self,*args)
sign=1
for i in range(0,len(self)): #Getting all the signs into the list
    item=self[i]
    sign=sign*Operator.getSign(item)
    if isinstance(item, Operator):
        item.sign=1
    else:
        if isinstance(item,complex):
            raise Exception("Complex not not implimented")
        self[i]=abs(float(item))
self.sign=sign

def __add__(self,other):
    if isinstance(other,Operator):
        other=other.new()
    newList=addList(self,other)
    return newList

def __mul__(self,other):
    if isinstance(other,Operator):
        other=other.new()
    if len(self)==0:
        return other
    if other==1: #identity
        temp=self.new()
        return temp #No more strings!
    if other==-1:
        temp=self.new()
        temp.sign=temp.sign*-1
        return temp
    if isinstance(other,multList):
        newList=multList()
        for item in self:

```

```

        newList=newList*item #build up the list from self
    for item in other:
        newList=newList*item #include items from the other list
    newList.sign=self.sign*other.sign
    return newList
if isinstance(other,addList): #defined in addList
    return other.__rmul__(self)
else: #Single Operator or a/n float/int
    if len(self)==0:#Empty list is identity
        return other
    newList=multList()
    for item in self:
        newList.append(item)
    s=Operator.getSign(other)
    if isinstance(other,complex):
        raise Exception("Complex not implimented yet")
    newList.append(abs(other))
    newList.sign=s*self.sign
    #Should I keep track if it's a Fermion?
    return newList
def pullOut(self):#pulls out any constants
    newList=multList()
    constant=1
    for item in self:
        if isinstance(item,int) or isinstance(item,float):
            constant*=item
        elif isinstance(item,complex):
            raise Exception("Complex not implimented yet")
        else:
            newList*=item
    newList.sign=self.sign
    return constant,newList

```

C.2 Diagrams

The code to produce diagrams is shown below. An interaction Hamiltonian is defined. The code assumes that the regular perturbation theory holds. The correlator associated with the n th order is produced. This correlator is then broken down with Wick's theorem. The equation that results can be interpreted as a sum of diagrams. Each term in the sum is converted to a diagram via an adjacency list. A depth first transversal is carried out to see if the diagram is connected. Finally equality between diagrams is defined. The diagrams are displayed with equal diagrams on the same page and differing diagrams on subsequent pages.

An AdjList is an object I created that stores the needed information about the adjacency list related to the graph (Diagram). A simple example is shown in figure C.1. In this case the adjacency list is slightly more complicated to take into account the different types of edges (Green's functions). The AdjList object stores this adjacency list as well as the functions to check for connectedness. Connectedness is determined using a depth first transversal, shown schematically in figure C.2.

The Diagram class has a list of Green's functions. From those Green's functions it makes and stores an AdjList. It then makes and stores every AdjList corresponding to a permutation of the internal vertices. This is done to define equality in an easy to use way. D_1 equals D_2 if the first AdjList in D_1 equals an AdjList in D_2 .

The Diagram object also allows transformation of the Green's function into frequency space. Each vertex has 3 lines coming in or out. Each of these lines is given a frequency (bose or fermion) ω . Incoming lines are positive, outgoing are negative. The sum of these frequencies are zero. Using the Sympy Solver method I can rewrite the frequencies in terms of the incoming frequency (ω_n) and $\frac{N}{2}$ free frequencies (N is the order).

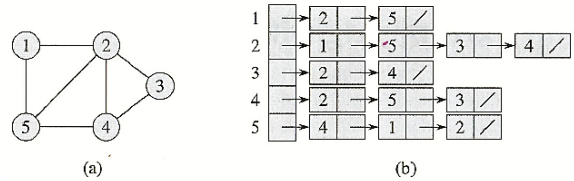


Figure C.1: Descriptions of an adjacency list, adapted from Introduction to Algorithms [3]. (a) An example of a undirected graph (b) The adjacency list for the graph in (a)

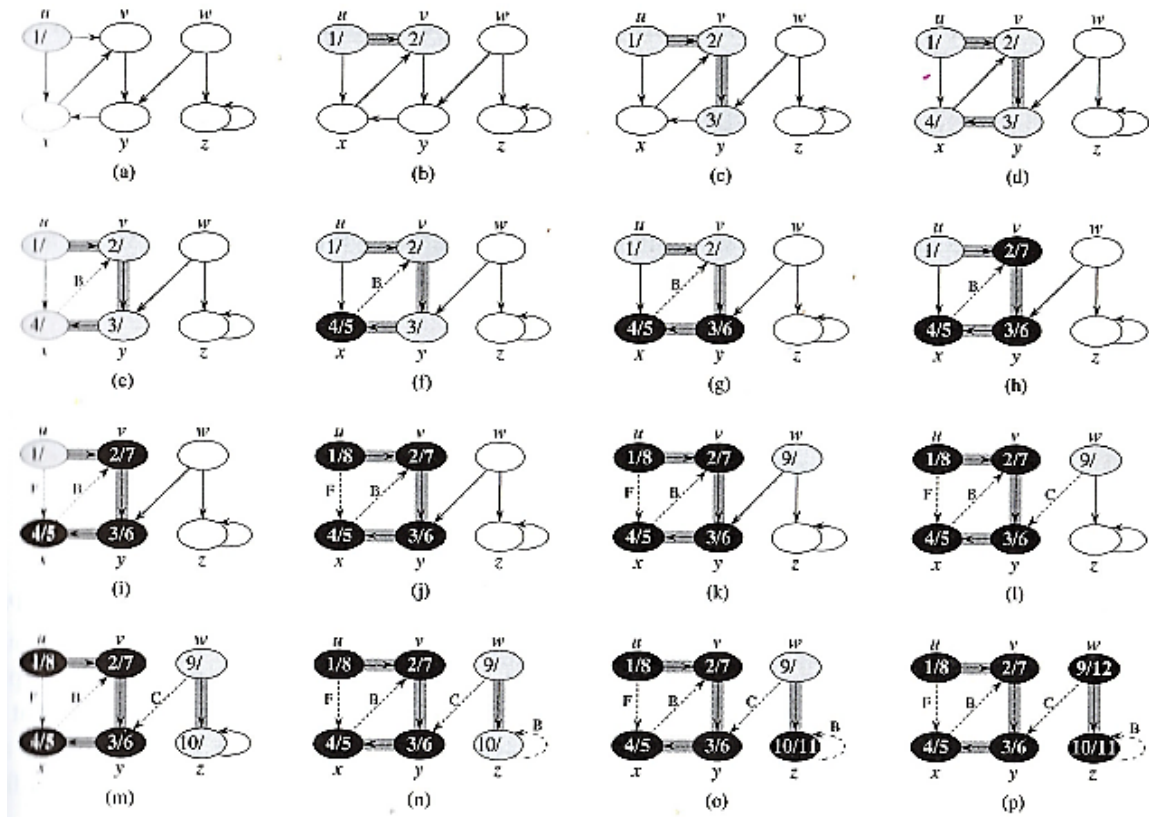


Figure C.2: Depth first transversal example. Taken from Introduction to Algorithms [3].

