SANDIA REPORT

SAND2019-0739
Unlimited Release
Printed December 2018

Quantum Optimization and Approximation Algorithms

Ojas Parekh, Ciaran Ryan-Anderson, and Sevag Gharibian

Prepared by
Sandia National Laboratories
Albuquerque, New Mexico 87185 and Livermore, California 94550

Sandia National Laboratories is a multimission laboratory managed and operated by National Technology and Engineering Solutions of Sandia, LLC, a wholly owned subsidiary of Honeywell International, Inc., for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA0003525.

Sandia National Laboratories

Issued by Sandia National Laboratories, operated for the United States Department of Energy by National Technology and Engineering Solutions of Sandia, LLC.

Available to DOE and DOE contractors from
U.S. Department of Energy
Office of Scientific and Technical Information
P.O. Box 62
Oak Ridge, TN 37831

Telephone: (865) 576-8401
Facsimile: (865) 576-5728
E-Mail: reports@adonis.osti.gov
Online ordering: http://www.osti.gov/bridge

Available to the public from
U.S. Department of Commerce
National Technical Information Service
5285 Port Royal Rd
Springfield, VA 22161

Telephone: (800) 553-6847
Facsimile: (703) 605-6900
E-Mail: orders@ntis.fedworld.gov
Online ordering: http://www.ntis.gov/help/ordermethods.asp?loc=7-4-0#online

SAND2019-0739
Unlimited Release
Printed December 2018

Quantum Optimization and Approximation Algorithms

Ojas Parekh
Discrete Math and Optimization
Sandia National Laboratories
P.O. Box 5800
Albuquerque, NM 87185-1326
odparek@sandia.gov

Ciaran Ryan-Anderson
Department of Physics
Swansea University
Swansea, SA2 8PP
United Kingdom

Sevan Gharibian
Department of Computer Science
Universität Paderborn
33098 Paderborn
Germany

Abstract

Shor's groundbreaking quantum algorithm for integer factoring provides an exponential speedup over the best-known classical algorithms. In the 20 years since Shor's algorithm was conceived, only a handful of fundamental quantum algorithmic kernels, generally providing modest polynomial speedups over classical algorithms, have been invented. To better understand the potential advantage quantum resources provide over their classical counterparts, one may consider other resources than execution time of algorithms. Quantum Approximation Algorithms direct the power of quantum computing towards optimization problems where quantum resources provide higher-quality solutions instead of faster execution times. We provide a new rigorous analysis of the recent Quantum Approximate Optimization Algorithm, demonstrating that it provably outperforms the best known classical approximation algorithm for special hard cases of the fundamental Maximum Cut graph-partitioning problem. We also develop new types of classical approximation algorithms

for finding near-optimal low-energy states of physical systems arising in condensed matter by extending seminal discrete optimization techniques. Our interdisciplinary work seeks to unearth new connections between discrete optimization and quantum information science.

Contents

Appendix

List of Figures

Nomenclature

AQC Adiabatic quantum computing

CSP Constraint satisfaction problem

***k*-LH** *k*-Local Hamiltonian problem

Max-Cut Maximum Cut problem

Max-SAT Maximum Boolean Satisfiability problem

NISQ Noisy Intermediate-Scale Quantum

OPT Objective value of an optimal solution to a discrete optimization problem

QAA Quantum approximation algorithm

QAOA Quantum Approximate Optimization Algorithm

This page intentionally left blank.

Chapter 1

Introduction

Approximation algorithms and constraint satisfaction problems (CSPs) are central topics in our work. To motivate both, we briefly discuss approximation algorithms in the context of the Maximum Boolean satisfiability problem (Max-SAT). The Boolean satisfiability problem (SAT) has become an essential ingredient in the analysis and verification of trustworthy and resilient software and hardware systems; applications include vulnerability discovery, program synthesis, and property checking. SAT seeks to find a satisfying assignment to the variables of a Boolean formula[1] and is a prototypical example of a computationally hard, NP-complete, problem. Many heuristics for SAT are known, but for certain types of mission-driven applications, one may desire a guarantee on the quality of solutions generated by the algorithm, which heuristics do not address. One such quality measure is the total fraction of clauses satisfied; however, this is a poor measure when few or no clauses are satisfiable. A better measure is the maximum possible fraction of clauses satisfied by any assignment, and we call this quantity OPT, which is shorthand for "optimal." An α-approximation algorithm is an algorithm that (1) runs in time polynomial in the size of its input, and (2) is guaranteed to always produce a solution whose quality is at least $\alpha \cdot OPT$, for some $\alpha \in (0,1]$. Although motivated by the maximization version of SAT (Max-SAT), this definition also applies naturally to other discrete optimization problems. Thus the approximation guarantee, α of an approximation algorithm indicates the quality of solution it produces relative to the best possible.

In order to design an approximation algorithm, one needs an estimation of OPT, yet for most hard discrete optimization problems, computing OPT is just as hard as actually finding an optimal solution. Thus we turn to a relaxation of the problem, in which some of the underlying constraints are relaxed at the benefit of yielding a problem that is easier to solve. Integer linear programming (ILP), which optimizes over integer-valued variables that satisfy a set of linear inequalities, has proven a consistent source of high-quality relaxations for discrete optimization problems. Max-SAT is readily captured by an ILP formulation, and we may obtain a relaxation by simply allowing variables to take rational values instead of only integer values.

For an example of Max-SAT, consider the following 4 SAT clauses on two variables: $(x_1 \vee x_2)$, $(x_1 \vee \neg x_2)$, $(\neg x_1 \vee x_2)$, and $(\neg x_1 \vee \neg x_2)$. These clauses were chosen as en example where it is impossible to satisfy all of them; however, any 3 of the clauses may be satisfied. We can represent

[1] An example of such a formula is, $(x_1 \vee \neg x_2) \wedge (\neg x_1 \vee x_2 \vee x_3) \wedge (\neg x_1)$, where $\vee$, $\wedge$, and $\neg$ are the logical OR, AND, and NOT operations, respectively, and parentheses denote clauses. This formula may be satisfied by setting $x_1 = x_2 = FALSE$ and x_3 arbitrarily.

these 4 clauses as linear constraints over variables that takes values either 0 or 1: $x_1 + x_2 \geq 1$, $x_1 + (1 - x_2) \geq 1$, $(1 - x_1) + x_2 \geq 1$, and $(1 - x_1) + (1 - x_2) \geq 1$. Analogously to the Boolean case above, if $x_1, x_2 \in \{0, 1\}$, then it is impossible to satisfy all 4 of these inequalities simultaneously. However, if we relax the domain of the variables so that $x_1, x_2 \in [0, 1]$ are now real numbers, instead of just 0 or 1, then we can satisfy all 4 inequalities simply by setting $x_1 = x_2 = \frac{1}{2}$.

Relaxing the integrality constraint on the variables above allows the relaxed problem above to satisfy 4 inequalities rather than 3. Remarkably, linear programming (LP) relaxations of hard integer-linear programs can be solved in polynomial time. Since a relaxed problem is less constrained than the original, an optimal relaxed solution is guaranteed to have quality at least that of an optimal feasible solution. However, a relaxed solution does not directly yield a feasible solution to the original problem; e.g., how does one interpret $\frac{1}{2}$ as Boolean value? An algorithm that converts an optimal relaxed solution, of quality OPT_R, into a true feasible solution is called a rounding algorithm. One may view rounding as a tradeoff between quality and feasibility. In fact, if a rounding algorithm is able to always produce a feasible solution of quality at least $\alpha \cdot OPT_R$, then we have an α-approximation algorithm, since $OPT_R \geq OPT$ and $\alpha \cdot OPT_R \geq \alpha \cdot OPT$. Although the term rounding stems from relaxations that drop integrality constraints, we will consider quite general relaxations and procedures for converting relaxed solutions into feasible solutions. We must also have a strong relaxation, one for which OPT_R is close to OPT. If our relaxation is weak, we will be unable to prove that our approximation algorithm performs well.

Our goal is to generalize the above type of framework to design new types of approximation algorithms in the quantum setting. By this we mean quantum approximation algorithms for classical discrete optimizations, as well as approximation algorithms for optimization problems that naturally arise in quantum physics. For the latter we will exploit connections between such problems and well-studied discrete optimization problems.

1.1 Quantum Optimization

Approximating *ground states* (i.e., lowest energy states) of physical systems underlie many applications in physics and chemistry. The feasible energy levels of a physical system are prescribed by the eigenvalues of a linear operator called a *Hamiltonian*. A Hamiltonian serves as a model for a physical system, and understanding the eigenspectrum of a Hamiltonian can help ascertain critical properties and predict behavior of the underlying physical system. Since low energy states tend to be more stable, nature routinely heuristically solves optimization problems by driving physical system towards low energy states. This provides an intrinsic and natural connection between quantum physics and discrete optimization: the energy levels of a finite quantum system may typically only take a discrete number of values, hence finding a ground state *is* a discrete optimization problem. The challenge in solving such problem is that the number of energy levels of physically relevant quantum systems is typically exponential in the number of parameters used to describe it. Chapter 2 and Appendix A outline this relationship between discrete optimization and quantum physics in more detail; the latter is a general introduction to quantum computing.

Our work aims to exploit this connection between discrete optimization and quantum physics in both directions: we seek to design and rigorously analyze (1) quantum algorithms that in effect approximately solve discrete optimization problems by trying to attain the ground state of quantum system, and (2) algorithms leveraging discrete optimization techniques to approximately find ground states of quantum systems.

1.2 Key Results

The Quantum Approximate Optimization Algorithm (QAOA) is a recently proposed quantum algorithm to approximately solve classical discrete optimization problems. QAOA is in some sense a discrete counterpart to the celebrated Quantum Adiabatic Algorithm, and enjoys many properties suggesting that it may be amenable to implementation on near-term to intermediate-term quantum computers. As a heuristic QAOA is well motivated, yet rigorous analysis and comparison with classical optimization techniques has proven challenging. We offer a new analysis of QAOA for a fundamental discrete optimization problem, Max-Cut. Our analysis precisely characterizes the behavior of QAOA with a concise mathematical formula. It was somewhat surprising that such a characterization is indeed possible. We use this formula to rigorously demonstrate that QAOA outperforms the best-known classical algorithm for a hard special case of Max-Cut. Chapter 2 describes QAOA and our results.

The Heisenberg model, $\sum_{<i,j>} \sigma_i^x \sigma_j^x + \sigma_i^y \sigma_j^y + \sigma_i^z \sigma_j^z$, is fundamental for describing quantum magnetism, superconductivity, and charge density waves. Beyond one dimension, the properties of the anti-ferromagnetic Heisenberg model are notoriously difficult to analyze. By generalizing a connection between the Ising model of magnetism and the fundamental discrete optimization problem, Max-Cut, we propose a new variant of the Heisenberg model, $\sum_{<i,j>} I - \sigma_i^x \sigma_j^x - \sigma_i^y \sigma_j^y - \sigma_i^z \sigma_j^z$. Finding a maximum energy eigenstate of our model is equivalent to finding the ground state of the aforementioned anti-ferromagnetic Heisenberg model; however, we demonstrate that the new model is easier to approximate in a rigorous way. We demonstrate a *classical* algorithm that is guaranteed to produce a classical product state whose energy is within a factor of 0.498 of that of the optimal *quantum* state. Thus although our algorithm is classical, its performance is measured against the best possible quantum state. Moreover, we show that there are instances where the best possible product state has at most $\frac{1}{2}$ the energy of the best possible quantum state, hence our result is essentially the best possible. Our techniques generalize to a much broader class of 2-local Hamiltonians. Our results rely on generalizations of powerful classical discrete optimization techniques such as semidefinite relaxations and hyperplane rounding. We view this as the initiation of a body of work that seeks to provide classical and quantum algorithms producing approximately optimal eigenstates with rigorous guarantees on their performance. Chapter 3 details these results.

This page intentionally left blank.

Chapter 2

An Analysis of the Quantum Approximate Optimization Algorithm

This chapter appears in the Ph.D. thesis of Ciaran Ryan-Anderson [RA18], who was employed at Sandia National Labs when this work was completed, jointly with Ojas Parekh.

Currently, large-scale, fault-tolerant, universally-programmable quantum-computers do not exist. It is, therefore, useful to consider what algorithms one potentially wants to run on noisy intermediate-scale quantum (NISQ) devices [Pre18] when designing architectures. These algorithms inform the architect about the type of interactions and connectivity that are needed and how difficult it may be to map the desired applications to the constraints of the hardware. Further analysis of the target algorithms and hardware implementation may also reveal the type and degree of noise the computation must be protected against to obtain satisfactory outputs.

The focus of this chapter is on evaluating a quantum algorithm that is of potential interest. More specifically, this chapter presents an analysis of the performance of the Quantum Approximate Optimization Algorithm (QAOA) applied to the Max-Cut problem. QAOA was designed to find approximate solutions to classical constraint satisfaction problems (CSPs). QAOA is an algorithm of interest for NISQ hardware since QAOA provides a natural tradeoff between quality of solution and circuit depth. Here we argue that QAOA outperforms the currently known best classical approximation algorithm for the Max-Cut problem on certain restricted graphs.

2.1 Introduction

QAOA was originally defined by Farhi, Goldstone, and Gutmann in 2014 [FGG14b] and is the first known quantum approximation algorithm for classical constraint satisfaction problems (CSPs). Specifically, QAOA is a quantum-algorithmic framework for finding approximate solutions to discrete optimization problems [FGG14b; FGG14a]. QAOA may be viewed as a discretized simulation of adiabatic quantum computation (AQC) and, as with AQC, is a universal model of quantum computation. Farhi *et al.* demonstrated worst-case bounds on the performance of QAOA for Max-Cut in 3-regular graphs [FGG14b] and Max-3-XOR [FGG14a] (*i.e.*, each clause is the XOR of 3 Boolean literals). QAOA has since been generalized by others [JRW17; Had+17; MW18].

In this work, we explore the performance of QAOA and present a closed-form expression for the expectation of QAOA on Max-Cut for any graph instance. Using this result, we show that QAOA for Max-Cut on k-regular triangle-free graphs outperforms the currently best known classical approximation algorithms.

This chapter is organized as follows: In Section 2.2, we present background on the QAOA algorithm and the Max-Cut problem. In Section 2.3, we discuss the previous results of QAOA on the Max-Cut problem. In Section 2.4, we make a brief comment on the difference between sampling and optimization. In Section 2.5, we discuss the specific results of this work as well as give an overview of the techniques used to derive the results. In Section 2.6, we give the proof of our results. Finally, in Section 2.7 we conclude.

2.2 Background

In this section, we will introduce CSPs and discuss how one may arrive at QAOA from an approximate simulation of AQC. We focus on the version of QAOA first described by Farhi *et al.* in [FGG14b].

2.2.1 Constraint Satisfaction Problems

Following the definition of a CSPs as defined in other works on QAOA (*e.g.*, see [FGG14b] and [FH16]), we now define the constraint satisfaction problem confined to Boolean function constraints.[1] A CSP is specified by n, the size of bit strings to be considered, and a set $\{C_\alpha(z)\}$ of m clauses (also known as constraints). Each clause is of the form

$$C_\alpha(z) = \begin{cases} 1 & \text{if } z \text{ satisfies the clause} \\ 0 & \text{otherwise,} \end{cases} \tag{2.1}$$

where for QAOA we let $z \in \{+1,-1\}^n$ (the eigenvalues of Pauli Z). Typically, clauses only evaluate a few bits; therefore, we will restrict all $C_\alpha(z)$ to evaluate the satisfiability of at most k bits, where k is some fixed integer. In other words, each clause is k-local.

A CSP is solved when a bit string z is found that satisfies as many clauses as possible. To accomplish this goal, we can define an objective function

$$C(z) = \sum_\alpha C_\alpha(z) \tag{2.2}$$

[1] One can define CSPs more generally, *e.g.*, see [LZ16].

and maximize $C(z)$ for all z. Since $C(z)$ merely counts the number of satisfied clauses, a z that maximizes $C(z)$ is a solution to the CSP.

The Maximum Cut problem (Max-Cut) is an example of a CSP and can be defined as follows: Given a set of vertices V and a set of edges E between the vertices in V, Max-Cut on a simple graph $G = (V, E)$ is the problem of finding a bi-partition of V that maximizes the number of edges that run between the two partitions. Note that the edge between vertices i and j will be indicated by ij. Since we will only consider undirected graphs, ij is equivalent to the set $\{i, j\}$. If we label one partition 0 and the other 1, then one choice of objective function for Max-Cut is

$$C(z) = \sum_{ij \in E} C_{ij}(z), \tag{2.3}$$

where

$$C_{ij}(z) = \frac{1}{2}(1 - z_i z_j). \tag{2.4}$$

Here the value of bit z_u indicates which partition vertex u belongs to. On inspection, we see that for an edge ij the clause $C_{ij}(z)$ assigns the value 1 if $z_i \neq z_j$ and 0 if $z_i = z_j$. Thus, $C(z)$ is maximized when the number of edges whose vertices are not in the same partition is maximized.

2.2.2 Adiabatic Quantum Computation

QAOA can be described as a discrete approximation of AQC. Arguably, understanding the connection between AQC and QAOA provides useful insight into QAOA. Therefore, in this section we will briefly review AQC.

AQC is an application of the adiabatic approximation for the purposes of quantum computation. Loosely speaking, the adiabatic approximation says that if a state is an instantaneous eigenstate of a time-dependent Hamiltonian, then the state will remain in the instantaneous eigenstate of the Hamiltonian as long as the evolution of the Hamiltonian is slow enough and there is always an energy gap between the instanteous eigenstate and the rest of eigenstates [BF28]. There are numerous proofs of varying rigor and refinements of the adiabatic approximation. Many of these proofs as well as a review of AQC are given in [AL16] by Albash and Lidar.

AQC is defined by two time-independent Hamiltonians H_D (the "driver Hamiltonian") and H_P (the "problem Hamiltonian"). These Hamiltonians are chosen to be k-local and, therefore, can be written as

$$H = \sum_{\substack{K \subseteq [n] \\ |K| \leq k}} H_K, \tag{2.5}$$

where $[n] \equiv \{1, 2, \cdots, n\}$ and each H_K only acts on the qubits indexed by the set K.

The Hamiltonian H_D is the initial Hamiltonian of the computation and is chosen so that the ground state of H_D is easy to prepare. The Hamiltonian H_P is the final Hamiltonian and is chosen so that the ground state of H_P encodes the solution to a problem of interest.

Inspired by the adiabatic approximation, AQC is performed by initializing in the (not necessarily unique) ground state of H_D and adiabatically interpolating to H_P. The state then remains in the ground state throughout the evolution with high probability. After the evolution, the final state is measured in the computational basis to determine the encoded solution as defined by H_P.

The interpolation is given by time-dependent Hamiltonian

$$H(t) = (1 - s(t))H_D + s(t)H_P, \tag{2.6}$$

where $s(t)$ is a smooth function of time t such that $s(t = 0) = 0$ and $s(t = T) = 1$. Here T is the time interval during which the interpolation between the driver and problem Hamiltonian occurs. The function $s(t)$ is often referred to as the *schedule* of the computation.

Note, assuming an AQC interpolation $H(t)$ given in Eq. 2.6 between two time-independent Hamiltonians H_D and H_P, a necessary condition for a finite gap to exist during a non-trivial computation is that H_D and H_P must not commute [Woo18]. As described by the simultaneous diagonalization theorem (see Theorem 2.2 of [NC11]), if $[H_D, H_P] = 0$ and both H_D and H_P act non-trivially over an entire joint Hilbert space, then H_D and H_P share the same set of energy eigenvectors. Therefore, assuming the two Hamiltonians are not the same, H_D and H_P assign different energy eigenvalues to at least some of the energy eigenvectors. Assuming the initial ground-state is not the solution to the problem, which would make the computation effectively trivial, the final ground state is not the same as the initial ground state. Since the initial ground-state is an eigenstate of H_D and the final eigenstate is an eigenstate of H_P, then for the computation to be non-trivial, the energy eigenvalues of the two states must cross during the computation. Thus, there is no gap if $[H_D, H_P] = 0$ and the computation is non-trivial.

Different formulations of the adiabatic theorem place different constraints on the schedule $s(t)$ such as the derivatives of the function. To meet the assumptions of the adiabatic approximation, T is often chosen to be $T = \mathcal{O}(1/g_{\min}^2)$, where $g_{\min}$ is the minimum energy gap between the ground state and the first excited-state, during the schedule for times on this order, the Euclidean norm between the actual final ground-state and the ideal ground-state of H_P (the "error") can be made arbitrarily small [AL16]. Selecting a version of the adiabatic theorem in order to specify the constraints on $s(t)$ and T that will ensure a target error is unnecessary to understand the connection between AQC and QAOA. For a discussion on AQC's robustness to experimental error, see [CFP01].