```
#Defines the Green's functions
class GreensFunction(SingleOperator):
    def __init__(
self,final,initial,fermion,eqn,drawParameters,char,subs=[],upOrDown=0):
        """Super class for Green's functions.Fermion is a boolean,
        eqn is the string representing the Green's function
        and drawParameters is a list of parameters"""
        self.final=final
        self.initial=initial
        SingleOperator.__init__(
self,char,"",subscripts=subs,dagger=False,fermion=fermion)
        self.arg = "t"+str(final)+"-t"+str(initial)
```

```

        self.w= NULLOP()
        self.feynman=""
        self.eqn=eqn
        self.k=0
        drawConditions=["lineType","start","extent",\
"style","dash","tags","UpOrDown"]
        self.drawDict={}
        for i in range(0,len(drawConditions)):
            self.drawDict[drawConditions[i]]=drawParameters[i]
        def draw(self,vi,vf,blackboard,l2h=0.2,scale=1):
            """Draws a line based on the parameters and the vertices.
            Draws to a tkinter canvas"""
        class dGreen(GreensFunction):
            def __init__(self,final,initial,subs=[],upOrDown=1):
                self.upOrDown=upOrDown
                #lineType,start,extent,style,dash,tags,UpOrDown
                drawParameters=["LINE",0,0,tk.NONE,False,\
"D"+str(final)+str(initial),-1]
                char=\
"\mathcal{D}^0(\tau_{"+str(final)+"}-\tau_{"+str(initial)+"})"
                GreensFunction.__init__(
self,final,initial,False,"D",drawParameters,char)
                self.sym="D0"
                self.feynman="\fmf{fermion,label=$\mathcal{D}^0$}{"+str(initial)+\
","+str(final)+"}" #defines the LaTeX code for FeynmanMP/MF
            def new(self):
                newGreen= self.__class__(
self.final,self.initial,self.subscripts,self.upOrDown)
                newGreen.sign=self.sign
                return newGreen

        class Vertex:
            """Has a label, and input, and an output. Has a draw method.

```



```

Operators are overloaded"""

class VertexEdge: #Green's function is the edge.
    """Contains a start point, end point, and Green's function.
    Addition and multiplication are overloaded"""

class singleAdjList():
    def __init__(self,order):
        self.adjList=np.empty(order+2,list)
        for i in range(0,order+2):
            self.adjList[i]=[]
        self.length=order+2
    def __getitem__(self,key):
        return self.adjList[key]
    def __eq__(self,other):
        if not isinstance(other,singleAdjList):return False
        equals=True
        i=0
        while i in range(0,self.length) and equals:
            selfVertex=self.adjList[i]
            otherVertex=other[i]
            if len(selfVertex)==len(otherVertex):
                inIt=True
                for edge in selfVertex:
                    inIt=inIt and edge in otherVertex
                equals=inIt
            else:
                equals=False
            i+=1
        return equals
    def __len__(self):
        return self.length
class AdjList:

```

```

def __init__(self,order,greenFuns,makeAll=True):
    self.order=order
    self.greenFuns=greenFuns
    self.adjLists=[self.MakeAdjList()]
    self.directedList=self.MakeAdjList()
    self.unDirected=self.CreateUndirected()
    self.disconnected=self.isDisconnected()
    if not self.disconnected:
        self.createAllAdj()
def MakeAdjList(self):
    """Takes a list of lists of green's functions and returns an
adjacency list"""
    tempArr=singleAdjList(self.order)
    vEList=[]
    for prodTerm in self.greenFuns:
        #for n order diagram it should have n + 2 times and (n+1)!!
#times(2n-1)!! total terms
        #(3/2n+1 per term)
        final=prodTerm.final #to
        initial=prodTerm.initial #from
        vertexEdge=VertexEdge(final,initial,prodTerm)
        vEList.append(vertexEdge.length)
        if not vertexEdge in tempArr[initial]:
            tempArr[initial].append(vertexEdge)
    return tempArr
def createAllAdj(self,undirected=True):
    """Produces all the adjLists which are permutations of the
internal vertices"""
def __eq__(self,other):
    if not isinstance(other,AdjList):return False
    equals=False
    tester=self[0]
    if len(self)!= len(other):return False

```

```

        i=0
        while not equals and i < len(self):
            item=other[i]
            equals=equals or tester==item# all(tester==item)
            i+=1
        return equals
class Diagram:
    def __init__(self,order,greenFuns,constant=1,makeAll=True):
        self.order=order
        self.constants=constant
        self.greenFuns=greenFuns
        self.commonFactor=1
        self.adjLists=AdjList(order,greenFuns,makeAll)
        self.adjList=self.adjLists[0]
        self.disconnected=self.adjLists.disconnected
        #self.greenFunsFreq=self.transform(greenFuns)
        self.vertices=self.MakeVertices()
        self.uVertices=self.MakeVertices(False)
    def MakeVertices(self,directed=True):#Check for accuracy
        """Makes the vertices for the diagram"""
        return vertexList
    def __eq__(self,other):
        if not isinstance(other,Diagram): return False
        areEqual=self.adjLists==other.adjLists
        return areEqual
    def __getitem__(self,key):
        return self.vertices[key]
    def draw(self,x,y,xLength,blackboard,l2h=0.2,scale=1):
        """draw method"""
        return maxHeight
    def toFrequency(self):
        tList=[] #represents each vertex, contains the frequencies in/out
        for i in range(0,self.order+2):

```

```

        tList.append([])
    greenWList=[]
    wn=boseFrequency("w_n",real=True)#Symbol("w_n",real=True)
    i=0
    for green in self.greenFuns:
        to=green.final
        fro=green.initial
        if fro ==0: #inital line
            tempW=wn
        else:
            char=str(i)
            if isinstance(green,dGreen):
                tempW=boseFrequency("w_"+char,real=True) #boson
            else:
                tempW=fermiFrequency("w_"+char,real=True) #Fermion
            i+=1
        greenWList.append([green,tempW])
        tList[to].append(tempW)
        tList[fro].append(-tempW)
    startEnd=tList.pop(0)[0]+tList.pop()[0] #wn + w_final=0
    solveList=[startEnd]
    for item in tList:#Creating list of equations to solve
        s=0
        for wTerm in item:
            s+=wTerm
        solveList.append(s) #w_1+w_2-w_3+.....=0
#Sympy solver, don't solve for wn in solution.keys())
    solution=solve(solveList,exclude=[wn])
    if solution==[]:
        solution=solve(solveList)
    finalGreenWList=[]
    for greenW in greenWList:
        green=greenW[0]

```

```

        w=greenW[1]
        if w in solution:
            newW = solution[w] #gets the solution for that frequency
        else:
            newW=w #not solved for, i.e. free frequency
        finalGreenWList.append((green,newW))
    return finalGreenWList

def WickIt(order,N=20): #make it for a more general interaction term
    """Creates operators and all the wick permutations to the order given,
    order should be an even number"""
    if order==0:
        return ([],[])
    gProd=multList()
    aProd=multList()
    i=1
    if regularHam:
        while i<= order: #Building interaction Hamiltonian terms
            gProd=gProd*(-2*c(i,True)*c(i,False))#Lambda and hbar removed
            aProd=aProd*A(i)
            i=i+1
        aProd=aProd*A(order+1)*A(0)*A(order+1)*A(0)
        contractedA=Pairings(aProd)
        contractedG=addList()
        if isinstance(gProd,addList):
            for item in gProd:
                #Either constants or products
                if isinstance(item,multList):
                    #pull out constants to do the pairings
                    constTerm,prodTerm=item.pullOut()
                    #add List times constant
                    contractedGTerms=constTerm*Pairings(prodTerm)
            #add all the contracted terms together

```

```

        contractedG=contractedG+contractedGTerms
    else:
        contractedG=contractedG+item
    return contractedA*contractedG
else:
    while i<= order: #Building interaction Hamiltonian terms
        gProd=gProd*(c(i,True)*c(i,False))
        aProd=aProd*A(i)
        i=i+1
    aProd=aProd*A(order+1)*A(0)
    contractedA=Pairings(aProd)
    contractedG=Pairings(gProd) #Lambda and hbar removed
    return contractedA*contractedG

def Pairings(toPair):
    """Takes in a multList to pair (really any iterable),
    creates an addList of all the different fully paired permutations.
    Returns a addList of products of contractions."""
    return pairList

```

C.3 Matsubara Summations

A skeleton version of the code used to analyze the fourth order TFIM diagrams. This program takes in the diagrams produced by the Diagram.py code and turns them into an equation format. It multiplies the values out so that each term in the equation is a product of simple or second order poles. The code solves the Matsubara summation symbolically and returns a string which is the Green's function. Typically this is a very large string. A description of the code is below

1. This program first defines the interaction Hamiltonian (H_I) in terms of QuantOps and then multiplies $A(\tau_f)A(\tau_i)$ by H_I N times.

2. Each term in the list is paired up (contracted) in every possible permutation (Wick's theorem). An addList of multLists of Green's functions is returned.
3. Each Green's function is turned into a "Diagram" object.
4. These diagrams have an adjacency list to represent the diagram (see *Introduction to Algorithms* by Cormen [3])
5. The adjacency list creates every possible permutation of the internal indexes to get every possible adjacency list representation of the diagram
6. An undirected adjacency list is created and a depth first transversal from start to end is done to see if it is connected. Disconnected diagrams will be ignored. To keep partially connected diagrams another transversal from end to start is also done.
7. The diagrams are converted to frequency space by giving each green's function in the diagram a unique ω symbol and sign, adding all the frequencies together for each vertex (requiring it equals zero), and then solving using the Sympy solver. The type of frequency is kept track of (boson or fermion)
8. The adjacency lists are taken and used to draw the diagrams. Each Green's function object has in it the information to draw the correct line.
9. Diagrams which are equal (have an adjacency list in common) are placed on the same page, each unique diagram gets its own page.
10. Improper diagrams are checked by hand and then discarded.
11. The counting factor for each unique diagram can then be recorded.

```
#Build the green's function
def buildGreen(highestOrder,N=20):

    global maxUKList
```

```

sumTerm=""
#Find all the proper diagrams and their counting factors
listOfImproperDiagrams_2=[]
if keepPartiallyConnectedTerms:
    listOfImproperDiagrams_4=\
    [0,3,6,7,22,23,28,29,32,33,37,38,42,45,48,49,50,51,52,53]
    listOfImproperDiagrams_4=[i+6 for i in listOfImproperDiagrams_4]
else:
    listOfImproperDiagrams_4=[12,13,18,19]
    listOfImproperDiagrams_4=[i+2 for i in listOfImproperDiagrams_4]
order=2
diagramList=[]
while order<=highestOrder:
    wickList=WickIt(order,N)
    #WickIt returns an addList of multLists of green's functions.
    #AKA the summation of the diagrams
    for prod in wickList: #each prod is a diagram
        constant,greens=prod.pullOut()
        diagram=Diagram(order,greens,constant)
        if not diagram.disconnected:
            i=0
            alreadyIn=False
            while i <len(diagramList) and not alreadyIn:
                diagram_in_list=diagramList[i]
                alreadyIn= (diagram == diagram_in_list)
                i+=1
            if not alreadyIn:
                diagramList.append(diagram)
    order+=2

listOfImproperDiagrams=listOfImproperDiagrams_2+\
                        listOfImproperDiagrams_4
for dCount,diagram in enumerate(diagramList):

```