AQC has been proven to be a quantum computation model that is universal [Aha+04]. A version of AQC that is restricted to solving optimization problems is known as the quantum adiabatic algorithm (QAA or, alternatively, QADI—for **Q**uantum **ADI**abatic algorithm) [Far+00; Far+01].

We will now consider QAA.

Given a CSP, it is relatively simple to construct a problem Hamiltonian. The a clause $C_\alpha(z)$ as defined by Eq. 2.2 can be described by a matrix

$$C_\alpha \equiv \sum_{z\in\{0,1\}^n} C_\alpha(z)\,|z\rangle\langle z|\,, \tag{2.7}$$

in the computational basis.

Restricting each clause $C_\alpha(z)$ to act on at most k bits and recalling that $|0\rangle\langle 0| = \frac{1}{2}(I+Z)$ and $|1\rangle\langle 1| = \frac{1}{2}(I-Z)$, we can re-express Eq. 2.7 as

$$C_\alpha = \sum_{\substack{K\subseteq[n]\\ |K|\leq k}} W_{\alpha,K} Z^K, \tag{2.8}$$

where $W_{\alpha,K}\in\mathbb{R}$ and we define

$$M^K \equiv \bigotimes_{j\in K} M_j. \tag{2.9}$$

Here M_j is a 2x2 matrix acting on qubit j and it is assumed that M^K acts as I on any qubit not indexed by K. Also, we define $M^\varnothing \equiv I$.

Each clause $C_\alpha(z)$ can be considered a Boolean function $f:\{0,1\}^n\to\{0,1\}$. The conversion of Boolean functions to operators of the form defined in Eq. 2.8 is described in [Bia08] and [Had18].

From Eq. 2.2, Eq. 2.8, and Eq. 2.7, it follows that the matrix or operator form of the objective function in the computational basis is

$$C \equiv \sum_\alpha C_\alpha = \sum_{z\in\{0,1\}^n} C(z)\,|z\rangle\langle z| = \sum_{\substack{K\subseteq[n]\\ |K|\leq k}} W_K Z^K, \tag{2.10}$$

where $W_K = \sum_\alpha W_{\alpha,K}$ and $C(z)$ is given in Eq. 2.2.

Since

$$C^\dagger = \sum_{z\in\{0,1\}^n} C(z)^*\,|z\rangle\langle z|^\dagger = \sum_{z\in\{0,1\}^n} C(z)\,|z\rangle\langle z| = C, \tag{2.11}$$

C is Hermitian and could be identified as a Hamiltonian. However, solutions to the CSP defined

by C corresponds to eigenvectors with the maximum eigenvalue. Since AQC is normally defined to track the ground state, for QAA we take

$$H_P = -C = -\sum_{\substack{K\subseteq[n]\\|K|\leq k}} W_K Z^K. \tag{2.12}$$

Also, for QAA problems, H_D is taken to be

$$H_D = -B, \tag{2.13}$$

where

$$B \equiv \sum_{j\in[n]} X_j. \tag{2.14}$$

The ground state of H_D, in which we initialize the state of our computation, is

$$|s\rangle \equiv |+\rangle^{\otimes n} = \frac{1}{2^{n/2}} \sum_{z\in\{0,1\}^n} |z\rangle. \tag{2.15}$$

The minimum energy gap between the ground state and first excited state is unknown for general AQC problems; however, since the $H(s)$ as defined by Eq. 2.12 and Eq. 2.13 is stoquastic (a Hamiltonian with off-diagonal elements only in $\mathbb{R}_{\leq 0}$), QAA always has a non-vanishing gap, which is assured by the Perron-Frobenius theorem [Per07; Fro12].

If we consider Max-Cut on a simple graph $G = (V,E)$ as defined in Section 2.2.1, then the matrix form of a clause $C_{ij}(z)$ as described by Eq. 2.4 is given as

$$C_{ij} = |01\rangle\langle 01|_{ij} + |10\rangle\langle 10|_{ij} = \frac{1}{2}(I - Z_i Z_j). \tag{2.16}$$

Thus, C for Max-Cut is

$$C = \sum_{ij\in E} |01\rangle\langle 01|_{ij} + |10\rangle\langle 10|_{ij} = \sum_{ij\in E} \frac{1}{2}(I - Z_i Z_j). \tag{2.17}$$

2.2.3 Discretizing Adiabatic Quantum Computation

The continuous evolution of Hamiltonians can be simulated by sequences of discrete, k-local unitaries for fixed k [Fey82; Llo96]. One algorithm used to find such simulations is commonly known as "Trotterization." It is through the Trotterization of AQC that we present the connection between QAOA and AQC.

We start by considering Trotterization since the AQC Hamiltonian is time-dependent and generally the Hamiltonian will not commute with itself at different points in time, the unitary induced by AQC is Eq. A.23, that is $U(T,0) = \mathscr{T}\exp\{-i\int_0^T H(t)\,dt\}$. Besides being potentially intractable to calculate, this AQC unitary may be difficult to experimentally implement. We can begin to simplify the unitary by first noting that we can break it up as follows:

$$\begin{aligned} U(T,0) &= U(T,T-\Delta t)U(T-\Delta t,T-2\Delta t)\cdots U(\Delta t,0) \\ &= \prod_{j=1}^{p} U(j\Delta t,(j-1)\Delta t), \end{aligned} \tag{2.18}$$

where $p\Delta t = T$.

Then, if we choose Δt to be small enough so that $H(t)$ is approximately constant over the time interval $[(j-1)\Delta t, j\Delta t]$ for all $1 \le j \le p$, we can use the time-independent unitary solution $U(t,t_0) = \exp\{-iH\,(t-t_0)\}$ (Eq. A.28) to approximate the unitaries in Eq. 2.18 as

$$U(j\Delta t,(j-1)\Delta t) \approx e^{-i\,H(j\Delta t)\,\Delta t}. \tag{2.19}$$

Applying Eq. 2.19 to Eq. 2.18, we now write

$$U(T,0) \approx \prod_{j=1}^{p} e^{-i\,H(j\Delta t)\,\Delta t}. \tag{2.20}$$

Note, for a version of Trotterization that accounts for time-dependence, see [Pou+11].

Given Eq. 2.20, We can further simplify the unitary sequence by recalling that the AQC Hamiltonian (Eq. 2.5) has the form $H = \sum_{\substack{K\subseteq[n]\\|K|\leq k}} H_K$ and that the Lie-Trotter product formula [Tro59] is

$$e^{i(A+B)x} = \lim_{p\to\infty}\left(e^{iAx/p}e^{iBx/p}\right)^p, \tag{2.21}$$

where $x \in \mathbb{C}$ and the matrices A and B are real or complex.

Suzuki derived recursive formulas for calculating approximations of Eq. 2.21 [Suz91; HS05], known as Lie-Trotter-Suzuki decompositions. It is common to use the lower order decompositions

$$e^{i(A+B)x} = e^{iAx}e^{iBx} + \mathcal{O}(x^2) \tag{2.22}$$

or

$$e^{i(A+B)x} = e^{iBx/2}e^{iAx}e^{iBx/2} + \mathcal{O}(x^3) \tag{2.23}$$

for simulating Hamiltonians.

Combining Eq. 2.20 and the lower-order approximation of Eq. 2.22, we arrive at

$$U(T,0) \approx \prod_{j=1}^{p}\prod_{\substack{K\subseteq[n]\\|K|\leq k}} e^{-i\,H_K(j\Delta t)\,\Delta t}. \tag{2.24}$$

We now have an approximation of the AQC unitary consisting of unitaries that each act on at most k qubits. If necessary, we might further simplify an instance of Eq. 2.24 by decomposing the unitaries $e^{-i\,H_K(j\Delta t)\,\Delta t}$ into sequences of simpler gates.

Note, for an analysis of the error of Trotterizations described by Eq. 2.24 and higher-order decompositions, see [Wie+08]. For a discussion of the stability of the Lie-Trotter-Suzuki decomposition due to imprecisions in applying unitaries, see [DS14; KM15].

If we now turn to approximating the QAA Hamiltonian defined by the H_P of Eq. 2.12 and the H_D of Eq. 2.13 with the approximate unitary given by Eq. 2.24, we get

$$U(T,0) \approx \prod_{j=1}^{p} \exp\{i\,[1-s(j\Delta t)]\,\Delta t\,B\}\,\exp\{i\,s(j\Delta t)\,\Delta t\,C\}. \tag{2.25}$$

2.2.4 Quantum Approximate Optimization Algorithm

The number of steps p needed to adequately approximate a QAA computation with Eq. 2.25 may be large in general. However, rather than finding an exact solution to a CSP, one may be satisfied with finding a high-quality solution that obtains a large fraction of the maximum number of satisfiable clauses. One may then be curious to what degree one can reduce the quality of a unitary approximation of QAA and still produce adequate results. This question naturally leads to the formulation of QAOA.

Given a CSP that is specified by an objective operator C (defined by Eq. 2.10), QAOA generates states of the form

$$|\gamma,\beta\rangle \equiv U_{\mathrm{QAOA}_p}(\gamma,\beta)|s\rangle, \tag{2.26}$$

where $\gamma \equiv (\gamma_1,\gamma_2,\cdots,\gamma_p)$ and $\beta \equiv (\beta_1,\beta_2,\cdots,\beta_p)$ are vectors of angles, the state $|s\rangle$ is the equal superposition of the computational basis-states as defined in Eq. 2.15, and

$$U_{\mathrm{QAOA}_p}(\gamma,\beta) \equiv \prod_{j=1}^{p} e^{-i\,\beta_j B} e^{-i\,\gamma_j C}. \tag{2.27}$$

Here B is the sum of Pauli Xs as defined in Eq. 2.14, and by QAOA_p we mean the subclass of QAOA where a fixed p specifies the number of $e^{-i\,\beta_j\,B}e^{-i\,\gamma_j\,C}$ products present in U_{QAOA_p}.

We see that if we identify γ_j as $-s(j\Delta t)\,\Delta t$ and β_j as $-[1-s(j\Delta t)]\,\Delta t$, then the QAOA_p unitary Eq. 2.27 is equivalent to the QAA unitary Eq. 2.25 for the same value of p. While the QAA and QAOA unitaries are of similar form and both the angles and p may be chosen so that they are equal, there are two differences.

First, since QAA is an AQC algorithm, QAA must turn the problem of finding a state that maximizes C to finding a ground state that minimizes $-C$. However, QAOA approximately simulates a process that instead initializes in an eigenstate with the *maximum* eigenvalue of the driver Hamiltonian and remains in the instantaneous eigenstate that has the *maximum* eigenvalue of the interpolating Hamiltonian $H(t) = (1-s(t))H_D + s(t)H_P$, where for QAOA we let $H_D = B$ and $H_P = C$. While this is inconsistent with ground-state AQC, the adiabatic theorem merely requires an energy gap between the instantaneous state and the rest of the energy spectrum.

The other distinction to note between the QAA and QAOA unitaries is that for QAOA, typically p is chosen to be some small integer. Currently, $p = 1$ or $p = 2$ is frequently considered.

To compensate for such an extremely crude approximation, Farhi *et al.* [FGG14b] showed that a hybrid classical and quantum gradient-search algorithm can be used to efficiently choose the

optimal values for γ and β given a fixed p, where optimal means obtaining the value

$$M_p = \max_{\gamma,\beta} \langle \gamma, \beta | C | \gamma, \beta \rangle . \tag{2.28}$$

Given a CSP, the goal of QAOA obtain a high approximation ratio

$$\alpha_p \equiv \frac{M_p}{C_{\max}}, \tag{2.29}$$

where

$$C_{\max} \equiv \max_{z \in \{0,1\}^n} C(z). \tag{2.30}$$

Note, for a given class of problems, $C_{\max}$ is not always know *a priori*.

Once good angles are found, the state $|\gamma, \beta\rangle$ is prepared and measured in the computational basis until a measurement output z is found such that $C(z) \geq M_p$.

If desired, the approximation ratio can be increased by raising p to obtain better solutions, since

$$\lim_{p \to \infty} M_p = C_{\max}. \tag{2.31}$$

This is because as p tends to infinity, then U_{QAOA_p} can at the very least simulate QAA and, due to the Perron-Frobenius theorem, QAA will in principle be able to find the solution to the CSP (although T might be *very* large). Further,

$$M_p \geq M_{p-1} \tag{2.32}$$

since $U_{\mathrm{QAOA}_p} = e^{-i\,\beta_p\,B} e^{-i\,\gamma_p\,C} U_{\mathrm{QAOA}_{p-1}}$ implies that finding optimal angles for $p-1$ means that we can at least obtain $M_p = M_{p-1}$ since we can set $\gamma_p, \beta_p = 0$. Therefore, M_p monotonically increases.

QAOA then provides a natural trade-off between depth of the resulting quantum circuit and the quality of solution produced. While we may increase p to produce better solutions, precise analysis of M_p versus p poses a challenge. However, Wang *et al.* [Wan+17] obtained an analytic expression for QAOA_p for Max-Cut on a simple cycle.

2.3 Previous Work

Previously, Farhi *et al.*'s analysis of QAOA_1 for Max-3-XOR in 2014 showed that QAOA_1 is expected to satisfy at least $\left(\frac{1}{2}+\frac{\mathscr{O}(1)}{d^{3/4}}\right)m$ clauses on instances with m clauses where each variable occurs in at most d of them. This surprising quantum result improved upon a longstanding classical approximation algorithm by Håstad [Hås00] from 2000, guaranteeing only at least $\left(\frac{1}{2}+\frac{\mathscr{O}(1)}{d}\right)m$ clauses. In 2015 Barak *et al.* [Bar+15] subsequently gave a classical randomized approximation algorithm for Max-k-XOR with an expected performance of at least $\left(\frac{1}{2}+\frac{\mathscr{O}(1)}{d^{1/2}}\right)m$ clauses. Farhi *et al.* [FGG14a] soon thereafter provided an improved analysis showing that QAOA_1 matches this bound within a $\mathscr{O}(\ln(d))$ factor, *i.e.*, $\left(\frac{1}{2}+\frac{\mathscr{O}(1)}{d^{1/2}\ln(d)}\right)m$. Substantial improvements in these results are unlikely since Trevisan showed in 2001 that there is a constant $c>0$ such that a $\left(\frac{1}{2}+\frac{c}{d^{1/2}}\right)$-approximation is NP-hard [Tre01].

For Max-Cut in 3-regular graphs, QAOA_1 is a 0.6924-approximation [FGG14b]; however, Halperin, Livnat, and Zwick [HLZ02] give a classical 0.9326-approximation based upon an improved SDP relaxation. Yet it is still possible that QAOA_1 may outperform classical algorithms on specific instances of 3-regular Max-Cut.

2.4 A Comment on Sampling versus Optimization

Comparing the performance of QAOA against classical algorithms requires a bit of care. Farhi and Harrow showed that QAOA_1, for C corresponding to a classical 2-local CSP, can produce quantum states that are hard to sample from classically [FH16]. However, QAOA is foremost an optimization algorithm, and as such the most relevant performance measure is $\langle\gamma,\beta|C|\gamma,\beta\rangle$, as we seek a state that maximizes C. Thus, although a classical algorithm may not be able to sample from $|\gamma,\beta\rangle$, it can still outperform QAOA as an optimization algorithm if its performance exceeds $\langle\gamma,\beta|C|\gamma,\beta\rangle$.

To further illustrate the distinction between sampling and optimization in the context of QAOA, observe that for C defined above corresponding to Max-Cut, we may multiply any term $\frac{1}{2}(I-Z_iZ_j)$ by $(1+2\pi l/\gamma)$ for an integer l without altering the resulting state $|\gamma,\beta\rangle$. This observation can be used to show that the states used by Farhi and Harrow to establish the classical hardness of sampling from QAOA correspond to a multitude of weighted optimization problems, including trivial ones.

2.5 Results

We now summarize the results of our analysis found later in Section 2.6. Note, in our analysis we consider QAOA's simplest form (QAOA_1):

$$|\gamma,\beta\rangle = e^{-i\beta B}e^{-i\gamma C}|s\rangle. \tag{2.33}$$

Theorem 2.1 (Shown formally by Theorem 2.4.)**.** *For any 3-regular graph, there is a deterministic linear-time classical algorithm that delivers at least as large a cut as $QAOA_1$ is expected to.*

On the other hand, we show:

Theorem 2.2 (Shown formally by Theorem 2.3.)**.** *The expected number of edges cut by $QAOA_1$ on a k-regular triangle-free graph with m edges is*

$$\langle C\rangle = \left(\frac{1}{2}+\frac{1}{2\sqrt{k}}\left(1-\frac{1}{k}\right)^{\frac{k-1}{2}}\right)m \geq \left(\frac{1}{2}+\frac{0.3032}{\sqrt{k}}\right)m. \tag{2.34}$$

This improves upon classical randomized algorithms developed by Hirvonen, Rybicki, Schmid, and Suomela [Hir+14] in 2014, with performance $(\frac{1}{2}+\frac{0.2812}{\sqrt{k}})m$, and by Shearer [She92] in 1992, with performance $(\frac{1}{2}+\frac{0.177}{\sqrt{k}})m$.

Our results here are enabled by a new exact closed-form expression for the expectation of QAOA_1 on any Max-Cut instance, $G=(V,E)$. The expectation for an edge $ij\in E$ is

$$\begin{aligned}\langle C_{ij}\rangle = \frac{1}{2} &- \frac{1}{4}\sin^2(2\beta)\cos(\gamma)^{\sigma_i+\sigma_j-2(n_{ij}+1)}(1-\cos(2\gamma)^{n_{ij}}) \\ &+ \frac{1}{4}\sin(4\beta)\sin(\gamma)\left[\cos(\gamma)^{\sigma_i-1}+\cos(\gamma)^{\sigma_j-1}\right]\end{aligned} \tag{2.35}$$

(the proof of which is given in Lemma 2.2), where σ_u is the degree of a vertex $u\in V$, and n_{ij} are the numbers of common neighbors of i and j, which is also the number of triangles in G containing the edge ij. The parameters β and γ are specified as input to QAOA_1, and one typically chooses them to maximize the expectation. We give precise values for β and γ that maximize the expectation of QAOA_1 on k-regular triangle-free instances of Max-Cut. Our formula helps shed light on the features of a graph that influence the performance of QAOA_1. We note that the above result was obtained simultaneously and independently by Wang et al. [Wan+17], who applied it in a different context.

2.6 Analysis of QAOA for Max-Cut

For Max-Cut on general graphs we obtain an exact closed-form expression for the expectation of QAOA_1. Since the objective function of a classical CSP can be written as a sum of tensor products of Pauli Zs, we start our analysis by considering the following lemma.

Lemma 2.1. *For $|\gamma,\beta\rangle$ as defined as in Eq. 2.33 with C as defined in Eq. 2.10 and $B = \sum_{j\in[n]} X_j$ as defined in Eq. 2.14,*

$$\langle\gamma,\beta|Z^K|\gamma,\beta\rangle = \langle s|Z^K\left[\sum_{L\subseteq K}(i)^{|L|}\cos(2\beta)^{|K|-|L|}\sin(2\beta)^{|L|}X^L\prod_{\substack{M\subseteq[n]\\ |M\cap L| \text{ is odd}}}\exp\left(-2i\gamma W_M Z^M\right)\right]|s\rangle \tag{2.36}$$

Proof. First note

$$\begin{aligned}
\langle\gamma,\beta|Z^K|\gamma,\beta\rangle &= \langle s|e^{i\gamma C}e^{i\beta B}Z^K e^{-i\beta B}e^{-i\gamma C}|s\rangle \\
&= \langle s|e^{i\gamma C}e^{i\beta B}\left[\bigotimes_{u\in K}Z_u\right]e^{-i\beta B}e^{-i\gamma C}|s\rangle \\
&= \langle s|e^{i\gamma C}\left[\bigotimes_{u\in K}e^{i\beta B}Z_u e^{-i\beta B}\right]e^{-i\gamma C}|s\rangle \\
&= \langle s|e^{i\gamma C}\left[\bigotimes_{u\in K}Z_u e^{-2i\beta X_u}\right]e^{-i\gamma C}|s\rangle \\
&= \langle s|e^{i\gamma C}\left[\bigotimes_{u\in K}Z_u\right]\left[\bigotimes_{u\in K}e^{-2i\beta X_u}\right]e^{-i\gamma C}|s\rangle \\
&= \langle s|Z^K e^{i\gamma C}\left[\bigotimes_{u\in K}e^{-2i\beta X_u}\right]e^{-i\gamma C}|s\rangle\,,
\end{aligned} \tag{2.37}$$

since $e^{-i\beta X_u}$ and Z_i commute if $u \neq i$ and anticommute otherwise. Also, both Z^K and $e^{-i\gamma C}$ commute since both are diagonal matrices.