```

#get rid of the improper diagrams
if not dCount in listOfImproperDiagrams:
    if dCount<2:
        order=2
    if dCount>=2:
        order=4
    constant=diagram.constants*diagram.countingFactor
    greensInFreq=diagram.toFrequency()
    uWList=[]
    product=multList()
    uKList=[]
    sign=diagram.greenFuns.sign
    bubbleCount=0
    #print(greensInFreq)
    for greenW in greensInFreq:
        green=greenW[0]
        w=greenW[1]
        tFrom=green.initial
        tTo=green.final

        args=w.args
        isBubble=False
        if tFrom==tTo:#bubble term
            bubbleCount+=1
            isBubble=True
        else:
            green.k=0
            if green!="DO" and not 0 in uKList:
                uKList.append(0)
    if not isBubble:# Finding all unique omegas to sum over
        if len(args)==0:#single arugmement
            shouldAppend=not w in uWList and w!=wn and w!=0\
            and w!=-wn and not -w in uWList and not isBubble

```

```

        if shouldAppend:
            uWList.append(w)
    else:
        #Term may be an addition of two terms
        for arr in args:
            args2=arr.args
            if len(args2)==0:
                shouldAppend=not arr in uWList and arr!=wn\
                and arr!=0 and arr!=-wn \
                and not -arr in uWList \
                and not isBubble\
                and not isinstance(arr,int)\
                and not arr==-1
                if shouldAppend:
                    uWList.append(arr)
            else:
                for arr2 in args2:
                    shouldAppend2=not arr2 in uWList and \
                    arr2!=wn and arr2!=0 and arr2!=-wn \
                    and not -arr2 in uWList \
                    and not isBubble and not\
                    isinstance(arr2,int) and not arr2==-1
                    if shouldAppend2:
                        uWList.append(arr2)
    if tFrom==0:
        if w==0:
            onlyIfZero=True
        else:
            onlyIfZero=False
    if not(tFrom==0 or tTo==order+1) and not isBubble:
        if constant<0:
            sign=-sign
            constant=-constant

```

```

        if w!= wn:
            product = product*gFun(green,w,green.k)
            if isinstance(product,Expr):
                product=multList(product)
numHbarBeta=(order+1+len(uWList)) - len(diagram.greenFuns)+\
            bubbleCount
hbarBetaString="*(hbar*beta)**("+str(numHbarBeta)+")"
    #number of (hbar*beta)^-1 is equal to the number of
    #green's functions
    #number of hbar*beta is the order, plus 1 (from integrals)+
    #the number of mat sums
if len(product)!=0:
    #carries out the summation
    integratedProduct=matSum(product,uWList)
    fullSum=0
    for gSumTerm in integratedProduct:
        fullProd=gSumTerm.sign
        for gProd in gSumTerm:
            fullProd*=gProd
        fullSum+=fullProd

    for item in uKList:
        if not item in maxUKList:
            #Gets all the ks we are going to need
            maxUKList.append(item)
    fullSum=fullSum.expand()
    #for easier debugging
    comments="\n#Diagram Number: "+str(dCount)
    sumTerm+=comments
    #k summation
    tabs="\t"
    kSumString=""
    if len(uKList)==1:

```

```

sumTerm=sumTerm.replace("*replaceMe*", "\n"+tabs+
                        "wk_0=wkArray_0"+ "\n"+tabs+
                        +"thetaK_0= thetaArray_0")

if sign>0:
    kSumString+="\n"+tabs+"sumTerm+=(2*1)**"+\
                str(order)+"*(np.sum("+str(fullSum)+")"
else:
    kSumString+="\n"+tabs+"sumTerm-=(2*1)**"+\
                str(order)+"*(np.sum("+str(fullSum)+")"
sumTerm+=kSumString+")*bubbleTerm("+str(N)+", "+\
str(bubbleCount)+", "+str(onlyIfZero)+")*( "+\
str(constant)+")"+hbarBetaString
#Bubble terms factor out, will be raised to the power
#bubbleCount in def. Multipliy by constants and hbars and
#betas (not the most effcient)
else:
    #for easier debugging
    comments="\n#Diagram Number: "+str(dCount)
    sumTerm+=comments
    if sign>0:
        sumTerm+="\n\tsumTerm+=(2*1)**"+str(order)+\
                "*bubbleTerm("+str(N)+", "+str(bubbleCount)+", "+\
                str(onlyIfZero)+")*( "+str(constant)+")"+hbarBetaString
    else:
        sumTerm+="\n\tsumTerm-=(2*1)**"+str(order)+\
                "*bubbleTerm("+str(N)+", "+str(bubbleCount)+", "+\
                str(onlyIfZero)+")*( "+str(constant)+")"+hbarBetaString
sumTerm+="\n\td=D0(I*w_n)"
#Lambda has hbar ^2 as well, cancels the sumTerm 1/hbar^2
sumTerm+="\n\tret=d/(1-d*sumTerm)"
sumTerm+="\n\treturn ret"
sumTerm+="\ngreenVec=np.vectorize(greenFunction)"

```

```

    return preTerm+sumTerm

def getUKList(kList,product):
    """Takes a kList(indices are one and finds the unique ks"""
    uKs=[]
    visited=[]
    kVertex=0
    while kVertex < len(kList):
        if kVertex in visited:
            kVertex+=1
        else:
            notVisited=[kVertex]
            if not kVertex in uKs:
                uKs.append(kVertex)
            #while there are more vertices that can be visited
            while notVisited !=[]:
                didVisit=notVisited.pop()
                visited.append(didVisit)
                if kVertex!=didVisit:
                    temp=[]
                    for prod in product:
                        temp2=[]
                        for sumTerm in prod:
                            temp2.append(replaceProd(
                                sumTerm,didVisit,kVertex))
                        temp.append(temp2)
                    product=temp
            #visit the ks linked to this one
            for verticies in kList[didVisit]:
                if verticies not in visited:notVisited.append(verticies)
            kVertex+=1
    return uKs,product

```

```

def replaceProd(product,toRep,repW):
    toReplaceString="wk_"+str(toRep)
    toReplace=Symbol(toReplaceString,real=True)
    replaceWithString="wk_"+str(repW)
    replaceWith=Symbol(replaceWithString,real=True)
    product=product.subs(toReplace,replaceWith)

    toReplaceString="uk_"+str(toRep)
    toReplace=Symbol(toReplaceString,real=True)
    replaceWithString="uk_"+str(repW)
    replaceWith=Symbol(replaceWithString,real=True)
    product=product.subs(toReplace,replaceWith)

    toReplaceString="vk_"+str(toRep)
    toReplace=Symbol(toReplaceString,real=True)
    replaceWithString="vk_"+str(repW)
    replaceWith=Symbol(replaceWithString,real=True)
    product=product.subs(toReplace,replaceWith)

    toReplaceString="thetaK_"+str(toRep)
    toReplace=Symbol(toReplaceString,real=True)
    replaceWithString="thetaK_"+str(repW)
    replaceWith=Symbol(replaceWithString,real=True)
    product=product.subs(toReplace,replaceWith)

    return product

def matSum(product,uWList):
    #Takes in product and a list of omegas to sum over
    """gFun is a list (representing multiplication) of
        lists(representing addition)"""
    z=Symbol('z')
    if not isinstance(uWList,list):raise TypeError("uWList must be a list")

```