Continuing, we have

$$\langle s|Z^K e^{i\gamma C}\left[\bigotimes_{u\in K} e^{-2i\beta X_u}\right]e^{-i\gamma C}|s\rangle =$$

$$\langle s|Z^K e^{i\gamma C}\left[\bigotimes_{u\in K}(\cos(2\beta)-iX_u\sin(2\beta))\right]e^{-i\gamma C}|s\rangle =$$

$$\langle s|Z^K e^{i\gamma C}\left[\sum_{L\subseteq K}\left((i)^{|L|}\cos(2\beta)^{|K|-|L|}\sin(2\beta)^{|L|}X^L\right)\right]e^{-i\gamma C}|s\rangle =$$

$$\langle s|Z^K\left[\sum_{L\subseteq K}(i)^{|L|}\cos(2\beta)^{|K|-|L|}\sin(2\beta)^{|L|}e^{i\gamma C}X^L e^{-i\gamma C}\right]|s\rangle =$$

$$\langle s|Z^K\left[\sum_{L\subseteq K}(i)^{|L|}\cos(2\beta)^{|K|-|L|}\sin(2\beta)^{|L|}\right.$$

$$\left.\left(\prod_{M\subseteq[n]}e^{i\gamma W_M Z^M}\right)X^L\left(\prod_{M\subseteq[n]}e^{-i\gamma W_M Z^M}\right)\right]|s\rangle =$$

$$\langle s|Z^K\left[\sum_{L\subseteq K}(i)^{|L|}\cos(2\beta)^{|K|-|L|}\sin(2\beta)^{|L|}X^L\prod_{\substack{M\subseteq[n]\\ |M\cap L| \text{ is odd}}}\exp\left(-2i\gamma W_M Z^M\right)\right]|s\rangle\,, \tag{2.38}$$

where the identity $e^{i\theta A} = I\cos(\theta) + iA\sin(\theta)$ (assuming $A^2 = I$ for matrix A) was used. Note that if $M = \varnothing$, then $|M\cap L| = 0$; thus, $M = \varnothing$ is not evaluated in the product. □

Restricting the claim of Lemma 2.1 to the case of the Max-Cut problem results in the following lemma.

Lemma 2.2. *Let σ_i be the degree of a vertex $i \in V$, and let n_{ij} be the number of common neighbors of i and j. For $|\gamma,\beta\rangle$ as defined as in Eq. 2.33 with $C = \sum_{ij\in E}\frac{1}{2}(I - Z_iZ_j)$ and $B = \sum_{i\in V}X_i$,*

$$\begin{aligned}\langle\gamma,\beta|Z_iZ_j|\gamma,\beta\rangle = &\frac{1}{2}\sin^2(2\beta)\cos(\gamma)^{\sigma_i+\sigma_j-2(n_{ij}+1)}(1-cos(2\gamma)^{n_{ij}})\\ &-\frac{1}{2}\sin(4\beta)\sin(\gamma)\left(\cos(\gamma)^{\sigma_i-1}+\cos(\gamma)^{\sigma_j-1}\right),\end{aligned} \tag{2.39}$$

when $ij \in E$.

Proof. Note that Eq. 2.17 can be written as

$$C = \frac{m}{2}I - \sum_{ij\in E}\frac{1}{2}Z_iZ_j. \tag{2.40}$$

We then first consider evaluating $\langle\gamma,\beta|Z_iZ_j|\gamma,\beta\rangle$. Applying Eq. 2.36 of Lemma 2.1 to $\langle\gamma,\beta|Z_iZ_j|\gamma,\beta\rangle$ for Max-Cut, we get

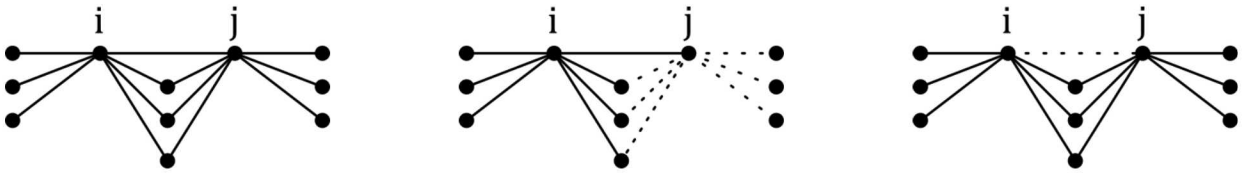
$$\begin{aligned}
\langle\gamma,\beta|Z_iZ_j|\gamma,\beta\rangle = \langle s|\cos^2(2\beta)Z_iZ_j \prod_{\substack{uv\in E\\ |uv\cap\varnothing| \text{ is odd}}} \exp(i\gamma Z_uZ_v) && (2.41)\\
-\sin(2\beta)\cos(2\beta)Z_iY_j \prod_{\substack{uv\in E\\ |uv\cap\{j\}| \text{ is odd}}} \exp(i\gamma Z_uZ_v) && (2.42)\\
-\sin(2\beta)\cos(2\beta)Y_iZ_j \prod_{\substack{uv\in E\\ |uv\cap\{i\}| \text{ is odd}}} \exp(i\gamma Z_uZ_v) && (2.43)\\
+\sin^2(2\beta)Y_iY_j \prod_{\substack{uv\in E\\ |uv\cap\{i,j\}| \text{ is odd}}} \exp(i\gamma Z_uZ_v)|s\rangle, && (2.44)
\end{aligned}$$

where we have used the identity $Y = iXZ$ to simplify the expression.

To simplify further derivations, we defining the edge set

$$\Delta_S \equiv \{uv\in E|\ |\{u,v\}\cap S| \text{ is odd }\} \text{ where } S\subseteq V. \tag{2.45}$$

For clarity, Fig. 2.1 depicts examples of edges sets $\Delta_{\{i\}}$ and $\Delta_{\{i,j\}}$.



(a) Example Graph (b) Edges in $\Delta_{\{i\}}$ (c) Edges in $\Delta_{\{i,j\}}$

Figure 2.1: Given the graph (a), the solid edges in (b) are the set of edges in $\Delta_{\{i\}}$ and the solid edges in (c) are the set of edges in $\Delta_{\{i,j\}}$. That is, (a) and (b) are graphical depictions of the set of edges (Δ_S) that are incident on the corresponding set of vertices (S) an odd number of times.

We then obtain

$$\begin{aligned}\langle \gamma,\beta|Z_iZ_j|\gamma,\beta\rangle = \langle s|\cos^2(2\beta)Z_iZ_j \\ -\sin(2\beta)\cos(2\beta)Z_iY_j\prod_{uv\in\Delta_{\{j\}}}\exp(i\gamma Z_uZ_v) \\ -\sin(2\beta)\cos(2\beta)Y_iZ_j\prod_{uv\in\Delta_{\{i\}}}\exp(i\gamma Z_uZ_v) \\ +\sin^2(2\beta)Y_iY_j\prod_{uv\in\Delta_{\{i,j\}}}\exp(i\gamma Z_uZ_v)|s\rangle. \end{aligned} \tag{2.46}$$

Note, the first term drops out since $\langle s|Z_iZ_j|s\rangle = 0$ for $i \neq j$.

We next derive closed-form expressions for $\langle s|Y_iY_j\left[\prod_{uv\in\Delta_{\{i,j\}}}e^{(i\gamma Z_uZ_v)}\right]|s\rangle$ and $\langle s|Z_iY_j\left[\prod_{uv\in\Delta_{\{j\}}}e^{(i\gamma Z_uZ_v)}\right]|s\rangle$; an expression $\langle s|Y_iZ_j\left[\prod_{uv\in\Delta_{\{i\}}}e^{(i\gamma Z_uZ_v)}\right]|s\rangle$ will follow from the analysis of the latter.

We consider $\langle s|Z_iY_j\left[\prod_{uv\in\Delta_{\{j\}}}e^{(i\gamma Z_uZ_v)}\right]|s\rangle$ first. Letting

$$\alpha_F \equiv (i\sin(\gamma))^{|F|}\cos(\gamma)^{|\Delta_{\{j\}}|-|F|} \tag{2.47}$$

for $F \subseteq \Delta_{\{j\}}$, we have

$$\begin{aligned}\langle s|Z_iY_j\left[\prod_{uv\in\Delta_{\{j\}}}e^{i\gamma(Z_uZ_v)}\right]|s\rangle &= \langle s|Z_iY_j\left[\prod_{uj\in\Delta_{\{j\}}}e^{i\gamma(Z_jZ_u)}\right]|s\rangle \\ &= \langle s|Z_iY_j\prod_{uj\in\Delta_{\{j\}}}(\cos(\gamma)I + i\sin(\gamma)Z_jZ_u)|s\rangle \\ &= \langle s|Z_iY_j\sum_{F\subseteq\Delta_{\{j\}}}\alpha_F\prod_{uj\in F}Z_jZ_u|s\rangle \\ &= \sum_{F\subseteq\Delta_{\{j\}}}\alpha_F\,\langle s|Z_iY_j\prod_{uj\in F}Z_jZ_u|s\rangle \\ &= \sum_{F\subseteq\Delta_{\{j\}}}\alpha_F\,\mathrm{Tr}\left(\left(\otimes_{i\in V}\frac{1}{2}(I+X_i)\right)Z_iY_j\prod_{uj\in F}Z_jZ_u\right). \end{aligned} \tag{2.48}$$

Note, here we have used the fact that $|+\rangle\langle+| = (I+X)/2$ to replace $|s\rangle\langle s|$ with $\otimes_{i\in V}\frac{1}{2}(I+X_i)$.

The operator within the trace is a sum of tensor product of Pauli operators on each qubit, hence the trace is proportional to the coefficient of the I term. The only way to obtain such a term is when F is chosen so that it contains an odd number of edges incident upon each of i and j, and an even number of edges incident upon all other vertices. The only such $F \subseteq \Delta_{\{j\}}$ is $F = \{\{i,j\}\}$.

In other words, it is a necessary condition for non-vanishing terms that the product $\prod_{uj\in F} Z_j Z_u$ yields the value $Z_i Z_j$ to cancel the $Z_i Z_j$ introduced by $Z_i Y_j$ in the expression in the last line of Eq. 2.48. Given that sum on F is restricted to $F \subseteq \Delta_{\{j\}}$ (*e.g.*, see Fig. 2.1b), the sufficient condition that $\prod_{uj\in F} Z_j Z_u$ yields $Z_i Z_j$ is only met for $F = \{\{i, j\}\}$. Terms where $F \neq \{\{i, j\}\}$ necessarily vanish.

Restricting the sum to $F = \{\{i, j\}\}$ leads to

$$\begin{aligned}
&\sum_{F\subseteq\Delta_{\{j\}}} \alpha_F \operatorname{Tr}\left(\left(\otimes_{i\in V}\frac{1}{2}(I+X_i)\right) Z_i Y_j \prod_{uj\in F} Z_j Z_u\right) \\
&= \alpha_{\{ij\}} \operatorname{Tr}\left(\left(\otimes_{i\in V}\frac{1}{2}(I+X_i)\right) Z_i Y_j Z_j Z_i\right) \\
&= \alpha_{\{ij\}} \operatorname{Tr}\left(\left(\otimes_{i\in V}\frac{1}{2}(I+X_i)\right) iX_j\right) \\
&= i\alpha_{\{ij\}} \\
&= -\sin(\gamma)\cos(\gamma)^{|\Delta_{\{j\}}|-1}.
\end{aligned} \tag{2.49}$$

The above, in conjunction with Eq. 2.48, yields

$$\langle s| Z_i Y_j \left[\prod_{uv\in\Delta_{\{j\}}} e^{i\gamma(Z_u Z_v)}\right] |s\rangle = -\sin(\gamma)\cos(\gamma)^{|\Delta_{\{j\}}|-1}. \tag{2.50}$$

The following may be derived analogously.

$$\langle s| Y_i Z_j \left[\prod_{uv\in\Delta_{\{i\}}} e^{i\gamma(Z_u Z_v)}\right] |s\rangle = -\sin(\gamma)\cos(\gamma)^{|\Delta_{\{i\}}|-1}. \tag{2.51}$$

It remains to analyze $\langle s|Y_iY_j\left[\prod_{uv\in\Delta_{\{i,j\}}} e^{i\gamma(Z_uZ_v)}\right]|s\rangle$. Following the derivation of Eq. 2.48:

$$\begin{aligned}
&\langle s|Y_iY_j\left[\prod_{uv\in\Delta_{\{i,j\}}} e^{i\gamma(Z_uZ_v)}\right]|s\rangle \\
&= \langle s|Y_iY_j \prod_{uv\in\Delta_{\{i,j\}}} (\cos(\gamma)I + i\sin(\gamma)Z_uZ_v)|s\rangle \\
&= \sum_{F\subseteq\Delta_{\{i,j\}}} \alpha_F \langle s|Y_iY_j \prod_{uv\in F} Z_uZ_v|s\rangle \\
&= \sum_{F\subseteq\Delta_{\{i,j\}}} \alpha_F \operatorname{Tr}\left((\otimes_{i\in V}\frac{1}{2}(I+X_i))Y_iY_j \prod_{uv\in F} Z_uZ_v\right).
\end{aligned} \tag{2.52}$$

As with our previous derivation, the only terms of the sum that do not vanish are those for which $F \subseteq \Delta_{\{ij\}}$ (for reference see, *e.g.*, Fig. 2.1c) contains an odd number of edges incident upon each of i and j, and an even number of edges incident upon all other vertices. Or, in other words, non-vanishing terms only arise when the product $\prod_{uv\in F} Z_uZ_v$ yields the value Z_iZ_j to cancel the Z_iZ_j introduced by Y_iY_j. Note that $ij \notin \Delta_{\{i,j\}}$; however, any path of length two from i to j results in a non-vanishing term in the sum. So, for example, the product Z_iZ_a times Z_aZ_j ((i.e.,) $F = \{\{i,a\},\{a,j\}\}$) would survive as it would yield Z_iZ_j. Moreover, a union of an odd number of such paths also yields a non-vanishing term. Let $W_{ij} \subseteq V$ be the set of common neighbors of both i and j (*e.g.*, see Fig. 2.2). Then each path of length two between i and j, $(ik;kj)$, is in one-to-one correspondence with a $k \in W_{ij}$. Finally, observe that α_F only depends on $|F|$; we will use α_l as shorthand for an α_F with $|F| = l$.

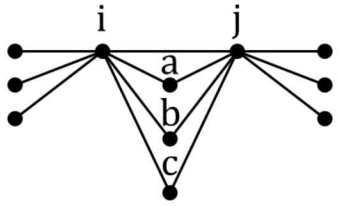


Figure 2.2: An example of W_{ij}. Here $W_{ij} = \{a, b, c\}$. That is, W_{ij} is the set of vertices that are common neighbors to i and j.

We consequently have

$$\begin{aligned}
&\sum_{F \subseteq \Delta_{\{i,j\}}} \alpha_F \operatorname{Tr}\left(\left(\otimes_{i \in V} \frac{1}{2}(I + X_i)\right) Y_i Y_j \prod_{uv \in F} Z_u Z_v\right) \\
&= \sum_{\substack{U \subseteq W_{ij}: \\ |U| \text{ odd}}} \alpha_{2|U|} \operatorname{Tr}\left(\left(\otimes_{i \in V} \frac{1}{2}(I + X_i)\right) i X_i X_j\right) \\
&= -\sum_{\substack{U \subseteq W_{ij}: \\ |U| \text{ odd}}} \alpha_{2|U|} = -\sum_{\substack{1 \le l \le |W_{ij}|: \\ l \text{ odd}}} \binom{|W_{ij}|}{l} \alpha_{2l} \\
&= -\sum_{\substack{1 \le l \le |W_{ij}|: \\ l \text{ odd}}} \binom{|W_{ij}|}{l} (i \sin(\gamma))^{2l} \cos(\gamma)^{|\Delta_{i,j}| - 2l} \\
&= \cos(\gamma)^{|\Delta_{\{i,j\}}| - 2|W_{ij}|} \sum_{\substack{1 \le l \le |W_{ij}|: \\ l \text{ odd}}} \binom{|W_{ij}|}{l} \left(\sin^2(\gamma)\right)^l \left(\cos^2(\gamma)\right)^{|W_{ij}| - l} \\
&= \cos(\gamma)^{|\Delta_{\{i,j\}}| - 2|W_{ij}|} \frac{1 - \left(\cos^2(\gamma) - \sin^2(\gamma)\right)^{|W_{ij}|}}{2} \\
&= \frac{1}{2} \cos(\gamma)^{|\Delta_{\{i,j\}}| - 2|W_{ij}|} \left(1 - \cos(2\gamma)^{|W_{ij}|}\right),
\end{aligned} \tag{2.53}$$

where, as mentioned, we use α_l as shorthand for an α_F with $|F| = l$ and the penultimate equality follows from the identity

$$\sum_{l \text{ odd}} \binom{n}{l} q^l p^{n-l} = ((p+q)^n - (p-q)^n)/2. \tag{2.54}$$

By Eq. 2.52, we have

$$\langle s | e^{-i\gamma C} Y_i Y_j e^{i\gamma C} | s \rangle = \frac{1}{2} \cos(\gamma)^{|\Delta_{\{i,j\}}| - 2|W_{ij}|} \left(1 - \cos(2\gamma)^{|W_{ij}|}\right). \tag{2.55}$$

Combining Eq. 2.46 with Equations 2.50, 2.51, and 2.55 yields the claim of the lemma, since $ij \in E$ and $|\Delta_{\{i,j\}}| = \delta_i + \delta_j - 2$ (the number of edges shared between i and j not including the edge ij). Recall that σ_u is the degree of vertex u and $|W_{ij}| = n_{ij}$ is the number of common neighbors between i and j. □

Corollary 2.2.1. *The expected number of edges cut by QAOA$_1$ on a k-regular graph with m edges is,*

$$\langle C \rangle = \left(\frac{1}{2} - \frac{1}{4} \sin^2(2\beta) \cos(\gamma)^{2k-2(n_{ij}+1)} \left(1 - \cos(2\gamma)^{n_{ij}}\right) + \frac{1}{2} \sin(4\beta) \sin(\gamma) \cos(\gamma)^{k-1} \right) m. \tag{2.56}$$

Proof. This follows from Lemma 2.2 since

$$\langle \gamma, \beta | C | \gamma, \beta \rangle = \frac{m}{2} - \frac{1}{2} \sum_{ij \in E} \langle \gamma, \beta | Z_i Z_j | \gamma, \beta \rangle . \tag{2.57}$$

□

Theorem 2.3. *We may set β and γ so that the expected number of edges cut by QAOA$_1$ on a k-regular triangle-free graph with m edges is*

$$\langle C \rangle = \left(\frac{1}{2} + \frac{1}{2\sqrt{k}} \left(1 - \frac{1}{k}\right)^{\frac{k-1}{2}} \right) m \geq \left(\frac{1}{2} + \frac{0.3032}{\sqrt{k}} \right) m. \tag{2.58}$$

Proof. Since $n_{ij} = 0$ for a triangle-free graph, Corollary 2.2.1 yields

$$\langle \gamma, \beta | C | \gamma, \beta \rangle = \left(\frac{1}{2} + \frac{1}{2} \sin(4\beta) \sin(\gamma) \cos(\gamma)^{k-1} \right) m. \tag{2.59}$$

We select $\beta = \pi/8$ to maximize the above. The quantity $\sin(\gamma)\cos(\gamma)^{k-1}$ is maximized when $\sin(\gamma) = \frac{1}{\sqrt{k}}$ and $\cos(\gamma) = \sqrt{\frac{k-1}{k}}$ so that,

$$\begin{aligned} \langle \gamma, \beta | C | \gamma, \beta \rangle &= \left(\frac{1}{2} + \frac{1}{2\sqrt{k}} \left(1 - \frac{1}{k}\right)^{\frac{k-1}{2}} \right) m \\ &\geq \left(\frac{1}{2} + \frac{1}{2\sqrt{e}} \frac{1}{\sqrt{k}} \right) m \geq \left(\frac{1}{2} + \frac{0.3032}{\sqrt{k}} \right) m. \end{aligned} \tag{2.60}$$

where the first inequality follows since for $k \geq 1$, $(1-1/k)^{k-1} \geq 1/e$ (see, *e.g.*, page 435 of Ref. [MR95]). □

Lemma 2.3. *For a 3-regular graph with m edges, the expectation of QAOA$_1$ is at most,*

$$\left(\frac{1}{2}+\frac{1}{3\sqrt{3}}\right)m \leq 0.6925m. \tag{2.61}$$

Proof. For a k-regular graph, we observe that the $\frac{1}{4}\sin^2(2\beta)\cos(\gamma)^{2k-2(n_{ij}+1)}(1-\cos(2\gamma)^{n_{ij}})$ term from Corollary 2.2.1 is non-negative, hence

$$\langle\gamma,\beta|C|\gamma,\beta\rangle \leq \left(\frac{1}{2}+\frac{1}{2}\sin(4\beta)\sin(\gamma)\cos(\gamma)^{k-1}\right)m. \tag{2.62}$$

We obtain the result by using the values of β and γ from the proof of Theorem 2.3 for $k=3$. □

Theorem 2.4. *For any 3-regular graph, there is a linear-time deterministic classical algorithm that delivers at least as large a cut as QAOA$_1$ is expected to.*

Proof. Locke [Loc82] gives a deterministic algorithm that yields a cut with at least $\frac{7}{9}m$ edges in any 3-regular graph except K_4. By Lemma 2.3, Locke's algorithm outperforms QAOA$_1$ on 3-regular graphs, except possibly K_4. The most time-consuming step of Locke's algorithm for 3-regular graphs is finding a Brooks' coloring, which can be done in linear time [Sku02]. □

2.7 Conclusion

In conclusion, we analytically evaluated the performance of QAOA$_1$ on the Max-Cut problem. In so doing, we developed a closed-form expression for the expectation of QAOA$_1$ for Max-Cut on any simple graph (see Lemma 2.2). Restricting QAOA$_1$ on Max-Cut to k-regular triangle-free graphs with m edges, we found the optimal values for the angles β and γ. These angles allowed us to determine that for k-regular triangle-free graphs $\langle C\rangle = \left(\frac{1}{2}+\frac{1}{2\sqrt{k}}\left(1-\frac{1}{k}\right)^{\frac{k-1}{2}}\right)m \geq \left(\frac{1}{2}+\frac{0.3032}{\sqrt{k}}\right)m$. These results were obtained simultaneously and independently by Wang *et al.* [Wan+17]; however, our derivation differs from theirs and may provide further insight into future analyses of QAOA. We observe that our analysis of performance of QAOA$_1$ on Max-Cut for k-regular triangle-free graphs improves upon the currently known best classical approximation algorithm for these graphs for $k > 3$, which was developed by Hirvonen *et al.* [Hir+14] in 2014. This classical randomized algorithm has a performance of $\overline{C(z)} = (\frac{1}{2}+\frac{0.2812}{\sqrt{k}})m$.

This page intentionally left blank.

Chapter 3

Approximation Algorithms for Ground States of Physically Motivated Hamiltonians

This chapter represents joint work of Sevag Gharibian and Ojas Parekh.

3.1 Introduction

The study of approximation algorithms for NP-complete problems is a central area of research in theoretical computer science (see, e.g., [Hoc97; Vaz01]). Indeed, the field has seen breakthroughs such as the celebrated Goemans-Williamson [GW95] 0.878-approximation algorithm for Max-Cut, and the PCP theorem [AS98; Aro+98], which yielded a general framework for showing hardness of approximation results. Here, an approximation algorithm A with ratio $0 < r < 1$ is defined as follows: Given an instance Π of a maximization problem with optimal value OPT, A runs in polynomial time and outputs a value $\widetilde{\text{OPT}}$ satisfying $r\text{OPT} \leq \widetilde{\text{OPT}} \leq \text{OPT}$. Focal points of study in approximation algorithms are Boolean constraint satisfaction problems (CSPs) such as Max-SAT and Max-Cut, in which one is roughly given a set of local constraints acting on $k \in O(1)$ bits each (out of a total of n bits), and asked to compute the largest number of constraints which are simultaneously satisfiable.

In the quantum setting, CSPs are naturally generalized by the *k-local Hamiltonian problem (k-LH)* [KSV02]. In the latter, one is given as input a local Hamiltonian $H = \sum_{S \subseteq [n]} H_S$ acting on n qubits, where each local "clause" H_S acts on a constant number k of qubits denoted by subset $S \subseteq [n]$ with $|S| = k$. (Formally, H_S implicitly denotes operator $I_{[n]\setminus S} \otimes H_S$.) The goal (roughly) is to estimate the smallest eigenvalue of H, $\lambda_{\min}(H)$, i.e. its *ground state energy*. Due to its physical significance and importance for quantum complexity theory (5-LH was the first known QMA-complete problem [KSV02], where QMA is Quantum Merlin-Arthur), k-LH has been a central problem of study in the field of Quantum Hamiltonian Complexity (see, e.g. [Osb12; Gha+14] for surveys).

Despite this, relatively little is known about the *approximability* of k-LH. What is particularly interesting here is that there are now two notions of approximation one may consider — *classical* approximation algorithms for QMA, or *quantum* approximation algorithms for either NP or QMA. In this work, we consider both notions.

3.1.1 Previous Work

Classical approximation algorithms for k-LH. A challenge in approximating k-LH is that the ground state of H (the eigenvector corresponding to $\lambda_{\min}(H)$) may require exponentially many bits to represent. Nevertheless, in 2009, Bansal, Bravyi, and Terhal [BBT09] gave a classical polynomial-time approximation scheme (PTAS) for k-LH on bounded degree planar graphs. Gharibian and Kempe [GK12] gave a PTAS for computing product-state solutions to dense CSPs, and showed their algorithm yielded a d^{1-k} approximation for dense k-LH on local d-dimensional systems. Brandão and Harrow [BH13] then gave PTAS-es for k-LH in three settings: Planar, dense, and low threshold rank. (The planar and dense settings are QMA-hard [OT08; GK12].) Each of these results optimizes over a set of succinctly representable states, or an *ansatz*, specifically the mean-field or tensor product ansatz $\rho = \bigotimes_{i=1}^{n} \rho_i$. Note that while mean-field states cannot capture entanglement, mean-field theory has proven remarkably useful in condensed matter theory [Gha+14].

In addition to the above, very recently, Bravyi, Gosset, König, and Temme used [Bra+18] techniques similar to ours to give a $O(\log n)$-approximation algorithm for traceless 2-local Hamiltonians (again, in the maximization setting). This complements our results nicely: We provide *constant*-factor (i.e. stronger) approximations for physical Hamiltonians such as the antiferromagnet (as well as a generalization to non-traceless symmetric Hamiltonians in Section 2.3 of our technical version), whereas [Bra+18] studies the class of all traceless Hamiltonians, at the expense of a weaker (logarithmic) approximation ratio. Both papers employ the same SDP relaxation, but use different rounding schemes generalizing that of [GW94]. Together, the two works hence provide a foundation for employing SDP relaxations for approximating 2-LH.

Next, we discuss Hamiltonians of *intermediate* complexity. Here, Bravyi [Bra15] and Bravyi and Gosset [BG16] showed fully polynomial randomized approximation schemes (FPRAS) for approximating the partition function[1] of certain ferromagnetic models, such as the ferromagnetic transverse field Ising model (ferromagnetic TIM). Bravyi and Hastings showed TIM is StoqMA-complete [BH14], for MA $\subseteq$ StoqMA $\subseteq$ QMA.

Finally, an important related open question is: Does a quantum PCP theorem hold [Aha+09; AAV13]? Recently, the "entangled non-local games" version of the PCP theorem has been established under randomized reductions [NV18], but the "hardness of approximation" version involving approximating ground state energies of local Hamiltonians, which is relevant to this work, remains open.

3.1.2 Our Results

In this work, we give two main results, which we now discuss. The first involves classical algorithms for QMA, and the second quantum algorithms for NP. For clarity, the first result studies the natural *maximization* variant of k-LH, in which one is given H and asked to estimate its *largest*

[1] The ability to compute the partition function allows one in turn to solve k-LH.

eigenvalue $\lambda_{\max}(H)$. We study this variant for two reasons (see also [GK11]): First, in the minimization setting, if $\lambda_{\min}(H) = 0$, the notion of an approximation ratio is not well-defined, and second, the maximization setting allows us to naturally align with classical approximation algorithms for CSPs such as Max-Cut. We remark that in the exact setting, computing $\lambda_{\min}(H)$ is equivalent in complexity to computing $\lambda_{\max}(H)$ since $\lambda_{\min}(H) = \lambda_{\max}(-H)$ — thus, both maximization and minimization variants of k-LH are QMA-complete. In terms of approximability, however, the complexity of both models need not coincide in general.

Classical approximation algorithms for the quantum Heisenberg model. An important task in both classical constraint satisfaction and condensed matter physics is to solve constraint satisfaction problems which arise from *well-motivated* constraint classes. For example, classically, such an important class is that of Max-Cut. It turns out that in the quantum setting, Max-Cut has a natural quantum analogue (in the context of k-LH), known as the *quantum Heisenberg model*. The latter is a fundamental model in the study of magnetism, and has received attention for at least almost a century now (e.g. the well-known Bethe ansatz of 1931 [Bet31]). Generally speaking, the model is given by a 2-local Hamiltonian, defined in this paper as having constraints H_{ij} acting on qubits i and j of the form (see Section 3.2 for formal definitions):

$$H_{ij} = I - \alpha X_i \otimes X_j - \beta Y_i \otimes Y_j - \gamma Z_i \otimes Z_j.$$

(Recall we study *maximization*, i.e. estimating $\lambda_{\max}(H)$. Also, here X, Y, Z denote the Pauli matrices.) Two important well-known special cases are the Heisenberg anti-ferromagnet ($\alpha = \beta = \gamma = 1$) and XY models ($\alpha = \beta = 1$, $\gamma = 0$). The anti-ferromagnet, for example, is a notoriously difficult model to solve, with solutions only known in the 1D [Bet31] and complete graph (see, e.g., [CM16]) settings. Indeed, when non-negative polynomial-size weights are allowed on each constraint, both models were recently shown to be QMA-hard [CM16; PM17].

Our first result is informally stated as follows (see Theorem 3.2 for a formal statement).

Theorem 3.1. *Let $\alpha, \beta, \gamma \in \{0,1\}$. Then, there exists a randomized, polynomial time classical algorithm which outputs a product state solution with ratio at least:*

- 0.878 *if* $\alpha + \beta + \gamma = 1$ *(equivalent to Max-Cut),*
- 0.649 *if* $\alpha + \beta + \gamma = 2$ *(includes the XY model),*
- 0.498 *if* $\alpha + \beta + \gamma = 3$ *(anti-ferromagnet).*

In fact, in Section 3.2.3 we generalize this algorithm to work for *any* 2-local Hamiltonian of the form $I - P$, where P is traceless and contains no linear terms, and the representation of P in the Pauli basis is an orthogonal projector. Formally, these are Hamiltonians of form:

$$H_{ij} := I - \sum_{t \leq 3} \alpha_t (q_{t,1} X_i + q_{t,2} Y_i + q_{t,3} Z_i)(q_{t,1} X_j + q_{t,2} Y_j + q_{t,3} Z_j),$$

with $\alpha_t \in \{0,1\}$, $\sum_{k \leq 3} q_{t,k}^2 = 1$ for all t, and $\sum_{k \leq 3} q_{s,k} q_{t,k} = 0$ for distinct s, t. Here, the variables $\alpha_1, \alpha_2, \alpha_3$ play the same role as α, β, γ, and the approximation ratio obtained matches that of

Theorem 3.1 (i.e. depending on $\sum_t \alpha_t$). Thus, in addition to the models listed in Theorem 3.1, we are able to get (e.g.) a 0.878-approximation for Hamiltonians with constraints such as $I - \frac{1}{2}(X_iX_j + X_iZ_j + Z_iX_j + Z_iZ_j)$.

The significance of Theorem 3.1 is threefold. (1) The ratios 0.649 and 0.498 are almost optimal in the following sense: We show that a product state ansatz can achieve ratios at most $2/3$ and $1/2$ for the XY model and anti-ferromagnet, respectively, in the worst case. (Our algorithm outputs a product-state solution.) In contrast, note the naive "random assignment" strategy (i.e. choose the maximally mixed state $I/2^n$) yields ratios of only $1/3$ and $1/4$ for the XY model and anti-ferromagnet, respectively. (2) As far as we are aware, our approximation algorithm is the first tailored to a physically significant QMA-complete model (i.e. the Heisenberg model). (3) Our efficiently obtainable approximation ratios for the XY model and anti-ferromagnet of $\approx 2/3$ and $\approx 1/2$, respectively, imply that a quantum PCP theorem cannot hold for these models for these ratios, unless QMA collapses to BPP. (Previously, a non-algorithmic ratio of $1/2$ was known for models including the anti-ferromagnet, via the Mixing Lemma of [GK11]. Unlike [GK11], however, we show how to *efficiently* obtain ratio 0.498.)

3.1.3 Techniques

Classical approximation algorithms for the quantum Heisenberg model. As mentioned earlier, since optimal solutions to k-LH may not have efficient descriptions, we must optimize over a restricted ansatz or class of states; similar to previous works [BBT09; GK11; BH13], we choose to optimize over *tensor product* (i.e. mean-field) states $\rho = \rho_1 \otimes \cdots \otimes \rho_n$ for single qubit states ρ_i. (Note that optimizing over the mean-field ansatz is NP-hard in general, and thus highly non-trivial.) Our approach uses the first level of a non-commutative generalization of the Lasserre SDP hierarchy (used also in [BH13]) to approximate this optimization. In contrast to the techniques of [GK11], our approach crucially relaxes not only the best *product state* objective function value, but rather yields a relaxation of $\lambda_{\max}(H)$ itself (i.e. it relaxes the true optimal value). This is why the ratios obtained in Theorem 3.2 can be close to optimal for a product state ansatz.

In addition to our approximation algorithm, we analyze the power of the product state ansatz itself for the Heisenberg model. Specifically, Corollary 3.1.1 gives general upper bounds on the best ratio achievable by a product state for *any* setting of the parameters $\alpha, \beta, \gamma \in \mathbb{R}$ for the Heisenberg model. Our approximation algorithm in Theorem 3.2 then shows these bounds are almost tight for the cases of the XY model and anti-ferromagnet.

3.1.4 Open Questions

Many questions in the study of approximability in the quantum setting remain open. For example, what are the best achievable approximation ratios classically for the Heisenberg model? Can tight ratios of $2/3$ and $1/2$ be obtained for the XY model and anti-ferromagnet, respectively? Are there non-trivial approximation algorithms for general k-LH? How well can one approximate "interme-

diate" Hamiltonian models such as the anti-ferromagnetic TIM? Can one optimize approximately over more general ansatzes than mean-field/product states, such as tensor network states? Can quantum approximation algorithms *provably* outperform the best classical approximation algorithms? Finally, does a quantum PCP theorem (in the sense of "hardness of approximation for quantum CSPs") hold? It is hoped that the current paper will act as a step towards resolutions for some of these problems.

3.1.5 Notation

Let $[n] := \{1, \ldots, n\}$. The sets $\mathscr{H}(\mathscr{X})$, $\mathscr{P}(\mathscr{X})$, and $\mathscr{D}(\mathscr{X})$ denote the sets of Hermitian, positive semidefinite, and density operators acting on complex Euclidean space $\mathscr{X}$. For $A, B \in \mathscr{H}(\mathscr{X})$, we say $A \succeq B$ if $A - B$ is positive semidefinite, i.e. $A - B \succeq 0$. The spectral/operator norm of A is denoted $\|A\|_\infty = \mathrm{tr}(\sqrt{A^\dagger A})$.

3.2 Physically Motivated 2-Local Hamiltonians

Let $G = (V, E)$ be a simple, undirected graph with $|V| = n$ and $|E| = m$. In this section, we study physically motivated 2-local Hamiltonians H based on the quantum Heisenberg model, $H = \sum_{(i,j)\in E} w_{ij} H_{ij}$ for $H_{ij} = \alpha X_i X_j + \beta Y_i Y_j + \gamma Z_i Z_j$ (more accurately, since we are in the setting of maximization, we use local terms as given in Equation (3.1)), where we consider $\alpha, \beta, \gamma \in \{0, 1\}$ and $w_{ij} \geq 0$. This includes QMA-hard special cases such as the quantum Heisenberg antiferromagnet [CM16; PM17]. Here, X, Y, Z are the Pauli matrices

$$X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \quad Y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}, \quad Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix},$$

and X_i, Y_i, Z_i refer to the Pauli matrices acting on the *ith* qubit (i.e., tensored with identity on all other qubits).

Specifically, we consider the equivalent (in the setting of exact computation) maximization variant where each local term is defined

$$H_{ij} = I - \alpha X_i X_j - \beta Y_i Y_j - \gamma Z_i Z_j, \tag{3.1}$$

and our goal is to estimate the *largest* eigenvalue of $H = \sum_{(i,j)\in E} w_{ij} H_{ij}$ with $w_{ij} \geq 0$. This variant is clearly still QMA-hard, and includes as a special case, for example, the canonical NP-complete problem Max-Cut, obtained up to scaling by a constant factor of 2) by setting $\alpha = 0, \beta = 0, \gamma = 1$.

We now set definitions for the rest of this section. Let $F_{\alpha,\beta,\gamma}$ denote the set of all H with (non-negative weighted) constraints of the form of Equation (3.1), with parameters α, β, γ and on all interaction graphs G (for all $n \geq 0$). For example, $F_{0,0,1}$ denotes the set of all possible Max-Cut instances. In this paper, we refer to the family $F = \bigcup_{\alpha,\beta,\gamma\in\{0,1\}} F_{\alpha,\beta,\gamma}$ as "the Heisenberg model". Let $\mathrm{SEP} = \mathrm{conv}(\bigotimes_{i=1}^n \rho_i \mid \rho_i \in \mathscr{D}(\mathbb{C}^2))$ for $\mathrm{conv}(S)$ the convex hull of set S, i.e. SEP is the set of fully separable quantum states on n qubits.

3.2.1 Upper Bounds on Product State Ratios

As quantum states on n qubits generally require exponential space to represent, a classical approximation algorithm for estimating ground state energies must generally optimize over a restricted class of quantum states, or an *ansatz*. Our ansatz in this section will be to optimize over SEP. To formalize this, we first define the notion of a product state ratio.

Product state ratio Let $H \in \mathscr{H}((\mathbb{C}^2)^{\otimes n})$ be a Hermitian operator with largest eigenvalue $\mathrm{OPT}(H) = \lambda_{\max}(H)$, and let

$$\mathrm{OPT}_{\mathrm{prod}}(H) := \max_{\rho \in \mathrm{SEP}} \mathrm{tr}(H\rho).$$

By convexity, the optimal ρ here is a (pure) product state. The *product state ratio* is defined as $\mathrm{OPT}_{\mathrm{prod}}(H)/\mathrm{OPT}(H)$. For the Heisenberg model in particular, for any fixed $\alpha, \beta, \gamma \in \{0,1\}$, define

$$\Gamma_{\alpha,\beta,\gamma} = \min_{H \in F_{\alpha,\beta,\gamma}} \frac{\mathrm{OPT}_{\mathrm{prod}}(H)}{\mathrm{OPT}(H)},$$

the worst-case product state ratio over all Hamiltonians in $F_{\alpha,\beta,\gamma}$.

By definition, $\Gamma_{\alpha,\beta,\gamma}$ yields an upper bound on the best approximation ratio achievable by any approximation algorithm using a product state ansatz. It is thus crucial to understand $\Gamma_{\alpha,\beta,\gamma}$, which we now do for the Heisenberg model. For this, we first give two lemmas which fully characterize the optimal product state ratio on a *single* edge. Note the characterization we give is more general than how we defined the Heisenberg model here, in that it applies for any $\alpha, \beta, \gamma \in \mathbb{R}$. The proofs of both lemmas are deferred to Appendix B.1.

Lemma 3.1. *Let $H = \alpha X \otimes X + \beta Y \otimes Y + \gamma Z \otimes Z$ for $\alpha, \beta, \gamma \in \mathbb{R}$. Then $\mathrm{OPT}_{\mathrm{prod}}(H) = \|(\alpha, \beta, \gamma)\|_\infty$.*

Lemma 3.2. *Let $H = \alpha X \otimes X + \beta Y \otimes Y + \gamma Z \otimes Z$ for $\alpha, \beta, \gamma \in \mathbb{R}$. Then*

$$\mathrm{OPT}(H) = \max(|\alpha - \beta| + \gamma, |\alpha + \beta| - \gamma).$$

The following corollary now follows essentially immediately from Lemmas 3.1 and 3.2.