```

#Break green's functions up
#Gives list (representing addition) of
#lists(representing multiplication)
toIntegrate=product
#flips it so the second order poles are summed last
uWList=flip(product,uWList)
for uW in uWList:
    integratedSum=addList()
    #Check if it's a boson or a fermion and add the correcting
    #weighting factor
    if isinstance(uW,boseFrequency):
        toIntegrate=-1*toIntegrate*nBE(I*uW)
    elif isinstance(uW,fermiFrequency):
        toIntegrate=toIntegrate*nFD(I*uW)
    else:
        raise ValueError(
            "Matsubara frequency is neither a bose nor a fermi frequency")
    if not isinstance(toIntegrate,addList):
        toIntegrate=addList(toIntegrate)
    for summand in toIntegrate:#integrate each term in the addList
        toIntegrate_term=[term.subs(I*uW,z) for term in summand]
        #Do the contour Integral, returns a sum (list)
        integrated=contourIntegral(toIntegrate_term,z)
        integratedSum=integratedSum+integrated #puts the sums together
    toIntegrate=integratedSum #Get ready to integrate again
return integratedSum#integratedSum

def contourIntegral(integrand,z):
    #integrand is a multList of terms, returns a product of sums
    wk=Symbol('wk_'+str(kVal), real=True)
    residues=addList()
    second_order_terms=[]
    constant=multList()

```

```

integrand_2=multList()
pole_sign=1
for term in integrand:
    if len(term.args)==0:
        constant=constant*term
    else:
        arg=str(term)
        if not 'z' in arg:
            if "z" in str(term):
                print(
                    term,z,term.args,term.args[0],term.args[0].atoms())
            if len(constant)==0:
                constant=multList(term)
            else:
                constant=constant*term
        else:
            if len(integrand_2)==0:
                integrand_2=multList(term)
            else:
                integrand_2=integrand_2*term
integrand=integrand_2
#find first and second order terms:
for term_index,term in enumerate(integrand):
    w_term=term.args[0]
    term_2_index=term_index
    second_order=False
    while term_2_index<len(integrand) and not second_order:
        if term_index!=term_2_index:
            term_2=integrand[term_2_index]
            if not isinstance(term_2,nFD) and not isinstance(term_2,nBE):
                w_term_2=term_2.args[0]
                if (w_term==w_term_2):
                    second_order=True

```



```

        if (w_term==w_term_2):
            second_order=True
    if second_order:
        second_order_terms.append(integrand.pop(term_index))
        #-1 to adjust for the first pop, term_2_index is
        #always>=term_index
        second_order_terms.append(integrand.pop(term_2_index-1))
    term_2_index+=1
#now integrand just contains first order poles
#Find the residue of the first order terms
for term in integrand:
    #get the poles and residue from the term
    if isinstance(term,mul_type):
        args=term.args
        assert(args[0]==-1)
        term=args[1]
        constant=-1*constant
    p_rs=term.pole_and_residue(z)
    #iterate through the poles/residues
    for p_r in p_rs:
        pole=p_r[0]
        residue=multList(p_r[1])
        for term_2 in integrand:
            if term!=term_2:
                new_term=term_2.subs(z,pole).expand()
                new_term=convert_distribution(new_term)
                residue= residue*new_term
        for term_2 in second_order_terms:
            #No distributions in second order terms
            residue=residue*term_2.subs(z,pole).expand()
    if len(residues)==0:
        #otherwise it returns a sympy object
        residues=addList(residue)

```

```

else:
    residues=residues+residue
assert(len(second_order_terms)==2 or len(second_order_terms)==0)
#lim z-> z0 d/dz ((z-z0)^2 integrand)
#the second order terms when differentiated reduce to
#(r1*r4+r2*r3)/(2*pole)
#and r1r3(r2r4) when not differentiated for the positive
#(negative) pole r1,r2 are the residues of the first term, r3,r4
#are the residues of the second
if second_order_terms:
    g1=second_order_terms[0]
    g2=second_order_terms[1]
    p_rs_1=g1.pole_and_residue(z)
    p_rs_2=g2.pole_and_residue(z)
    #The assumption here is that the poles are ordered and equal
    pole_1=pole_sign*p_rs_1[0][0]
    pole_2=pole_sign*p_rs_1[1][0]
    r_1=pole_sign*p_rs_1[0][1]
    r_2=pole_sign*p_rs_1[1][1]
    r_3=pole_sign*p_rs_2[0][1]
    r_4=pole_sign*p_rs_2[1][1]
    #first pole
    #The wk term in the denominator only works for the F and G functions
    res=multList((r_1*r_4+r_2*r_3)/(2*wk))
    for term in integrand:
        new_term=term.subs(z,pole_1).expand()
        new_term=convert_distribution(new_term)
        res=res*new_term
    residues=residues+res
for term in integrand:
    new_term=term.diff(z).subs(z,pole_1).expand()
    new_term=convert_distribution(new_term)
    diff_term=multList(new_term)

```

```

    for term_2 in integrand:
        if term_2!=term:
            new_term=term_2.subs(z,pole_1).expand()
            new_term=convert_distribution(new_term)
            diff_term=diff_term*new_term
    diff_term=diff_term*r_1*r_3
    residues=residues+diff_term

#second pole
res=multList((r_1*r_4+r_2*r_3)/(-2*wk))
for term in integrand:
    new_term=term.subs(z,pole_2).expand()
    new_term=convert_distribution(new_term)
    res=res*new_term
residues=residues+res
for term in integrand:
    new_term=term.diff(z).subs(z,pole_2).expand()
    new_term=convert_distribution(new_term)
    diff_term=multList(new_term)
    for term_2 in integrand:
        if term_2!=term:
            new_term=term_2.subs(z,pole_2).expand()
            new_term=convert_distribution(new_term)
            diff_term=diff_term*new_term
    diff_term=diff_term*r_2*r_4
    residues=residues+diff_term
return constant*residues

def convert_distribution(term):
    """Takes care of the matsubara frequencies in the distributions"""
    if not (isinstance(term,nFD) or isinstance(term,nBE) or \
            isinstance(term,dnBE) or isinstance(term,dnFD)): return term
    mat_freq=[]

```