Corollary 3.1.1. *For any $\alpha, \beta, \gamma \in \mathbb{R}$,*

$$\Gamma_{\alpha,\beta,\gamma} \leq \frac{1 + \max(|\alpha|, |\beta|, |\gamma|)}{1 + \max(|\alpha - \beta| - \gamma, |\alpha + \beta| + \gamma)}.$$

Proof. Combine Lemmas 3.1 and 3.2 with the following additional observation: The values α, β, γ, as defined for $F_{\alpha,\beta,\gamma}$, should be interpreted as $-\alpha, -\beta, -\gamma$ for Lemmas 3.1 and 3.2 due to how Equation 3.1 is stated. As a result, the positions of the γ and $-\gamma$ terms are swapped in the result of Lemma 3.2. □

We thus have the following for the special case of the Heisenberg model we consider here (i.e. $\alpha, \beta, \gamma \in \{0,1\}$).

Corollary 3.1.2. *For any $\alpha,\beta,\gamma \in \{0,1\}$, if:*

- $\alpha+\beta+\gamma = 1$, *then* $\Gamma_{\alpha,\beta,\gamma} = 1$.
- $\alpha+\beta+\gamma = 2$, *then* $\Gamma_{\alpha,\beta,\gamma} \leq 2/3$.
- $\alpha+\beta+\gamma = 3$, *then* $\Gamma_{\alpha,\beta,\gamma} \leq 1/2$.

Proof. When $\alpha,\beta,\gamma \in \{0,1\}$, the bound of Corollary 3.1.1 simplifies to

$$\Gamma_{\alpha,\beta,\gamma} \leq \frac{2}{1+\alpha+\beta+\gamma},$$

from which the upper bounds claimed follow. The matching lower bound for $\alpha+\beta+\gamma = 1$ is obtained since H can be mapped via local Pauli gates to $H' \in F_{0,0,1}$, i.e. H' is diagonal in the standard basis. Thus, product states are optimal in this case. For example, applying local Hadamard gates to each qubit maps any $H \in F_{1,0,0}$ to $H' \in F_{0,0,1}$. (A matching lower bound can also be obtained for $\alpha+\beta+\gamma = 3$ by observing that $H_{ij} \succeq 0$, and using the general result that any local Hamiltonian H' (not necessarily from the Heisenberg model) with positive semidefinite constraints satisfies $\mathrm{OPT}_{\mathrm{prod}}(H')/\mathrm{OPT}(H') \geq 1/2$ [GK11]. However, unlike Theorem 3.2, the lower bound of [GK11] is not known to be efficiently achievable.) □

3.2.2 Almost Optimal Mean-Field Approximation Algorithms

In Section 3.2.1, we gave upper bounds on $\Gamma_{\alpha,\beta,\gamma}$ for the Heisenberg model. In this section, we give almost matching algorithmic lower bounds on $\Gamma_{\alpha,\beta,\gamma}$ when $\alpha+\beta+\gamma \in \{2,3\}$ (recall $\alpha+\beta+\gamma = 1$ is equivalent to Max-Cut, and so $\Gamma_{\alpha,\beta,\gamma} = 1$). Specifically, we give an approximation algorithm which is almost optimal in the following sense: Given $H \in F_{\alpha,\beta,\gamma}$, it outputs a product state ρ_{prod} with approximation ratio at least 0.649 and 0.498 when $\alpha+\beta+\gamma$ equals 2 and 3, respectively, which by Corollary 3.1.2 almost matches the best possible mean-field ratios of $2/3$ and $1/2$, respectively.

Theorem 3.2. *Let $H \in F_{\alpha,\beta,\gamma}$ for $\alpha,\beta,\gamma \in \{0,1\}$. There exists a randomized, polynomial-time algorithm which obtains approximation ratios at least* 0.878, 0.649 *or* 0.498, *when $\alpha+\beta+\gamma$ equals* 1, 2 *or* 3, *respectively.*

Proof. Suppose H has interaction graph $G = (V,E)$ for $|V| = n$ and edge weights $w_{ij} \geq 0$ for $(i,j) \in E$. We first define a semidefinite programming (SDP) relaxation of $\mathrm{OPT}(H)$ via the first level of the Lasserre hierarchy (see, e.g., [BH13] for a similar exposition for the setting of low threshold rank graphs). We then show that applying a generalization of the Goemans-Williamson (GW) [GW95; BOFV14] rounding scheme yields the desired result.

The SDP Each solution of the SDP relaxation will be a "moment matrix" $M \in \mathbb{R}^{3n\times 3n}$, whose rows (resp., columns) are indexed by 2-tuples $(i,k) \in [n]\times[3]$ (resp., $(j,l) \in [n]\times[3]$) such that ideally, i,j denote qubits, and k,j a choice of Pauli matrix from sequence $(\sigma_1,\sigma_2,\sigma_3) = (X,Y,Z)$. Under this interpretation, an ideal solution M corresponds to a density matrix $\rho \in \mathscr{D}((\mathbb{C}^2)^{\otimes n})$, such that

$$M(ik,jl) = \mathrm{tr}(\rho\sigma_k^i\sigma_l^j), \tag{3.2}$$

where σ_k^i corresponds to Pauli operator σ_k applied to qubit i, i.e. implicitly we have $\sigma_k^i \otimes I_{[n]\setminus\{i\}}$.

Let us remark about the assumption that M is real. Note that for an ideal solution (i.e. as in Equation (3.2)), M is Hermitian. Indeed, for $i \neq j$, $M(ik,jl) = M(jl,ik) \in \mathbb{R}$, since the Pauli terms act on different qubits and hence commute. (A similar argument holds for $i = j$ and $k = l$.) If, however, $i = j$ and $k \neq l$, then since the Pauli matrices anti-commute, we have $M(ik,jl) = -M(jl,ik)$, and indeed $M(ik,jl), M(jl,ik) \in \mathbb{C}\setminus\mathbb{R}$ (since, e.g., $XY = iZ$), implying $M(ik,jl)^* = M(jl,ik)$ (thus M is Hermitian; here, * denotes complex conjugate). Note, however, that the case of $i = j$ and $k \neq l$ corresponds to *linear* local terms, i.e. those of the form σ_k^i, and these are the only non-real entries of M. Since our objective function involves only quadratic local terms (i.e. $\sigma_k^i\sigma_l^j$ for $i \neq j$), we can hence eliminate entries of M with $i = j$ and $k \neq l$ by replacing M with moment matrix $M' = (M + M^*)/2$, which is real and matches M on all entries with $i \neq j$ (as well as on $i = j$ and $k = l$). The real symmetric matrix M' is positive semidefinite if the Hermitian M is, and M' results in an equal objective value to that of M, hence the restriction to real moment matrices is without loss of generality.

We have thus far described the ideal solutions, M. Next, we add constraints to the SDP to help enforce this ideal interpretation of M:

1. For all $i \in [n], k \in [3]$, set $M(ik,ik) = 1$, since ideally $M(ik,ik) = \mathrm{tr}(\rho\sigma_k^i\sigma_k^i) = \mathrm{tr}(\rho) = 1$.
2. For all $i \in [n], k \neq l \in [3]$, set $M(ik,il) = -M(il,ik)$, since distinct Pauli matrices anti-commute.
3. Set $M \succeq 0$. This is since, ideally, for all $s \in \mathbb{R}^{3n}$, we have

$$s^T M s = \sum_{ijkl} s_{ik}s_{jl}M(ik,jl) = \mathrm{tr}\left(\rho\left(\sum_{ik} s_{ik}\sigma_k^i\right)\left(\sum_{jl} s_{jl}\sigma_l^j\right)\right) = \mathrm{tr}(\rho S^2) \geq 0, \tag{3.3}$$

   where $S := \sum_{ik} s_{ik}\sigma_k^i$, and since $\rho, S^2 \succeq 0$.

Finally, the relaxed objective function is obtained by replacing each term $\mathrm{tr}(\rho\sigma_k^i\sigma_l^j)$ with $M(ik,jl)$. For example, the relaxed objective function for $F_{1,1,1}$ becomes $\sum_{(i,j)\in E} w_{ij}(1 - M(i1,j1) - M(i2,j2) - M(i3,j3))$.

Let us remark that our formulation is essentially the first level $s = 1$ of the Lasserre SDP hierarchy. Higher levels $s > 1$ are obtained by considering s-local terms for the moment matrices, i.e. $M(i_1k_1,\ldots,i_sk_s) = \mathrm{tr}(\rho\sigma_{k_1}^{i_1}\cdots\sigma_{k_s}^{i_s})$.

Rounding solutions to the SDP Given any solution M to the SDP, we take the Cholesky decomposition of M to obtain a set of vectors $v_{ik} \in \mathbb{R}^{3n}$ for $i \in [n]$ and $k \in [3]$, such that $M(ik, jl) = v_{ik}^T v_{jl}$. Since $M(ik, ik) = 1$, each v_{ik} is a unit vector. Now, our aim is to round M to a product state solution $\rho_{\text{prod}} = \rho_1 \otimes \cdots \otimes \rho_n$ on n qubits. Thus, writing ρ_i in terms of its Bloch vector $\rho_i = (I + r_{i1}X + r_{i2}Y + r_{i3}Z)/2$ each v_{ik} should be thought of as a $3n$-dimensional relaxation of $r_{ik} \in \mathbb{R}$. For any $v \in \mathbb{R}^n$, $w \in \mathbb{R}^m$, define operation

$$v \circ w = \begin{cases} 0 & \text{if } v = 0 \text{ and } w = 0 \\ v & \text{if } v \neq 0 \text{ and } w = 0 \\ w & \text{if } w \neq 0 \text{ and } v = 0 \\ (v^T, w^T)^T & \text{otherwise,} \end{cases}$$

where $(v^T, w^T)^T \in \mathbb{R}^{n+m}$ denotes the concatenation of v and w. Recalling that $H \in F_{\alpha,\beta,\gamma}$ for $\alpha, \beta, \gamma \in \{0, 1\}$, we now set

$$u_i := (\alpha v_{i1}) \circ (\beta v_{i2}) \circ (\gamma v_{i3}) \in \mathbb{R}^{(\alpha+\beta+\gamma)n}.$$

This yields first that $w_{ij}(1 - u_i^T u_j)$ equals the term in the relaxed SDP objective function for H corresponding to edge $(i, j) \in E$. For example, if $H \in F_{1,1,0}$ (i.e. the local terms are $w_{ij}(I - X_iX_j - Y_iY_j)$), then $u_i \in \mathbb{R}^{2n}$ and for edge $(i, j) \in E$ we have $M(i1, j1) + M(i2, j2) = u_i^T u_j$. Second, we have $\|u_i\|_2 = \sqrt{\alpha + \beta + \gamma}$.

To obtain the desired claim, define now $x_i = u_i / \|u_i\|_2$. We use a generalization of the Goemans-Williamson (GW) [GW95] rounding procedure due to Briët, de Oliveira Filho and Vallentin [BOFV14]. Specifically, we randomly round each $x_i \in \mathbb{R}^{(\alpha+\beta+\gamma)n}$ to a Bloch vector $y_i \in \mathbb{R}^{\alpha+\beta+\gamma}$ as follows. Let M be a random $(\alpha + \beta + \gamma) \times (\alpha + \beta + \gamma)n$ matrix, each of whose entries is chosen independently from a standard normal distribution with mean 0 and variance 1. Then, for each i, set

$$y_i = Mx_i / \|Mx_i\|_2 \in \mathbb{R}^{\alpha+\beta+\gamma}.$$

We map this to a (pure) single-qubit state ρ_i as follows. Let $I(k)$ be the index in sequence (α, β, γ) of the kth non-zero entry (if it exists), for $k \in \{1, 2, 3\}$. Then, set the $I(k)$-th Bloch vector entry of ρ_i to $y_{i,k}$. For example, if $\alpha = \beta = \gamma = 1$, this yields $\rho_i = (I + y_{i,1}X + y_{i,2}Y + y_{i,3}Z)/2$, if $\alpha = \beta = 1$ and $\gamma = 0$, this yields $\rho_i = (I + y_{i,1}X + y_{i,2}Y)/2$, and if $\alpha = \beta = 0$ and $\gamma = 1$, this yields $\rho_i = (I + y_{i,1}Z)/2$ (note the subscript 1 in $y_{i,1}$). For ease of exposition, henceforth we refer to the Bloch vector for ρ_i as $\mathbf{r_i} = (r_1, r_2, r_3)$, where the entries of $\mathbf{r_i}$ which are not set in the rounding scheme above being implicitly set to 0. For example, $\mathbf{r_i} = (y_{i,1}, y_{i,2}, y_{i,3})$, $\mathbf{r_i} = (y_{i,1}, y_{i,2}, 0)$, and $\mathbf{r_i} = (0, 0, y_{i,1})$, respectively, in the examples above.

Approximation ratio. To analyze the approximation ratio obtained, note that for edge $(i,j) \in E$, we have

$$\begin{aligned} w_{ij}\operatorname{tr}(H_{ij}\rho_{\text{prod}}) &= \frac{w_{ij}}{4}\operatorname{tr}(H_{ij}(I + r_{i,1}X_i + r_{i,2}Y_i + r_{i,3}Z_i)(I + r_{j,1}X_j + r_{j,2}Y_j + r_{j,3}Z_j)) \\ &= \frac{w_{ij}}{4}\operatorname{tr}((I - \alpha X_iX_j - \beta Y_iY_j - \gamma Z_iZ_j)\cdot \\ &\qquad (I + r_{i,1}X_i + r_{i,2}Y_i + r_{i,3}Z_i)(I + r_{j,1}X_j + r_{j,2}Y_j + r_{j,3}Z_j)) \\ &= w_{ij}(1 - \alpha r_{i,1}r_{j,1} - \beta r_{i,2}r_{j,2} - \gamma r_{i,3}r_{j,3}) \\ &= w_{ij}(1 - y_i \cdot y_j). \end{aligned}$$

On the other hand, recall the SDP obtains value $w_{ij}(1 - u_i^T u_j)$ on edge $(i,j) \in E$. For brevity, let $F[u \cdot v]$ denote the right hand side of Equation B.6 (Lemma B.1 in Appendix B.2), such that Lemma B.1 says $\mathbb{E}[u \cdot v] = F[u \cdot v]$ (in the notation of Lemma B.1, $r = \alpha + \beta + \gamma$). Then, by linearity of expectation, the expected approximation ratio is given by the expected ratio attained on each edge, which is

$$\frac{1 - \mathbb{E}[y_i \cdot y_j]}{1 - u_i \cdot u_j} = \frac{1 - F[y_i \cdot y_j]}{1 - u_i \cdot u_j} = \frac{1 - F[y_i \cdot y_j]}{1 - (\alpha + \beta + \gamma)x_i \cdot x_j} = \frac{1 - F[t]}{1 - (\alpha + \beta + \gamma)t},$$

where we defined $t = x_i \cdot x_j$ (note the value of F only depends on t; see Appendix B.1). Numerically evaluating via Mathematica (see Appendix B.2 for Mathematica code)

$$\min_{t \in [-1, 1/(\alpha+\beta+\gamma))} \frac{1 - F[t]}{1 - (\alpha + \beta + \gamma)t},$$

we obtain ratios of 0.878 (for $\alpha + \beta + \gamma = 1$), 0.649 (for $\alpha + \beta + \gamma = 2$), and 0.498 (for $\alpha + \beta + \gamma = 3$), respectively. Note we minimize over $t \in [-1, 1/(\alpha + \beta + \gamma))$, since for $t \in [1/(\alpha + \beta + \gamma), 1]$ the ratio can only be negative (the denominator is negative, and the numerator is in range $[0, 2]$). This completes the proof. □

3.2.3 Approximation for a Broader Class of Hamiltonians

Here we extend the results of the previous section to address 2-local Hamiltonians of the form:

$$H_{ij} := I - \sum_{t \leq 3} \alpha_t (q_{t,1}X_i + q_{t,2}Y_i + q_{t,3}Z_i)(q_{t,1}X_j + q_{t,2}Y_j + q_{t,3}Z_j),$$

with $\alpha_t \in \{0, 1\}$, $\sum_{k \leq 3} q_{t,k}^2 = 1$ for all t, and $\sum_{k \leq 3} q_{s,k}q_{t,k} = 0$ for distinct s, t. The variables $\alpha_1, \alpha_2, \alpha_3$ play the same role as α, β, γ from the previous section, and the approximation ratio depends on $\sum_t \alpha_t$ and as indicated in Theorem 3.2. For example, we get a 0.878-approximation for the Hamiltonian $I - \frac{1}{2}(X_iX_j + X_iZ_j + Z_iX_j + Z_iZ_j)$.

Given a Hamiltonian as above, we define vectors $q_t := (q_{t,1}, q_{t,2}, q_{t,3}) \in \mathbb{R}^3$ and construct a matrix $Q := \sum_{t \leq 3} \alpha_t q_t q_t^T \in \mathbb{R}^{3 \times 3}$. Observe that by the conditions on α_t and q_t, Q is an orthogonal

projector. We diagonalize, $Q = PDP^T$, where $P \in \mathbb{R}^{3\times 3}$ is an orthogonal matrix, and $D \in \mathbb{R}^{3\times 3}$ is a diagonal matrix with entries in $\{0,1\}$.

Suppose we are given an instance on an interaction graph G with Hamiltonian H_{ij} acting on qubits i and j for all $(i,j) \in E$. We appeal to the semidefinite relaxation of the previous section; however, the objective function will be based on the Hamiltonian $H := \sum_{(i,j)\in E} H_{ij}$. We employ the vector perspective of the SDP relaxation, where the variables are vectors $v_{ik} \in \mathbb{R}^n$. The index i corresponds to a qubit, and k indicates one of the three Pauli bases, as in the previous section. The v_{ik} are related to the moment matrix M by a Cholesky decomposition. For convenience, we will also define $V_i \in \mathbb{R}^{n\times 3}$ as the matrix whose kth column is v_{ik}. The following is equivalent to the relaxation from the previous section:

$$\begin{aligned}
\max \sum_{(i,j)\in E} (1 - \sum_{k,l\leq 3} (\sum_{t\leq 3} \alpha_t q_{t,k} q_{t,l}) v_{ik}^T v_{jl}) &= \sum_{(i,j)\in E} (1 - \operatorname{tr}\big(QV_i^T V_j\big)) \\
v_{ik}^T v_{ik} &= 1, \text{ for all } i \in [n], k \in [3] \\
v_{ik} &\in \mathbb{R}^n, \text{ for all } i \in [n], k \in [3].
\end{aligned}$$

We henceforth let v_{ik} refer to an optimal solution to the above relaxation.

In order to appeal to the rounding algorithm devised in the previous section, we introduce a modified Hamiltonian of form (3.1), as previously analyzed. In particular we replace each H_{ij} with $\widetilde{H}_{ij} := I - D_{1,1}X_iX_j - D_{2,2}Y_iY_j - D_{3,3}Z_iZ_j$. This yields a relaxation akin to the above with objective $\sum_{(i,j)\in E}(1 - \operatorname{tr}\Big(D\widetilde{V}_i^T\widetilde{V}_j\Big))$, where $\tilde{v}_{ik} \in \mathbb{R}^n$ refer to a set of feasible vectors for the above relaxation, and $\widetilde{V}_i \in \mathbb{R}^{n\times 3}$ has $\tilde{v}_{ik}$ as its kth column.

Our algorithm is as follows.

1. Obtain optimal vectors $\tilde{v}_{ik} \in \mathbb{R}^n$ for the relaxation with objective $\widetilde{H}$.
2. Employ the rounding algorithm from the previous section on the $\tilde{v}_{ik}$ to yield Bloch vectors $\tilde{y}_i \in \mathbb{R}^3$.
3. Output the Bloch vectors $y_i := P\tilde{y}_i \in \mathbb{R}^3$.

To see that this algorithm produces a product state matching the approximation ratios from Theorem 3.2, we observe that $\operatorname{tr}\big(QV_i^T V_j\big) = \operatorname{tr}\big(PDP^T V_i^T V_j\big) = \operatorname{tr}\big(D(V_iP)^T(V_jP)\big)$. This implies that if the V_i represent a feasible solution to the relaxation with objective H, then setting $\widetilde{V}_i := V_iP$ yields a solution of the same objective value for the relaxation with objective $\widetilde{H}$ (recall that P is orthogonal). The other direction holds as well, and consequently, the relaxations with objectives H and $\widetilde{H}$ must have equal optimal values. A similar argument shows that the y_i produced in Step 3 of the algorithm are feasible for the relaxation with objective H and have the same cost that the $\tilde{y}_i$ have for the relaxation with objective $\widetilde{H}$.