```

num_fermi=0
arg=term.args[0]
atoms=arg.atoms()
for atom in atoms:
    if isinstance(atom, boseFrequency):
        mat_freq.append(atom)
    if isinstance(atom, fermiFrequency):
        mat_freq.append(atom)
        num_fermi+=1
for freq in mat_freq:
    arg=arg.subs(freq,0)
if num_fermi%2==0: #no change
    if isinstance(term, nFD):
        return nFD(arg)
    if isinstance(term,nBE):
        return nBE(arg)
    if isinstance(term,dnFD):
        return dnFD(arg)
    if isinstance(term,dnBE):
        return dnBE(arg)
else:
    if isinstance(term, nFD):
        return -nBE(arg)
    if isinstance(term,nBE):
        return -nFD(arg)
    if isinstance(term,dnFD):
        return -dnBE(arg)
    if isinstance(term,dnBE):
        return -dnFD(arg)

def replace(term, toRep,rep):
    if str(toRep) in str(rep) or str(toRep**2) in str(rep):
        raise ValueError(

```


Bibliography

- [1] F. Yoshihara, T. Fuse, S. Ashhab, K. Kakuyanagi, S. Saito, and K. Semba. Characteristic spectra of circuit quantum electrodynamics systems from the ultrastrong- to the deep-strong-coupling regime. *Phys. Rev. A*, 95:053824, May 2017.
- [2] Zhen Chen, Yimin Wang, Tiefu Li, Lin Tian, Yueyin Qiu, Kunihiro Inomata, Fumiki Yoshihara, Siyuan Han, Franco Nori, J. S. Tsai, and J. Q. You. Single-photon-driven high-order sideband transitions in an ultrastrongly coupled circuit-quantum-electrodynamics system. *Phys. Rev. A*, 96:012325, Jul 2017.
- [3] Thomas H. Cormen. *Introduction to Algorithms*. The MIT Press, 2009.
- [4] Seth Lloyd. Universal quantum simulators. *Science*, 273(5278):1073, Aug 23 1996. Copyright - Copyright American Association for the Advancement of Science Aug 23, 1996; Last updated - 2017-10-31; CODEN - SCIEAS.
- [5] Richard P. Feynman. Simulating physics with computers. *International Journal of Theoretical Physics*, 21(6):467–488, Jun 1982.
- [6] Philipp Hauke, Fernando M Cucchietti, Luca Tagliacozzo, Ivan Deutsch, and Maciej Lewenstein. Can one trust quantum simulators? *Reports on Progress in Physics*, 75(8):082401, jul 2012.
- [7] J. Ignacio Cirac and Peter Zoller. Goals and opportunities in quantum simulation. *Nature Physics*, 8:264 EP –, Apr 2012.

- [8] Feng Mei, Vladimir M. Stojanović, Irfan Siddiqi, and Lin Tian. Analog superconducting quantum simulator for holstein polarons. *Phys. Rev. B*, 88:224502, Dec 2013.
- [9] A. P. Burgers, L. S. Peng, J. A. Muniz, A. C. McClung, M. J. Martin, and H. J. Kimble. Clocked atom delivery to a photonic crystal waveguide. *Proceedings of the National Academy of Sciences*, 116(2):456–465, 2019.
- [10] Alexandre Cooper, Jacob P. Covey, Ivaylo S. Madjarov, Sergey G. Porsev, Marianna S. Safronova, and Manuel Endres. Alkaline-earth atoms in optical tweezers. *Phys. Rev. X*, 8:041055, Dec 2018.
- [11] Hannes Bernien, Sylvain Schwartz, Alexander Keesling, Harry Levine, Ahmed Omran, Hannes Pichler, Soonwon Choi, Alexander S. Zibrov, Manuel Endres, Markus Greiner, Vladan Vuletic, and Mikhail D. Lukin. Probing many-body dynamics on a 51-atom quantum simulator. *Nature*, 551:579 EP –, Nov 2017. Article.
- [12] Anton Mazurenko, Christie S. Chiu, Geoffrey Ji, Maxwell F. Parsons, Márton Kanász-Nagy, Richard Schmidt, Fabian Grusdt, Eugene Demler, Daniel Greif, and Markus Greiner. A cold-atom fermi–hubbard antiferromagnet. *Nature*, 545(7655):462–466, May 2017.
- [13] D. Ballester, G. Romero, J. J. García-Ripoll, F. Deppe, and E. Solano. Quantum simulation of the ultrastrong-coupling dynamics in circuit quantum electrodynamics. *Phys. Rev. X*, 2:021007, May 2012.
- [14] E. Manousakis. A quantum-dot array as model for copper-oxide superconductors: A dedicated quantum simulator for the many-fermion problem. *Journal of Low Temperature Physics*, 126(5):1501–1513, Mar 2002.
- [15] A A Clerk, F Marquardt, and K Jacobs. Back-action evasion and squeezing of a mechanical resonator using a cavity detector. *New Journal of Physics*, 10(9):095010, 2008.

- [16] C. W. Gardiner and P. Zoller. *Quantum Noise*. Springer-Verlag Berlin Heidelberg, 2004.
- [17] Iris Schwenk, Jan-Michael Reiner, Sebastian Zanker, Lin Tian, Juha Leppäkangas, and Michael Marthaler. Reconstructing the ideal results of a perturbed analog quantum simulator. *Phys. Rev. A*, 97:042310, Apr 2018.
- [18] Liang-Hui Du, J. Q. You, and Lin Tian. Superconducting circuit probe for analog quantum simulators. *Phys. Rev. A*, 92:012330, Jul 2015.
- [19] Wei Chen and Aashish A. Clerk. Photon propagation in a one-dimensional optomechanical lattice. *Phys. Rev. A*, 89:033854, Mar 2014.
- [20] L Magazzù and M Grifoni. Transmission spectra of an ultrastrongly coupled qubit-dissipative resonator system. *Journal of Statistical Mechanics: Theory and Experiment*, 2019(10):104002, oct 2019.
- [21] L. Tian, I. Schwenk, and M. Marthaler. Dector readout of analog quantum simulators. *arXiv:1612.07419*, 2016.