This page intentionally left blank.

Chapter 4

Conclusion

Through two bodies of complementary work, we have demonstrated that (1) advances in quantum algorithms can be brought to bear to provide rigorous approximations to discrete optimization problems, and (2) advances in discrete optimization can be brought to bear in approximating low energy states of quantum systems. We feel that our work is a prelude to deeper interactions between discrete optimization and quantum computation. Aside from technical hurdles, there is an inherent challenge in bridging two fields with rich but disparate cultures. We have used the expertise developed within this project to secure follow-on opportunities to continue bridging discrete optimization and quantum information science.

This page intentionally left blank.

References

[Aar05] Scott Aaronson. “NP-complete Problems and Physical Reality”. In: (2005). arXiv: quant-ph/0502072.

[Aar13] Scott Aaronson. *Quantum Computing since Democritus*. Cambridge University Press, 2013. DOI: 10.1017/CBO9780511979309.

[AAV13] D. Aharonov, I. Arad, and T. Vidick. “The quantum PCP conjecture”. In: *ACM SIGACT News*. Vol. 44. 2. 2013, pp. 47–79.

[Aha03] Dorit Aharonov. *A Simple Proof that Toffoli and Hadamard are Quantum Universal*. 2003. arXiv: quant-ph/0301040.

[Aha+04] Dorit Aharonov, Wim van Dam, Julia Kempe, Zeph Landau, Seth Lloyd, and Oded Regev. “Adiabatic Quantum Computation is Equivalent to Standard Quantum Computation”. In: (2004). arXiv: quant-ph/0405098.

[Aha+09] D. Aharonov, I. Arad, Z. Landau, and U. Vazirani. “The detectibility lemma and quantum gap amplification”. In: *Proceedings of 41st ACM Symposium on Theory of Computing (STOC 2009)*. Vol. 287. 2009, pp. 417–426.

[AL16] Tameem Albash and Daniel A. Lidar. *Adiabatic Quantum Computing*. 2016. arXiv: 1611.04471.

[AL98] Daniel S. Abrams and Seth Lloyd. “Nonlinear quantum mechanics implies polynomial-time solution for NP-complete and #P problems”. In: (1998). DOI: 10.1103/PhysRevLett.81.3992. arXiv: quant-ph/9801041.

[Aro+98] S. Arora, C. Lund, R. Motwani, M. Sudan, and M. Szegedy. “Proof verification and the hardness of approximation problems”. In: 45.3 (1998). Prelim. version FOCS ’92, pp. 501–555.

[AS98] S. Arora and S. Safra. “Probabilistic checking of proofs: A new characterization of NP”. In: 45.1 (1998). Prelim. version FOCS ’92, pp. 70–122.

[Bar+14] Stefanie Barz, Ivan Kassal, Martin Ringbauer, Yannick Ole Lipp, Borivoje Dakic, Alán Aspuru-Guzik, and Philip Walther. “A two-qubit photonic quantum processor and its application to solving systems of linear equations”. In: *Scientific Reports* 4 (Aug. 2014). Article, p. 6115. arXiv: 1302.1210. URL: http://dx.doi.org/10.1038/srep06115.

[Bar+15] Boaz Barak, Ankur Moitra, Ryan O’Donnell, Prasad Raghavendra, Oded Regev, David Steurer, Luca Trevisan, Aravindan Vijayaraghavan, David Witmer, and John Wright. *Beating the random assignment on constraint satisfaction problems of bounded degree*. 2015. arXiv: 1505.03424.

[Bar+95] Adriano Barenco, Charles H. Bennett, Richard Cleve, David P. DiVincenzo, Norman Margolus, Peter Shor, Tycho Sleator, John A. Smolin, and Harald Weinfurter. "Elementary gates for quantum computation". In: *Phys. Rev. A* 52.5 (Nov. 1995), pp. 3457–3467. DOI: 10.1103/PhysRevA.52.3457. arXiv: quant-ph/9503016.

[BBT09] N. Bansal, S. Bravyi, and B. M. Terhal. "Classical approximation schemes for the ground-state energy of quantum and classical Ising spin Hamiltonians on planar graphs". In: 9.7&8 (2009), pp. 0701–0720.

[Bet31] H. Bethe. "Zur Theorie der Metalle". In: *Zeitschrift für Physik* 71.3–4 (1931), pp. 205–226.

[BF28] M. Born and V. Fock. "Beweis des Adiabatensatzes". In: *Zeitschrift für Physik* 51 (Mar. 1928), pp. 165–180. DOI: 10.1007/BF01343193.

[BG16] S. Bravyi and D. Gosset. "Polynomial-time classical simulation of quantum ferromagnets". Available at arXiv.org e-Print quant-ph/1612.05602. 2016.

[BH13] F. Brandão and A. Harrow. "Product-state Approximations to Quantum Ground States". In: *Proceedings of the 45th ACM Symposium on the Theory of Computing (STOC 2013)*. 2013, pp. 871–880.

[BH14] S. Bravyi and M. Hastings. "On complexity of the quantum Ising model". In: *Communications in Mathematical Physics* 349.1 (2014), pp. 1–45.

[Bia08] J. D. Biamonte. "Nonperturbative k-body to two-body commuting conversion Hamiltonians and embedding problem instances into Ising spins". In: *Phys. Rev. A* 77.5 (May 2008), p. 052331. DOI: 10.1103/PhysRevA.77.052331. arXiv: 0801.3800.

[BOFV14] J. Briët, F. M. de Oliveira Filho, and F. Vallentin. "Grothendieck inequalities for semidefinite programs with rank constraint". In: *Theory of Computing* 10 (2014), pp. 77–105.

[Bor26] Max Born. "Zur Quantenmechanik der Stoßvorgänge". In: *Zeitschrift für Physik* 37.12 (Dec. 1926), pp. 863–867. ISSN: 0044-3328. DOI: 10.1007/BF01397477. URL: https://doi.org/10.1007/BF01397477.

[Boy+99] P. Oscar Boykin, Tal Mor, Matthew Pulver, Vwani Roychowdhury, and Farrokh Vatan. *On Universal and Fault-Tolerant Quantum Computing*. 1999. DOI: 10.1109/SFFCS.1999.814621. arXiv: quant-ph/9906054.

[Bra15] S. Bravyi. "Monte Carlo simulation of stoquastic Hamiltonians". In: 15.13&14 (2015), pp. 1122–1140.

[Bra+18] S. Bravyi, D. Gosset, R. Koenig, and K. Temme. "Approximation algorithms for quantum many-body problems". Available at arXiv.org e-Print quant-ph/arXiv:1808.01734. 2018.

[Bre+98] Gavin K. Brennen, Carlton M. Caves, Poul S. Jessen, and Ivan H. Deutsch. "Quantum Logic Gates in Optical Lattices". In: (1998). DOI: 10.1103/PhysRevLett.82.1060. arXiv: quant-ph/9806021.

[Bri+09] H. J. Briegel, D. E. Browne, W. Dür, R. Raussendorf, and M. Van den Nest. "Measurement-based quantum computation". In: (2009). DOI: 10.1038/nphys1157. arXiv: 0910.1116.

[BW08] Rainer Blatt and David Wineland. “Entangled states of trapped atomic ions”. In: *Nature* 453 (June 2008), p. 1008. URL: http://dx.doi.org/10.1038/nature07125.

[CFP01] Andrew M. Childs, Edward Farhi, and John Preskill. “Robustness of adiabatic quantum computation”. In: *Phys. Rev. A* 65.1 (Dec. 2001), p. 012322. DOI: 10.1103/PhysRevA.65.012322. arXiv: quant-ph/0108048.

[CH13] Lilian Childress and Ronald Hanson. “Diamond NV centers for quantum computing and quantum networks”. In: *MRS Bulletin* 38.2 (2013), pp. 134–138. DOI: 10.1557/mrs.2013.20.

[CM16] T. Cubitt and A. Montanaro. “Complexity classification of local Hamiltonian problems”. In: 45.2 (2016), pp. 268–316.

[Col+18] Patrick J. Coles, Stephan Eidenbenz, Scott Pakin, Adetokunbo Adedoyin, John Ambrosiano, Petr Anisimov, William Casper, Gopinath Chennupati, Carleton Coffrin, Hristo Djidjev, David Gunter, Satish Karra, Nathan Lemons, Shizeng Lin, Andrey Lokhov, Alexander Malyzhenkov, David Mascarenas, Susan Mniszewski, Balu Nadiga, Dan O’Malley, Diane Oyen, Lakshman Prasad, Randy Roberts, Phil Romero, Nandakishore Santhi, Nikolai Sinitsyn, Pieter Swart, Marc Vuffray, Jim Wendelberger, Boram Yoon, Richard Zamora, and Wei Zhu. *Quantum Algorithm Implementations for Beginners*. 2018. arXiv: 1804.03719.

[Deu89] David Deutsch. “Quantum computational networks”. In: *Proc. Roy. Soc. London A* 425.1868 (Sept. 1989), pp. 73–90. URL: http://www.jstor.org/stable/2398494.

[Die82] D. Dieks. “Communication by EPR devices”. In: *Phys. Lett. A* 92.6 (Nov. 1982), pp. 271–272. DOI: 10.1016/0375-9601(82)90084-6.

[Dir39] P. A. M. Dirac. “A new notation for quantum mechanics”. In: *Mathematical Proceedings of the Cambridge Philosophical Society* 35.3 (1939), pp. 416–418. DOI: 10.1017/S0305004100021162.

[DiV95] David P. DiVincenzo. “Two-bit gates are universal for quantum computation”. In: *Phys. Rev. A* 51.2 (Feb. 1995), pp. 1015–1022. DOI: 10.1103/PhysRevA.51.1015. arXiv: cond-mat/9407022.

[DS14] Ish Dhand and Barry C. Sanders. “Stability of the Trotter-Suzuki decomposition”. In: (2014). DOI: 10.1088/1751-8113/47/26/265206. arXiv: 1403.3469.

[Dut+07] M. V. Gurudev Dutt, L. Childress, L. Jiang, E. Togan, J. Maze, F. Jelezko, A. S. Zibrov, P. R. Hemmer, and M. D. Lukin. “Quantum Register Based on Individual Electronic and Nuclear Spin Qubits in Diamond”. In: *Science* 316.5829 (2007), pp. 1312–1316. ISSN: 00368075, 10959203. URL: http://www.jstor.org/stable/20036388.

[Far+00] Edward Farhi, Jeffrey Goldstone, Sam Gutmann, and Michael Sipser. *Quantum computation by adiabatic evolution*. 2000. arXiv: quant-ph/0001106.

[Far+01] Edward Farhi, Jeffrey Goldstone, Sam Gutmann, Joshua Lapan, Andrew Lundgren, and Daniel Preda. “A quantum adiabatic evolution applied to random instances of an NP-complete problem”. In: *Science* 292.5516 (Apr. 2001), pp. 472–475. DOI: 10.1126/science.1057726. arXiv: quant-ph/0104129.

[Fey82] Richard P. Feynman. “Simulating physics with computers”. In: *Int. J. Theo. Phys.* 21.6/7 (1982), pp. 467–488. DOI: 10.1007/BF02650179.

[Fey98] Richard Phillips Feynman. *Feynman Lectures on Computation*. Ed. by J. G. Hey and Robin W. Allen. Boston, MA, USA: Addison-Wesley Longman Publishing Co., Inc., 1998. ISBN: 0201386283.

[FGG14a] Edward Farhi, Jeffrey Goldstone, and Sam Gutmann. *A Quantum Approximate Optimization Algorithm Applied to a Bounded Occurrence Constraint Problem*. 2014. arXiv: 1412.6062.

[FGG14b] Edward Farhi, Jeffrey Goldstone, and Sam Gutmann. *A Quantum Approximate Optimization Algorithm*. 2014. arXiv: 1411.4028.

[FH16] Edward Farhi and Aram W Harrow. *Quantum Supremacy through the Quantum Approximate Optimization Algorithm*. 2016. arXiv: 1602.07674.

[Fro12] Georg Frobenius. *Ueber Matrizen aus nicht negativen Elementen*. May 1912.

[Gha+14] Sevag Gharibian, Yichen Huang, Zeph Landau, and Seung Woo Shin. “Quantum Hamiltonian Complexity”. In: *Foundations and Trends® in Theoretical Computer Science* 10.3 (2014), pp. 159–282. ISSN: 1551-305X. DOI: 10.1561/0400000066. URL: http://dx.doi.org/10.1561/0400000066.

[GK11] S. Gharibian and J. Kempe. “Approximation algorithms for QMA-complete problems”. In: *Proceedings of 26th IEEE Conference on Computational Complexity (CCC 2011)*. DOI: 10.1109/CCC.2011.15, © 2011 IEEE, ieeexplore.ieee.org. 2011, pp. 178–188.

[GK12] S. Gharibian and J. Kempe. “Hardness of approximation for quantum problems”. In: *Proceedings of 39th International Colloquium on Automata, Languages and Programming (ICALP 2012)*. DOI: 10.1007/978-3-642-31594-7, © 2012 Springer. 2012, pp. 387–398.

[GW94] M. Goemans and D. Williamson. “New $\frac{3}{4}$-approximation algorithms for the maximum satisfiability problem”. In: *SIAM Journal on Discrete Mathematics* 7 (1994), pp. 656–666.

[GW95] M. Goemans and D. Williamson. “Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming”. In: 42 (1995), pp. 1115–1145.

[GZ18] M. Fernando Gonzalez-Zalba. “Solid-state qubits”. In: (2018). arXiv: 1801.06722.

[Had18] Stuart Hadfield. *On the representation of Boolean and real functions as Hamiltonians for quantum computing*. 2018. arXiv: 1804.09130.

[Had+17] Stuart Hadfield, Zhihui Wang, Bryan O’Gorman, Eleanor G. Rieffel, Davide Venturelli, and Rupak Biswas. *From the Quantum Approximate Optimization Algorithm to a Quantum Alternating Operator Ansatz*. 2017. arXiv: 1709.03489.

[Hir+14] Juho Hirvonen, Joel Rybicki, Stefan Schmid, and Jukka Suomela. “Large Cuts with Local Algorithms on Triangle-Free Graphs”. In: (2014). arXiv: 1402.2543.

[HLZ02] Eran Halperin, Dror Livnat, and Uri Zwick. *MAX CUT in Cubic Graphs*. Philadelphia, PA, USA, 2002. URL: http://dl.acm.org/citation.cfm?id=545381.545449.

[Hoc97] D. Hochbaum. *Approximation Algorithms for NP-Hard Problems*. Wadsworth Publishing Company, 1997.

[Hol73] Alexander S. Holevo. "Bounds for the quantity of information transmitted by a quantum communication channel". In: *Problems of Information Transmission* 9.3 (1973), pp. 177–183. URL: http://www.mathnet.ru/php/archive.phtml?wshow=paper\&\#38;jrnid=ppi\&\#38;paperid=903\&\#38;option_lang=eng.

[HS05] Naomichi Hatano and Masuo Suzuki. "Finding Exponential Product Formulas of Higher Orders". In: (2005). DOI: 10.1007/11526216_2. arXiv: math-ph/0506007.

[Hum+18] Travis S. Humble, Himanshu Thapliyal, Edgard Munoz-Coreas, Fahd A. Mohiyaddin, and Ryan S. Bennink. *Quantum Computing Circuits and Devices*. 2018. arXiv: 1804.10648.

[Hås00] Johan Håstad. "On Bounded Occurrence Constraint Satisfaction". In: *Inf. Process. Lett.* 74.1-2 (Apr. 2000), pp. 1–6. ISSN: 0020-0190. DOI: 10.1016/S0020-0190(00)00032-6. URL: http://dx.doi.org/10.1016/S0020-0190(00)00032-6.

[Jak+99] D. Jaksch, H.-J. Briegel, J. I. Cirac, C. W. Gardiner, and P. Zoller. "Entanglement of atoms via cold controlled collisions". In: *Phys. Rev. Lett.* 82.9 (Mar. 1999), pp. 1975–1978. DOI: 10.1103/PhysRevLett.82.1975. arXiv: quant-ph/9810087.

[Jor09] Stephen Jordan. *Quantum Algorithm Zoo*. Site moved to/hoted by nist.gov in 2011. 2009. URL: https://math.nist.gov/quantum/zoo/ (visited on 05/01/2018).

[Joz02] Richard Jozsa. *A stronger no-cloning theorem*. 2002. arXiv: quant-ph/0204153.

[Joz05] Richard Jozsa. *An introduction to measurement based quantum computation*. 2005. arXiv: quant-ph/0508124.

[JRW17] Zhang Jiang, Eleanor G. Rieffel, and Zhihui Wang. *A QAOA-inspired circuit for Grover's unstructured search using a transverse field*. 2017. arXiv: 1702.02577.

[KM15] George C. Knee and William J. Munro. "Optimal Trotterization in universal quantum simulators under faulty control". In: (2015). DOI: 10.1103/PhysRevA.91.052327. arXiv: 1502.04536.

[Kok+05] Pieter Kok, W. J. Munro, Kae Nemoto, T. C. Ralph, Jonathan P. Dowling, and G. J. Milburn. "Review article: Linear optical quantum computing". In: (2005). DOI: 10.1103/RevModPhys.79.135. arXiv: quant-ph/0512071.

[KSV02] A. Kitaev, A. Shen, and M. Vyalyi. *Classical and Quantum Computation*. American Mathematical Society, 2002.

[Lan18] J. M. Landsberg. *A very brief introduction to quantum computing and quantum information theory for mathematicians*. 2018. arXiv: 1801.05893.

[LB13] Daniel A. Lidar and Todd A. Brun, eds. *Quantum Error Correction*. Cambridge University Press, 2013. DOI: 10.1017/CBO9781139034807.

[LD97] Daniel Loss and David P. DiVincenzo. “Quantum Computation with Quantum Dots”. In: (1997). DOI: 10.1103/PhysRevA.57.120. arXiv: cond-mat/9701055.

[Llo96] Seth Lloyd. “Universal quantum simulators”. In: *Science* 273.5278 (Aug. 1996), pp. 1073–1078. DOI: 10.1126/science.273.5278.1073.

[Loc82] S. C. Locke. “Maximum kcolorable subgraphs”. In: *Journal of Graph Theory* 6.2 (1982), pp. 123–132. DOI: 10.1002/jgt.3190060206. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/jgt.3190060206. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/jgt.3190060206.

[LZ16] Cedric Yen-Yu Lin and Yechao Zhu. *Performance of QAOA on Typical Instances of Constraint Satisfaction Problems with Bounded Degree*. 2016. arXiv: 1601.01744.

[Mal+15] K. M. Maller, M. T. Lichtman, T. Xia, Y. Sun, M. J. Piotrowicz, A. W. Carr, L. Isenhower, and M. Saffman. “A Rydberg blockade CNOT gate and entanglement in a 2D array of neutral atom qubits”. In: (2015). DOI: 10.1103/PhysRevA.92.022336. arXiv: 1506.06416.

[Mer07] N. David Mermin. *Quantum Computer Science: An Introduction*. New York, NY, USA: Cambridge University Press, 2007. ISBN: 0521876583.

[MM11] Cristopher Moore and Stephan Mertens. *The Nature of Computation*. New York, NY, USA: Oxford University Press, Inc., 2011. ISBN: 0199233217, 9780199233212.

[Mon15] Ashley Montanaro. “Quantum algorithms: an overview”. In: (2015). DOI: 10.1038/npjqi.2015.23. arXiv: 1511.04206.

[MR95] Rajeev Motwani and Prabhakar Raghavan. *Randomized Algorithms*. New York, NY, USA: Cambridge University Press, 1995. ISBN: 0-521-47465-5, 9780521474658.

[MSS00] Yuriy Makhlin, Gerd Schoen, and Alexander Shnirman. “Quantum state engineering with Josephson-junction devices”. In: (2000). DOI: 10.1103/RevModPhys.73.357. arXiv: cond-mat/0011269.

[Muk08] Shubu Mukherjee. *Architecture Design for Soft Errors*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2008. ISBN: 9780080558325, 9780123695291.

[MW18] Samuel Marsh and Jingbo Wang. *A quantum walk assisted approximate algorithm for bounded NP optimisation problems*. 2018. arXiv: 1804.08227.

[Nan17] Giacomo Nannicini. *An Introduction to Quantum Computing, Without the Physics*. 2017. arXiv: 1708.03684.

[NC11] Michael A. Nielsen and Isaac L. Chuang. *Quantum Computation and Quantum Information: 10th Anniversary Edition*. 10th. Cambridge University Press, 2011. ISBN: 1107002176, 9781107002173.