- [22] Sal J. Bosman, Mario F. Gely, Vibhor Singh, Alessandro Bruno, Daniel Bothner, and Gary A. Steele. Multi-mode ultra-strong coupling in circuit quantum electrodynamics. *npj Quantum Information*, 3(1):46, 2017.
- [23] Sigmund Kohler. Dispersive readout: Universal theory beyond the rotating-wave approximation. *Phys. Rev. A*, 98:023849, Aug 2018.
- [24] D. I. Schuster, A. Wallraff, A. Blais, L. Frunzio, R.-S. Huang, J. Majer, S. M. Girvin, and R. J. Schoelkopf. ac stark shift and dephasing of a superconducting qubit strongly coupled to a cavity field. *Phys. Rev. Lett.*, 94:123602, Mar 2005.
- [25] A. Ridolfo, M. Leib, S. Savasta, and M. J. Hartmann. Photon blockade in the ultrastrong coupling regime. *Phys. Rev. Lett.*, 109:193602, Nov 2012.
- [26] P. Forn-Díaz, J. ?. J. García-Ripoll, B. Peropadre, J.-L. Orgiazzi, M. ?. A. Yurtalan, R. Belyansky, C. ?. M. Wilson, and A. Lupascu. Ultrastrong coupling of

- a single artificial atom to an electromagnetic continuum in the nonperturbative regime. *Nature Physics*, 13:39 EP –, Oct 2016.
- [27] Luigi Garziano, Alessandro Ridolfo, Simone De Liberato, and Salvatore Savasta. Cavity qed in the ultrastrong coupling regime: Photon bunching from the emission of individual dressed qubits. *ACS Photonics*, 4(9):2345–2351, 2017.
- [28] G. Romero, D. Ballester, Y. M. Wang, V. Scarani, and E. Solano. Ultrafast quantum gates in circuit qed. *Phys. Rev. Lett.*, 108:120501, Mar 2012.
- [29] Alexander L Fetter. *Quantum theory of many-particle systems*. Dover Publications, 2003.
- [30] Gerald D. Mahan. *Many-Particle Physics*. Springer US, 2000.
- [31] Richard D. Mattuck. *A Guide to Feynman Diagrams in the Many-Body Problem: Second Edition*. Dover Publications, 1992.
- [32] R.D. Mattuck and Börje Johansson. Quantum field theory of phase transitions in fermi systems. *Advances in Physics*, 17(68):509–562, 1968.
- [33] M O. C. Pires. Generalized wick’s theorem at finite temperature for a quadratic hamiltonian. 08 2012.
- [34] Yung-Li Wang, S. Shtrikman, and Herbert Callen. Wick’s theorem for spin-one-half operators, with an application to spin waves in antiferromagnets. *Phys. Rev.*, 148:419–432, Aug 1966.
- [35] T.S Evans and D.A Steer. Wick’s theorem at finite temperature. *Nuclear Physics B*, 474(2):481 – 496, 1996.
- [36] Thorsten Ohl. Drawing feynman diagrams with fx340-1 and metafont. *Computer Physics Communications*, 90(2):340 – 354, 1995.
- [37] Joseph W. Britton, Brian C. Sawyer, Adam C. Keith, C.-C. Joseph Wang, James K. Freericks, Hermann Uys, Michael J. Biercuk, and John J. Bollinger.

- Engineered two-dimensional ising interactions in a trapped-ion quantum simulator with hundreds of spins. *Nature*, Apr 2012.
- [38] D. Porras and J. I. Cirac. Effective quantum spin systems with trapped ions. *Phys. Rev. Lett.*, 92:207901, May 2004.
- [39] Peter L. McMahon, Alireza Marandi, Yoshitaka Haribara, Ryan Hamerly, Carsten Langrock, Shuhei Tamate, Takahiro Inagaki, Hiroki Takesue, Shoko Utsunomiya, Kazuyuki Aihara, Robert L. Byer, M. M. Fejer, Hideo Mabuchi, and Yoshihisa Yamamoto. A fully programmable 100-spin coherent ising machine with all-to-all connections. *Science*, 354(6312):614–617, 2016.
- [40] K. Kim, M.-S. Chang, S. Korenblit, R. Islam, E. E. Edwards, J. K. Freericks, G.-D. Lin, L.-M. Duan, and C. Monroe. Quantum simulation of frustrated ising spins with trapped ions. *Nature*, 465(7298):590–593, Jun 2010.
- [41] Nicholas Walker, Ka-Ming Tam, and Mark Jarrell. Deep learning on the 2-dimensional ising model to extract the crossover region with a variational autoencoder. *Scientific Reports*, 10(1):13047, Aug 2020.
- [42] S. Sachdev. *Quantum Phase Transitions*. Cambridge University Press, 2011.
- [43] Alexander Shnirman, Gerd Schön, and Ziv Hermon. Quantum manipulations of small josephson junctions. *Phys. Rev. Lett.*, 79:2371–2374, Sep 1997.
- [44] Jens Koch, Terri M. Yu, Jay Gambetta, A. A. Houck, D. I. Schuster, J. Majer, Alexandre Blais, M. H. Devoret, S. M. Girvin, and R. J. Schoelkopf. Charge-insensitive qubit design derived from the cooper pair box. *Phys. Rev. A*, 76:042319, Oct 2007.
- [45] J. E. Mooij, T. P. Orlando, L. Levitov, Lin Tian, Caspar H. van der Wal, and Seth Lloyd. Josephson persistent-current qubit. *Science*, 285(5430):1036–1039, 1999.

- [46] Alexandre Blais, Ren-Shou Huang, Andreas Wallraff, S. M. Girvin, and R. J. Schoelkopf. Cavity quantum electrodynamics for superconducting electrical circuits: An architecture for quantum computation. *Phys. Rev. A*, 69:062320, Jun 2004.
- [47] A. Wallraff, D. I. Schuster, A. Blais, L. Frunzio, R.-S. Huang, J. Majer, S. Kumar, S. M. Girvin, and R. J. Schoelkopf. Strong coupling of a single photon to a superconducting qubit using circuit quantum electrodynamics. *Nature*, 431(7005):162–167, 2004.