[NV18] A. Natarajan and T. Vidick. “Low-degree testing for quantum states, and a quantum entangled games PCP for QMA”. Available at arXiv.org e-Print quant-ph/1801.03821. 2018.

[Osb12] T. J. Osborne. “Hamiltonian complexity”. In: *Reports on Progress in Physics* 75.2 (2012), p. 022001. URL: http://stacks.iop.org/0034-4885/75/i=2/a=022001.

[OT08] R. Oliveira and B. M. Terhal. “The complexity of quantum spin systems on a two-dimensional square lattice”. In: 8.10 (2008), pp. 0900–0924.

[Per07] Oskar Perron. “Zur Theorie der Matrices”. In: *Mathematische Annalen* 64.2 (June 1907), pp. 248–263. ISSN: 0025-5831. DOI: 10.1007/bf01449896. URL: http://dx.doi.org/10.1007/bf01449896.

[PM17] Stephen Piddock and Ashley Montanaro. “The Complexity of Antiferromagnetic Interactions and 2D Lattices”. In: *Quantum Info. Comput.* 17.7-8 (June 2017), pp. 636–672. ISSN: 1533-7146. URL: http://dl.acm.org/citation.cfm?id=3179553.3179559.

[Pou+11] David Poulin, Angie Qarry, R. D. Somma, and Frank Verstraete. “Quantum Simulation of Time-dependent Hamiltonians and the Convenient Illusion of Hilbert Space”. In: (2011). DOI: 10.1103/PhysRevLett.106.170501. arXiv: 1102.1360.

[Pre18] John Preskill. *Quantum Computing in the NISQ era and beyond*. 2018. arXiv: 1801.00862.

[Pre98] John Preskill. *Lecture notes for Caltech Ph 219: Quantum Information and Computation*. 1998. URL: http://www.theory.caltech.edu/~preskill/ph219/.

[RA18] Ciaran Ryan-Anderson. “Quantum Algorithms, Architecture, and Error Correction”. PhD thesis. University of New Mexico, 2018. URL: https://arxiv.org/abs/1812.04735.

[RB01] Robert Raussendorf and Hans J. Briegel. “A one-way quantum computer”. In: *Phys. Rev. Lett.* 86.22 (May 2001), pp. 5188–5191. DOI: 10.1103/PhysRevLett.86.5188.

[RG02] Terry Rudolph and Lov Grover. *A 2 rebit gate universal for quantum computing*. 2002. arXiv: quant-ph/0210187.

[RP11] Eleanor Rieffel and Wolfgang Polak. *Quantum Computing: A Gentle Introduction*. 1st. The MIT Press, 2011. ISBN: 9780262015066.

[RP98] Eleanor G. Rieffel and Wolfgang Polak. “An Introduction to Quantum Computing for Non-Physicists”. In: (1998). arXiv: quant-ph/9809016.

[She13] Henry Maurice Sheffer. “A set of five indepedent postulates for Boolean algebras, with application to logical constants”. In: *Transactions of the American Mathematical Society* 14.4 (Apr. 1913), pp. 481–481. DOI: 10.1090/s0002-9947-1913-1500960-1. URL: https://doi.org/10.1090/s0002-9947-1913-1500960-1.

[She92] James B. Shearer. “A note on bipartite subgraphs of trianglefree graphs”. In: *Random Structures & Algorithms* 3.2 (1992), pp. 223–226. DOI: 10.1002/rsa.3240030211. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/rsa.3240030211. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/rsa.3240030211.

[Shi03] Yaoyun Shi. “Both Toffoli and controlled-NOT need little help to do universal quantum computation”. In: *Quant. Inf. Comp.* 3.1 (Jan. 2003), pp. 84–92. arXiv: quant-ph/0205115. URL: http://www.rintonpress.com/xqic3/qic-3-1/084-092.ps.

[Sho00] Peter W. Shor. *Introduction to Quantum Algorithms*. 2000. arXiv: quant-ph/0005003.

[Sin+10] Kilian Singer, Ulrich Poschinger, Michael Murphy, Peter Ivanov, Frank Ziesel, Tommaso Calarco, and Ferdinand Schmidt-Kaler. "Colloquium: Trapped ions as quantum bits: Essential numerical tools". In: *Rev. Mod. Phys.* 82 (3 Sept. 2010), pp. 2609–2632. DOI: 10.1103/RevModPhys.82.2609. arXiv: 0912.0196. URL: https://link.aps.org/doi/10.1103/RevModPhys.82.2609.

[Sku02] San Skulrattanakulchai. Δ-*List Vertex Coloring in Linear Time*. Ed. by Martti Penttonen and Erik Meineche Schmidt. Berlin, Heidelberg, 2002.

[Suz91] Masuo Suzuki. *General theory of fractal path integrals with applications to many-body theories and statistical physics*. 1991. DOI: 10.1063/1.529425. eprint: https://doi.org/10.1063/1.529425. URL: https://doi.org/10.1063/1.529425.

[Tre01] Luca Trevisan. *Non-approximability Results for Optimization Problems on Bounded Degree Instances*. New York, NY, USA, 2001. DOI: 10.1145/380752.380839. URL: http://doi.acm.org/10.1145/380752.380839.

[Tro59] H. F. Trotter. "On the Product of Semi-Groups of Operators". In: *Proceedings of the American Mathematical Society* 10.4 (1959), pp. 545–551. ISSN: 00029939, 10886826. URL: http://www.jstor.org/stable/2033649.

[Vaz01] V. Vazirani. *Approximation Algorithms*. Springer, 2001.

[Wan+17] Zhihui Wang, Stuart Hadfield, Zhang Jiang, and Eleanor G. Rieffel. "The Quantum Approximation Optimization Algorithm for MaxCut: A Fermionic View". In: (2017). DOI: 10.1103/PhysRevA.97.022304. arXiv: 1706.02998.

[Wen16] G. Wendin. "Quantum information processing with superconducting circuits: a review". In: (2016). DOI: 10.1088/1361-6633/aa7e1a. arXiv: 1610.02208.

[Wie+08] Nathan Wiebe, Dominic W. Berry, Peter Hoyer, and Barry C. Sanders. "Higher Order Decompositions of Ordered Operator Exponentials". In: (2008). DOI: 10.1088/1751-8113/43/6/065203. arXiv: 0812.0562.

[Wil13] Mark M. Wilde. *Quantum Information Theory*. Cambridge University Press, 2013. DOI: 10.1017/CBO9781139525343.

[Woo18] James Wootton. *Why is it crucial that the initial Hamiltonian does not commute with the final Hamiltonian in adiabatic quantum computation?* 2018. eprint: https://quantumcomputing.stackexchange.com/a/1941. URL: https://quantumcomputing.stackexchange.com/a/1941.

[WZ82] W.K. Wootters and W.H. Zurek. "A single quantum cannot be cloned". In: *Nature* 299 (Oct. 1982), pp. 802–803. DOI: 10.1038/299802a0.

[Yao+12] Norman Y. Yao, Liang Jiang, Alexey V. Gorshkov, Peter C. Maurer, Geza Giedke, J. Ignacio Cirac, and Mikhail D. Lukin. "Scalable architecture for a room temperature solid-state quantum information processor". In: *Nat. Comm.* 3 (2012), p. 800. DOI: 10.1038/ncomms1788. arXiv: 1012.2864.

[Zwa+12] Floris A. Zwanenburg, Andrew S. Dzurak, Andrea Morello, Michelle Y. Simmons, Lloyd C. L. Hollenberg, Gerhard Klimeck, Sven Rogge, Susan N. Coppersmith, and Mark A. Eriksson. “Silicon Quantum Electronics”. In: (2012). DOI: 10.1103/RevModPhys.85.961. arXiv: 1206.5202.

This page intentionally left blank.

Appendix A

A Brief Introduction to Quantum Computation

This chapter, appearing Ciaran Ryan-Anderson's Ph.D. thesis [RA18], presents the notation used in this dissertation by giving a brief introduction to the language of quantum computation (QC)—quantum mechanics. As the primary intent of this chapter is to present notation, the style of this chapter will be that of giving a high-level survey rather than discussing the subject in a rigorous, axiomatic fashion. Therefore, this chapter may serve as a quick overview of quantum computation for those outside the field.

For curious readers who might be interested in arguments about why nature's choice of the axioms of quantum mechanics might make sense, see Aaronson's book [Aar13]. For short introductions to QC for non-physicists see [RP98], [Nan17], and [Lan18]. More comprehensive presentations of QC can be found in works such as [Mer07; RP11; NC11; Wil13; Pre98].

The organization of this chapter is as follows: In Section A.1, I introduce the representation of noiseless classical and quantum states, *i.e.*, bits and qubits. Then, I discuss the evolution and measurement of qubits. In Section A.2, I present how quantum states and operations are represented pictorially. In Section A.3, I discuss a formalism for describing noisy quantum-systems. Finally, on Section A.4 I give a brief comparison between classical and quantum computation.

A.1 Ideal Classical and Quantum Systems

A.1.1 Classical States

Before discussing quantum systems, I will start by presenting how the states of digital computers[1], which are so ubiquitous in modern life, are represented. The fundamental unit of information in classical information theory is the binary digit, or *bit*. This unit takes the value of 0 or 1; therefore, a single bit is represented as a one-dimensional vector of the set $\{0,1\}$. Alternatively, a bit is represented as an element of $\{\text{TRUE},\text{FALSE}\}$ or $\{+1,-1\}$.

[1] By "digital computer," unless otherwise indicated, I will mean a binary, transistor-based computing device with finite memory.

Physically, bits can be represented by two-level classical systems such as the on-off states of switches, the up-down spins of magnetic domains, or the not-charged (1) or charged (0) states of floating gate transistors in NAND solid-state drives (SSDs).

In the parlance of theoretical computer science, digital computers are finite-state machines (although given the large size of a typical digital computer's memory, they are practically equivalent to Turing machines, whose memory is formally infinite). The state of these devices is represented as a *binary vector*, or *binary string*, $z \in \{0,1\}^n$. Here $\{0,1\}^n$ is the set formed by taking the n-fold Cartesian product of $\{0,1\}$. Often, a sequence of bits is presented as a concatenation

$$z = z_1 z_2 \cdots z_n, \tag{A.1}$$

where $z_i \in \{0,1\}$. For example, $z = 101010 = (1,0,1,0,1,0) \in \{0,1\}^6$.

A.1.2 Quantum States

The quantum analog of the bit is the quantum bit, or *qubit*. A pure qubit state is represented by a two-dimensional vector in $\mathbb{C}^2$. As I discuss later, a norm is defined for the complex vector spaces of which quantum states are elements. Thus, such a space is a Hilbert space (commonly denoted as $\mathscr{H}$). Qubits can represent any physical two-level quantum-system, which include the spin of an electron, the ground and excited energy-levels of an atom, or the polarization of a photon.

Qubits may be generalized to d-level quantum states, known as *qudits*, or even to non-discrete, infinite-dimensional states; however, this work focuses on qubit systems.

A standard choice for the basis states of a two-level quantum system is described by the notation $|0\rangle$ and $|1\rangle$, known as the *computational basis*. Here the ket, $|\cdot\rangle$, of the Dirac notation [Dir39] is used to represent vectors. The notation was developed to facilitate expressing the linear algebra of Hilbert space. Note that the Dirac notation can be used to represent binary vectors as well. For example, a bit can be represented as an element in $\{|0\rangle, |1\rangle\}$, and a bit string z is equivalent to $|z\rangle$. However, kets tend to be reserved for indicating that a vector is a member of a Hilbert space rather than just an element of $\{0,1\}^n$.

Besides the ket of the Dirac notation, computational-basis states can also be represented as column vectors, where

$$|0\rangle = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \text{ and } |1\rangle = \begin{bmatrix} 0 \\ 1 \end{bmatrix}. \tag{A.2}$$

Written in terms of these basis states, a general pure state $|\psi\rangle$ can then be described as a vector

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle = \begin{bmatrix} \alpha \\ \beta \end{bmatrix}, \tag{A.3}$$

where the coefficients $\alpha, \beta \in \mathbb{C}$ and $|\alpha|^2 + |\beta|^2 = 1$. The last condition is known as *normalization*. As we will discuss later, normalization is a useful condition that allows the complex coefficients of the basis states to be related to the probability of measurement outcomes. Note that these coefficients are often referred to as *amplitudes*.

The computational basis is not the only possible basis. I discuss two more common bases when introducing the Pauli matrices in Section A.1.3.3.

A collection of n qubits can be represented as unit a vector in the vector space $\mathbb{C}^{2^n} \cong \left(\mathbb{C}^2\right)^{\otimes n}$— the n-fold tensor product of the vector space $\mathbb{C}^2$ ($\mathbb{C} \times \mathbb{C}$). A basis for such a vector space is the n-fold tensor product of single-qubit computational-basis states. So, for example, a basis for a two-qubit state would be

$$|0\rangle_1 \otimes |0\rangle_2,\ |0\rangle_1 \otimes |1\rangle_2,\ |1\rangle_1 \otimes |0\rangle_2, \text{ and } |1\rangle_1 \otimes |1\rangle_2, \tag{A.4}$$

where the subscripts identify qubits. It is common to drop the tensor product and allow the order of the vectors to identify qubits. Thus, the set of basis vectors in Eq. A.4 are often presented as

$$|00\rangle,\ |01\rangle,\ |10\rangle, \text{ and } |11\rangle. \tag{A.5}$$

In terms of the n-qubit computational basis, a general n-qubit state is described as a vector

$$|\psi\rangle = \sum_{z \in \{0,1\}^n} \alpha_z |z\rangle = \begin{bmatrix} \alpha_{00\cdots00} \\ \alpha_{00\cdots01} \\ \vdots \\ \alpha_{11\cdots11} \end{bmatrix}, \tag{A.6}$$

where for all the amplitudes $\alpha_z \in \mathbb{C}$ and the normalization condition is

$$\sum_{z \in \{0,1\}^n} |\alpha_z|^2 = 1. \tag{A.7}$$

Note, unless otherwise stated, the vectors and matrices presented in this work are written in terms of the computational basis using standard positional notation, where bit strings labeling the computational-basis states start at zero, the value of each successive string increases by one, and the significance of the bits increases from right-to-left. Also note, that Eq. A.6 could just have easily been written in terms of any basis that spans the space.

The dual of a qubit vector is represented in Dirac notation as a *bra*, $\langle\cdot|$. The bra of a state $|\psi\rangle$ is the conjugate transpose of $|\psi\rangle$. For example, the dual of the state $|\psi\rangle$ as defined in Eq. A.6 is the bra or row vector

$$\langle\psi| \equiv |\psi\rangle^\dagger = (|\psi\rangle^*)^T = \sum_{z\in\{0,1\}^n} \alpha_z^* \langle z| = [\alpha_{00\cdots00}^* \quad \alpha_{00\cdots01}^* \quad \cdots \quad \alpha_{11\cdots11}^*], \tag{A.8}$$

where $*$ is the complex conjugate, T is the transpose operator, and $\dagger$ is the conjugate transpose.

In Dirac notation, an inner product between states $|\psi\rangle$ and $|\phi\rangle$ is notated as $\langle\psi|\phi\rangle$, where, given two vectors $|\psi\rangle = \sum_j \psi_j |j\rangle$ and $|\phi\rangle = \sum_j \phi_j |j\rangle$ expressed in the same basis, the inner product is defined as

$$\langle\psi|\phi\rangle \equiv \langle\psi||\phi\rangle = [\psi_1^* \quad \psi_2^* \quad \cdots \quad \psi_n^*] \begin{bmatrix} \phi_1 \\ \phi_2 \\ \vdots \\ \phi_n \end{bmatrix} = \sum_j \psi_j^* \phi_j. \tag{A.9}$$

As we will see, the inner product is useful in defining things such as normalization and probability of measurement outcomes. Note that since $\langle\psi|$ is a bra and $|\phi\rangle$ is a ket, the inner product $\langle\lambda|\psi\rangle$ is also known as a *bra-ket*. This is an example of physicist notation-humor.

Like the inner product, an outer product, also known as a dyad, is simply defined as

$$|\psi\rangle\langle\phi| \equiv \begin{bmatrix} \psi_1 \\ \vdots \\ \psi_n \end{bmatrix} [\phi_1^* \quad \cdots \quad \phi_n^*] = \begin{bmatrix} \psi_1\phi_1^* & \psi_1\phi_1^* & \cdots & \psi_1\phi_n^* \\ \psi_2\phi_1^* & \psi_2\phi_2^* & \cdots & \psi_2\phi_n^* \\ \vdots & \vdots & \ddots & \vdots \\ \psi_n\phi_1^* & \psi_n\phi_2^* & \cdots & \psi_n\phi_n^* \end{bmatrix}, \tag{A.10}$$

where the vectors $|\psi\rangle$ and $|\phi\rangle$ are defined as they were in Eq. A.9.

Given a set of basis vectors $\{|j\rangle\}$ that span a Hilbert space $\mathscr{H} = \langle\{|j\rangle\}\rangle_{\mathbb{C}}$, where $\langle\cdot\rangle_{\mathbb{C}}$ is the span over the field $\mathbb{C}$, it is clear that the identity matrix for $\mathscr{H}$ is

$$I_{\mathscr{H}} = \sum_j |j\rangle\langle j|. \tag{A.11}$$

This equation is known as the *resolution of the identity*. We can see that by linearity $I_{\mathscr{H}}$ sends any vector in $\mathscr{H}$ to itself. Often, the subscript indicating the Hilbert space of an identity is dropped and is understood by context. Also, it is common to equate I with the scalar 1.

Using the inner product, the Euclidean norm, also known as the 2-norm, of a vector $|\psi\rangle$ is written as

$$\||\psi\rangle\|_2 \equiv \sqrt{\langle\psi|\psi\rangle}. \tag{A.12}$$

Note, other norms can be defined and are distinguished by subscripts such as 2 for the Euclidean norm. In this work, if a norm for a vector is not identified by a subscript, it can be assumed that the Euclidean norm is being used.

Likewise, norms of operators can be defined. If a norm of any operator A on some Hilbert space $\mathscr{H}$ is not specified in this work, then $\|A\|$ is taken to be the *operator norm* defined as

$$\|A\|_{\text{op}} = \sup\{\|A|\psi\rangle\| \mid \forall|\psi\rangle \in \mathscr{H}, \||\psi\rangle\| = 1\}. \tag{A.13}$$

The Euclidean norm is used to define a *unit vector*. Such a vector $|\psi\rangle$ has the property

$$\||\psi\rangle\| = 1. \tag{A.14}$$

Note that Eq. A.14 is equivalent to Eq. A.7.

Two vectors $|\psi\rangle$ and $|\phi\rangle$ are said to be *orthogonal* if and only if

$$\langle\psi|\phi\rangle = 0. \tag{A.15}$$

A set of distinct vectors are said to be *orthonormal* if and only if for any pair of vectors $|j\rangle$ and $|k\rangle$ in the set

$$\langle j|k\rangle = \delta_{ij}. \tag{A.16}$$

A.1.3 Dynamics

Now that I have presented a representation of pure states, I now discuss how states evolve through time. That is, how the amplitudes of states change. The evolution of states is described by operators that act on states. As states can be represented as complex vectors, linear operators can be represented as complex matrices. See [AL98; Aar05; Aar13] for arguments why quantum mechanics is linear.

A.1.3.1 Schrödinger's equation

The evolution of a closed quantum system is described by Schrödinger's equation

$$H(t)\,|\psi\rangle = i\hbar\frac{\mathrm{d}|\psi\rangle}{\mathrm{d}t}, \tag{A.17}$$

where H is an operator known as a Hamiltonian. A *Hamiltonian* is a Hermitian operator that describes the energy of the system. A *Hermitian operator* A is a matrix such that $A = A^\dagger$. A Hamiltonian is defined as

$$H(t) \equiv \sum_j E_j(t)\ |j\rangle\langle j|, \tag{A.18}$$

where $E_j(t) \in \mathbb{R}$ and the vectors $\{|j\rangle\}$ form an orthonormal basis, which are known as energy eigenvectors. Note, in this dissertation I will assume that basis vectors are time-independent; however, one may also consider basis vectors that are time-dependent. The (possibly time-dependent) eigenvalues $E_j(t)$ are referred to as energy eigenvalues. The expectation of the Hamiltonian $H(t)$ of a system is the classical energy of the system (see Section A.1.4 for how to calculate an expectation value).