- [48] Ravinder R. Puri. *Mathematical Methods of Quantum Optics*. Springer-Verlag Berlin Heidelberg, 2001.
- [49] Javier Puertas Martínez, Sébastien Léger, Nicolas Gheeraert, Rémy Dassonneville, Luca Planat, Farshad Foroughi, Yuriy Krupko, Olivier Buisson, Cécile Naud, Wiebke Hasch-Guichard, Serge Florens, Izak Snyman, and Nicolas Roch. A tunable josephson platform to explore many-body quantum optics in circuit-qed. *npj Quantum Information*, 5(1):19, 2019.
- [50] J. Casanova, G. Romero, I. Lizuain, J. J. García-Ripoll, and E. Solano. Deep strong coupling regime of the jaynes-cummings model. *Phys. Rev. Lett.*, 105:263603, Dec 2010.
- [51] Simone De Liberato. Light-matter decoupling in the deep strong coupling regime: The breakdown of the purcell effect. *Phys. Rev. Lett.*, 112:016401, Jan 2014.
- [52] N. K. Langford, R. Sagastizabal, M. Kounalakis, C. Dickel, A. Bruno, F. Luthi, D. J. Thoen, A. Endo, and L. DiCarlo. Experimentally simulating the dynamics of quantum light and matter at deep-strong coupling. *Nature Communications*, 8(1):1715, Nov 2017.
- [53] Th. Richter and W. Vogel. Nonclassicality of quantum states: A hierarchy of observable conditions. *Phys. Rev. Lett.*, 89:283601, Dec 2002.

- [54] Werner Vogel. Nonclassical states: An observable criterion. *Phys. Rev. Lett.*, 84:1849–1852, Feb 2000.
- [55] P. Forn-Díaz, J. Lisenfeld, D. Marcos, J. J. García-Ripoll, E. Solano, C. J. P. M. Harmans, and J. E. Mooij. Observation of the bloch-siegert shift in a qubit-oscillator system in the ultrastrong coupling regime. *Phys. Rev. Lett.*, 105:237001, Nov 2010.
- [56] L. Teuber, P. Grünwald, and W. Vogel. Nonclassical light from an incoherently pumped quantum dot in a microcavity. *Phys. Rev. A*, 92:053857, Nov 2015.
- [57] Weber Arfken. *Mathematical Methods for Physicists*. Elsevier, 2013.
- [58] A Yu Kitaev. Unpaired majorana fermions in quantum wires. *Physics-Uspokhi*, 44(10S):131–136, oct 2001.
- [59] Jianlong Fu, Johannes Knolle, and Natalia B. Perkins. Three types of representation of spin in terms of majorana fermions and an alternative solution of the kitaev honeycomb model. *Phys. Rev. B*, 97:115142, Mar 2018.
- [60] B. Chang and J. Paaske. Majorana fermion representations of magnetic impurity problems. 2013.
- [61] W. Mao, P. Coleman, C. Hooley, and D. Langreth. Spin dynamics from majorana fermions. *Phys. Rev. Lett.*, 91:207203, Nov 2003.
- [62] H. J. Vidberg and J. W. Serene. Solving the eliashberg equations by means ofn-point padé approximants. *Journal of Low Temperature Physics*, 29(3):179–192, Nov 1977.
- [63] Peter Staar, Bart Ydens, Anton Kozhevnikov, Jean-Pierre Locquet, and Thomas Schulthess. Continuous-pole-expansion method to obtain spectra of electronic lattice models. *Phys. Rev. B*, 89:245114, Jun 2014.
- [64] E.G. Klepfish, C.E. Creffield, and E.R. Pike. Analytic continuation of matsubara green’s functions. *Nuclear Physics B - Proceedings Supplements*, 63(1):655 – 657, 1998. Proceedings of the XVth International Symposium on Lattice Field Theory.

- [65] O. Gunnarsson, M. W. Haverkort, and G. Sangiovanni. Analytical continuation of imaginary axis data using maximum entropy. *Phys. Rev. B*, 81:155107, Apr 2010.
- [66] Xing-Jie Han, Hai-Jun Liao, Hai-Dong Xie, Rui-Zhen Huang, Zi-Yang Meng, and Tao Xiang. Analytic continuation with padé decomposition. *Chinese Physics Letters*, 34(7):077102, jul 2017.
- [67] A. J. Leggett, S. Chakravarty, A. T. Dorsey, Matthew P. A. Fisher, Anupam Garg, and W. Zwerger. Dynamics of the dissipative two-state system. *Rev. Mod. Phys.*, 59:1–85, Jan 1987.
- [68] G. Ithier, E. Collin, P. Joyez, P. J. Meeson, D. Vion, D. Esteve, F. Chiarello, A. Shnirman, Y. Makhlin, J. Schrieffer, and G. Schön. Decoherence in a superconducting quantum bit circuit. *Phys. Rev. B*, 72:134519, Oct 2005.
- [69] Charles R. Harris, K. Jarrod Millman, St’efan J. van der Walt, Ralf Gommers, Pauli Virtanen, David Cournapeau, Eric Wieser, Julian Taylor, Sebastian Berg, Nathaniel J. Smith, Robert Kern, Matti Picus, Stephan Hoyer, Marten H. van Kerkwijk, Matthew Brett, Allan Haldane, Jaime Fern’andez del R’io, Mark Wiebe, Pearu Peterson, Pierre G’erard-Marchant, Kevin Sheppard, Tyler Reddy, Warren Weckesser, Hameer Abbasi, Christoph Gohlke, and Travis E. Oliphant. Array programming with NumPy. *Nature*, 585(7825):357–362, September 2020.
- [70] Aaron Meurer, Christopher P. Smith, Mateusz Paprocki, Ondřej Čertík, Sergey B. Kirpichev, Matthew Rocklin, AMiT Kumar, Sergiu Ivanov, Jason K. Moore, Sartaj Singh, Thilina Rathnayake, Sean Vig, Brian E. Granger, Richard P. Muller, Francesco Bonazzi, Harsh Gupta, Shivam Vats, Fredrik Johansson, Fabian Pedregosa, Matthew J. Curry, Andy R. Terrel, Štěpán Roučka, Ashutosh Saboo, Isuru Fernando, Sumith Kulal, Robert Cimrman, and Anthony Scopatz. Sympy: symbolic computing in python. *PeerJ Computer Science*, 3:e103, January 2017.