Note that Eq. A.18 is a natural definition for a Hamiltonian operator since it assigns real numbers $E_j(t)$, representing amounts of energy, to eigenvectors. Given this definition, the Hamiltonian H must be Hermitian since

$$H(t)^\dagger = \sum_j E_j(t)^*\,|j\rangle\langle j|^\dagger = \sum_j E_j(t)\,|j\rangle\langle j| = H(t). \tag{A.19}$$

Note, by the spectral decomposition theorem (see Chapter 2 of [NC11]), all Hermitian operators can be described by Eq. A.18. Thus, all Hermitian matrices can be identified as Hamiltonians.

A.1.3.2 Unitary Evolution

While dynamics are physically implemented through Hamiltonians, it is often useful to think about dynamics in terms of the evolution operators that Hamiltonians induce. Such evolution operators that take a state $|\psi(t_0)\rangle$ to state $|\psi(t)\rangle$ are define as

$$|\psi(t)\rangle = U(t,t_0)|\psi(t_0)\rangle, \tag{A.20}$$

where $U(t,t_0)$ is a unitary operator. A *unitary operator* U is represented as a matrix such that $U^{-1} = U^\dagger$. Thus, $UU^\dagger = U^\dagger U = I$. Note, here I have suppressed the time dependence for the

unitaries. I will often drop the dependency when it is unnecessary or cumbersome.

The class of unitary operators is the class of square matrices that leaves the norm of kets unchanged since

$$\|U|\psi\rangle\| = \sqrt{\langle\psi|U^\dagger U|\psi\rangle} = \sqrt{\langle\psi|\psi\rangle} = \||\psi\rangle\|. \tag{A.21}$$

This implies that the association of probabilities to the square of amplitudes of basis vectors still holds after the evolution of a state by a unitary operator.

The differential formula that relates a Hamiltonian H to the unitary operator U is given as

$$H(t)\,U(t,t_0) = i\hbar\frac{\mathrm{d}U(t,t_0)}{\mathrm{d}t} \tag{A.22}$$

with the boundary condition that $U(t_0,t_0) = I$.

The solution to Eq. A.22 depends on the time-dependence and the commutativity of the Hamiltonian with itself. The general solution to Eq A.22 is

$$U(t,t_0) = \mathscr{T}\,\exp(-\frac{i}{\hbar}\int_{t_0}^{t} d\tau\, H(\tau)) \tag{A.23}$$

$$= 1 + \sum_{n=1}^{\infty}\left(\frac{-i}{\hbar}\right)^n \int_{t_0}^{t} d\tau_n \int_{t_0}^{\tau_n} d\tau_{n-1}\cdots\int_{t_0}^{\tau_2} d\tau_1 H(\tau_i)H(\tau_2)\cdots H(\tau_n), \tag{A.24}$$

where $\mathscr{T}$ is the time-ordering operator, which is defined as

$$\mathscr{T}A(t_2)B(t_1) = \begin{cases} A(t_2)B(t_1) & \text{if } t_2 > t_1 \\ B(t_1)A(t_2) & \text{otherwise.} \end{cases} \tag{A.25}$$

Two operators A and B are said to commute if and only if the commutator

$$[A,B] \equiv AB - BA \tag{A.26}$$

is equal to zero.

If for all times t_1 and t_2, $[H(t_1), H(t_2)] = 0$, then effectively the time-ordering operator (Eq A.25) acts like identity. Thus in such a case, Eq. A.23 simplifies to

$$U(t,t_0) = \exp\{-\frac{i}{\hbar}\int_{t_0}^{t} d\tau\, H(\tau)\}. \tag{A.27}$$

It is clear that if the Hamiltonian is time-independent, then Eq. A.27 further simplifies to

$$U(t,t_0) = \exp\{-\frac{i}{\hbar}H\,[t-t_0]\}. \tag{A.28}$$

Note, the interval between two times is often notated as $\Delta t = t - t_0$.

One can easily prove that $U(t,t_0)$, as defined by equation A.23, A.27, or A.28, must be unitary if H is Hermitian (which it is by definition).

Note, so far, I have explicitly included Planck's constant $\hbar \approx 1.0546$ x 10^{-34} Joule seconds; however, in QC it is common to choose units such that $\hbar = 1$. In this work, I often follow this convention.

In the following sections, I discuss unitaries that are commonly used in the study of quantum information.

A.1.3.3 Single-qubit Unitaries

A particularly useful set of single-qubit unitaries are the Pauli matrices, which are

$$\sigma_1 = X \equiv \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}, \ \sigma_2 = Y \equiv \begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix}, \text{ and } \sigma_3 = Z \equiv \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}. \tag{A.29}$$

Although not technically a member of the Pauli matrices, the single-qubit identity

$$\sigma_0 = I \equiv \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \tag{A.30}$$

is often informally considered a Pauli matrix. In this work, I also include I as a member of the Pauli matrices.

As these matrices are ubiquitous in quantum information text, it is useful to highlight some of their properties. The Pauli matrices are Hermitian matrices that square to identity. Excluding identity, the Pauli matrices anticommute with each other. That is,

$$\sigma_i \sigma_j = -\sigma_j \sigma_i. \tag{A.31}$$

In terms of the anticommutator

$$\{\sigma_i, \sigma_j\} \equiv \sigma_i \sigma_j + \sigma_j \sigma_i, \tag{A.32}$$

this can be alternatively stated that for all i and j, $\{\sigma_i, \sigma_j\}$ is equal to zero when $i \neq j$ and $i, j \neq 0$.

The Pauli matrices may serve as observables in measurements. The eigenvectors associated with the non-identity Pauli measurements are commonly used basis-vector sets.

The Pauli Z operator is defined as

$$Z \equiv |0\rangle\langle 0| - |1\rangle\langle 1|. \tag{A.33}$$

Thus, the associated basis vectors of Z are the eigenvectors $|0\rangle$ and $|1\rangle$ with eigenvalues $+1$ and -1, respectively. Therefore, the computational basis is also known as the *Z-basis*. As the unitary Z adds a -1 phase to a $|1\rangle$ state and leave a $|0\rangle$ alone, Z is also known as the *phase-flip operator*.

The Pauli X operator is

$$X \equiv |+\rangle\langle +| - |-\rangle\langle -|, \tag{A.34}$$

where the eigenvectors

$$|+\rangle \equiv \frac{1}{\sqrt{2}}\left(|0\rangle + |1\rangle\right) \text{ and } |-\rangle \equiv \frac{1}{\sqrt{2}}\left(|0\rangle - |1\rangle\right) \tag{A.35}$$

with eigenvalues $+1$ and -1, respectively. This basis is therefore known as the *X-basis*. The operator X can also be written as

$$X = |0\rangle\langle 1| + |1\rangle\langle 0| \tag{A.36}$$

Therefore, X is known as the *bit-flip operator*.

The Pauli Y operator is given as

$$Y \equiv |+i\rangle\langle+i| - |-i\rangle\langle-i|, \tag{A.37}$$

where the eigenvectors

$$|+i\rangle \equiv \frac{1}{\sqrt{2}}(|0\rangle + i|1\rangle) \text{ and } |-i\rangle \equiv \frac{1}{\sqrt{2}}(|0\rangle - i|1\rangle) \tag{A.38}$$

with eigenvalues $+1$ and -1, respectively. Since $Y = iXZ$, the operator Y performs both a bit-flip and phase-flip.

Another common operator seen in quantum information is the *Hadamard gate*, which is defined as

$$H \equiv |0\rangle\langle+| + |1\rangle\langle-| = |+\rangle\langle 0| + |-\rangle\langle 1| = \frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}. \tag{A.39}$$

Thus, the Hadamard exchanges the basis states $|0\rangle$ and $|+\rangle$ as well as the basis states $|1\rangle$ and $|-\rangle$.

Two other common unitaries used in quantum information are roots of Z. The first is known as the *phase gate*, S, which is defined as

$$S \equiv \sqrt{Z} = \begin{bmatrix} 1 & 0 \\ 0 & i \end{bmatrix} = |0\rangle\langle 0| + i|1\rangle\langle 1|. \tag{A.40}$$

The other commonly used root of Z is the T operator,

$$T \equiv \sqrt[4]{Z} = \begin{bmatrix} 1 & 0 \\ 0 & e^{i\pi/4} \end{bmatrix} = |0\rangle\langle 0| + e^{i\pi/4}|1\rangle\langle 1|. \tag{A.41}$$

A.1.3.4 Two-qubit Unitaries

One of the simplest two-qubit unitaries is the SWAP operator. The SWAP operator simply exchanges the state of two qubits. That is, in the computational basis SWAP maps

$$|00\rangle \to |00\rangle, |01\rangle \to |10\rangle, |10\rangle \to |01\rangle, \text{ and } |11\rangle \to |11\rangle. \tag{A.42}$$

The SWAP gate is equivalent to the matrix

$$\text{SWAP} \equiv \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \tag{A.43}$$

with respect to the computational basis.

Other common two-qubit operations are controlled unitaries. For each of these operations, one qubit is called the "target" qubit and the other, the "control" qubit. These controlled unitaries have the property that if the control qubit is in the $|0\rangle$ state, then identity is applied to the target qubit; however, if the control qubit is in the $|1\rangle$ state, a single-qubit unitary is applied. This can be expressed as

$$\text{CU} \equiv |0\rangle\langle 0| \otimes I + |1\rangle\langle 1| \otimes U, \tag{A.44}$$

where the tensor product has been suppressed.

One of the most commonly discussed two-qubit operations is the CNOT or controlled-X (CX). This unitary is equivalent to

$$\text{CNOT} \equiv |0\rangle\langle 0| \otimes I + |1\rangle\langle 1| \otimes X = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}, \tag{A.45}$$

in the computational basis.

A related controlled unitary is the controlled-Z (CZ). This unitary is described as

$$\text{CZ} \equiv |0\rangle\langle 0| \otimes I + |1\rangle\langle 1| \otimes Z = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & -1 \end{bmatrix}, \tag{A.46}$$

in the computational basis. It is easy to prove that

$$CZ = |0\rangle\langle 0| \otimes I + |1\rangle\langle 1| \otimes Z = I \otimes |0\rangle\langle 0| + Z \otimes |1\rangle\langle 1|. \tag{A.47}$$

That is, either qubit can be considered the control qubit.

A.1.3.5 Three-qubit Unitaries

In quantum information, perhaps the most common three-qubit unitary discussed is the Toffoli gate. This gate is similar to the CNOT except that the Toffoli gate has two controls instead of one; thus, the Toffoli gate is also known as the CCNOT gate. The Toffoli gate can therefore be represented as

$$\text{Toffoli} \equiv (II - |11\rangle\langle 11|) \otimes I + |11\rangle\langle 11| \otimes X = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}, \tag{A.48}$$

in the computational basis.

The CZ analog of the Toffoli gate is the CCZ gate, that is, the doubly-controlled Z gate. The CCZ gate can be written as

$$\text{CCZ} \equiv (II - |11\rangle\langle 11|) \otimes I + |11\rangle\langle 11| \otimes Z = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 \end{bmatrix}, \tag{A.49}$$

A.1.4 Projective Measurements

Normalization allows the complex coefficients of a ket to be associated with probabilities. For example, if the state $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$ is projectively measured in the computational basis, then after the measurement the qubit is found in either the state $|0\rangle$ with probability $|\alpha|^2$ or the state $|1\rangle$ with probability $|\alpha|^2$. Likewise, if the state $|\psi\rangle = \sum_{z=\{0,1\}^n} \alpha_z |z\rangle$ is measured in the computational basis, then the probability of finding the state as $|z\rangle$ after measurement is $|\alpha_z|^2$. The identification of the square of amplitudes to probabilities is known as the *Born rule* [Bor26].

In general, a *projective measurement*, also known as a *von Neumann measurement*, is associated with an *observable* M, which is written as

$$M = \sum_{\lambda} \lambda P_{\lambda}, \tag{A.50}$$

where P_λ is a projector onto an eigenspace of M with eigenvalues λ. The projector P_λ is a sum

$$P_\lambda = \sum_j |j\rangle\langle j|, \tag{A.51}$$

where the vectors $\{|j\rangle\}$ are a set of basis vectors that span the λ-eigenspace.

Note that a projector P_λ has the property

$$P_\lambda^2 = P_\lambda \tag{A.52}$$

since

$$P_\lambda^2 = \sum_j |j\rangle\langle j| \sum_k |k\rangle\langle k| = \sum_j \sum_k |j\rangle\langle j||k\rangle\langle k| = \sum_j \sum_k \delta_{j,k} |j\rangle\langle k| = \sum_j |j\rangle\langle j| = P_\lambda, \tag{A.53}$$

where the sums run over the same basis.

Also, since the projectors are a sum of basis vectors that span the λ-eigenspaces, the projectors have the property that

$$\sum_\lambda P_\lambda = I. \tag{A.54}$$

The probability that the state $|\psi\rangle$ is projected into the λ-eigenspace is

$$p(\lambda) = \langle\psi|P_\lambda|\psi\rangle. \tag{A.55}$$

Note that by identifying probabilities with Eq. A.55 and assuming the state $|\psi\rangle$ is normalized, $0 \leq p(\lambda) \leq 1$, as we would want for a quantity representing probabilities.

We see Eq. A.51 and Eq. A.55 guarantees that $p(\lambda) \geq 0$ since

$$p(\lambda) = \langle\psi|P_\lambda|\psi\rangle = \sum_j \langle\psi|j\rangle\langle j|\psi\rangle = \sum_j |\langle j|\psi\rangle|^2 \geq 0. \tag{A.56}$$

Further, normalization of the pure states and Eq. A.54 mean that $p(\lambda) \leq 1$ since

$$\sum_\lambda p(\lambda) = \sum_\lambda \langle\psi|P_\lambda|\psi\rangle = \langle\psi|\sum_\lambda P_\lambda|\psi\rangle = \langle\psi|\psi\rangle = 1. \tag{A.57}$$

Upon measuring, a measurement device outputs a result corresponding to the eigenvalue of the eigenspace that the state was projected to, and the state $|\psi\rangle$ becomes

$$|\psi'\rangle = \frac{P_\lambda|\psi\rangle}{\sqrt{p(\lambda)}}. \tag{A.58}$$

Note that if two states $|\psi\rangle$ and $|\phi\rangle$ differ by a *global phase*, *i.e.*, $|\psi\rangle = e^{i\theta}|\phi\rangle$ where $\theta \in \mathbb{R}$, then the measurement statistics of the two states are the same. For this reason, global phases are said to be physically meaningless.

The average eigenvalue found when measuring an observable M is expected to be

$$\mathbb{E}[M] \equiv \langle\psi|M|\psi\rangle = \sum_\lambda \lambda \, \langle\psi|P_\lambda|\psi\rangle = \sum_\lambda \lambda \; p(\lambda), \tag{A.59}$$

where $|\psi\rangle$ is the state being measured. This value is known as an *expectation value*.

So far, I have only discussed projective measurements. These type of measurements take pure states to pure states. However, a more general type of quantum measurement is known as the *Positive Operator-Valued Measure* (POVM), which I discuss later in Section A.3.3.

A.2 Quantum Circuits

A *quantum circuit* is a space-time diagram depicting the sequence of quantum gates. In this text, I use the term "gate" to refer to any quantum operation including unitaries, measurements, and state preparations. An example of a quantum circuit is shown in the diagram:

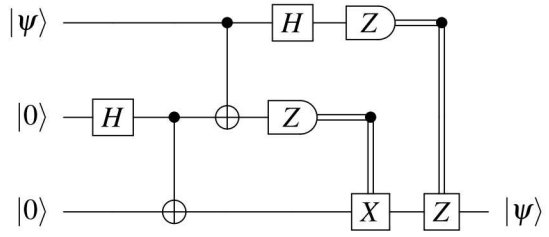


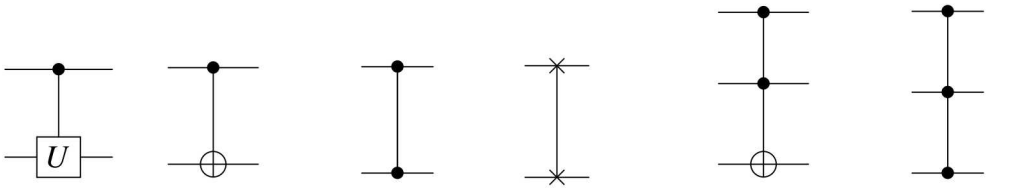
Figure A.1: An example of a teleportation quantum circuit depicting input states, unitaries, and measurements.

In these circuits, time moves from left-to-right and each wire corresponds to the world-line of a qubit. Input states generally appear on the far left, while output states appear on the far right.

A measurement is usually depicted as a stylistic drawing of a meter or as a cap-shape as seen in Fig. A.1. A label may be included in the drawing to indicate what basis the measurement is

being made in. If no basis is indicated, then it often assumed that the measurement is in the Z basis. Double lines drawn exiting the right-side of a measurement are used to represent classical information determined from measuring. On such lines, a filled in circle is used to indicate classical control of unitaries as seen in Fig A.1.

Unitaries are represented as boxes with symbols indicating the unitary that is being applied. Examples of the Pauli X, Pauli Z, and Hadamard gates are depicted in quantum circuit seen in Fig. A.1. The lines representing the qubits being affected by a unitary will enter from the left of the unitary and exit from the right. Some multi-qubit gates are indicated by unique symbols rather than boxes. Examples such symbols are seen in Fig. A.2.



(a) CU (b) CNOT (c) CZ (d) SWAP (e) Toffoli (f) CCZ

Figure A.2: Quantum-circuit diagrams for commonly used multi-qubit unitaries. In these figures the filled-in circles represent controls, and the hollow circles represent targets for which Pauli X may be applied to the qubit. Note that the CZ symbol shown in (c) is depicted with filled-in circles on both qubits. This is to indicate that either qubit can be treated as the control in the manner described by Eq. A.47.

A.3 Noisy Quantum Systems

So far, I have discussed the quantum mechanical formalism for pure states, which are ideal states that we assume we have perfect knowledge of. I now briefly describe the density-matrix formalism, which allows one to describe quantum systems where there is some lack of knowledge of the state of the system as well as its evolutions and measurements. Ideally, we would avoid using such a formalism, but in practice our knowledge and control of any system is never perfect.

A.3.1 Density Matrix

We begin by considering an ensemble of pure states

$$\mathscr{E} \equiv \{p(i), |\psi_i\rangle\}, \tag{A.60}$$

where i indexes the pure states $|\psi_i\rangle$. For such an ensemble, it is not necessary that states $|\psi_i\rangle$ are

orthogonal to each other; however, since the p_i represent probabilities, there is the requirement that

$$\sum_i p(i) = 1. \tag{A.61}$$

A *density matrix*, also known as a *density operator*, is used to describe the state of such an ensemble of pure states. A density operator is defined as

$$\rho \equiv \sum_i p(i) \, |\psi_i\rangle\langle\psi_i| \, . \tag{A.62}$$

The probabilities $p(i)$ of the density matrix also have the requirement of Eq. A.61 and the states $|\psi_i\rangle$ are normalized.

The density matrix is an alternative representation of a state. Any pure state $|\psi\rangle$ can be represented as a density operator

$$\rho = |\psi\rangle\langle\psi| \, . \tag{A.63}$$

For the density matrix representation, the trace is often useful when defining such things as normalization, measurements, and expectation values. Using the Dirac notation, the trace is written as

$$\mathrm{tr}(A) = \sum_i \langle i|A|i\rangle = \sum_i A_{ii}, \tag{A.64}$$

where the vectors $|i\rangle$ form a complete, orthonormal basis appropriate for the space that the operator A belongs to.

A useful property of the trace to note is

$$\mathrm{tr}(AB) = \sum_i \sum_j A_{ij} B_{ji} = \sum_j \sum_i B_{ji} A_{ij} = \mathrm{tr}(BA). \tag{A.65}$$

It is straightforward to show that the trace is invariant for general cyclic permutations, *e.g.*,

$$\mathrm{tr}(ABC) = \mathrm{tr}(BCA) = \mathrm{tr}(CAB). \tag{A.66}$$

Using the trace, given that a density operator is an ensemble of normalized pure-states and that the probabilities of these states sum to one, it is natural to express the normalization of density operator as

$$\begin{aligned}\mathrm{tr}(\rho) &= \sum_i \langle i|\rho|i\rangle = \sum_i \sum_j p(j)\langle i||\psi_j\rangle\langle\psi_j||i\rangle = \sum_j p(j)\langle\psi_j|\left(\sum_i |i\rangle\langle i|\right)|\psi_j\rangle \\ &= \sum_j p(j)\left\langle\psi_j\middle|I\middle|\psi_j\right\rangle = \sum_j p(j) = 1. \end{aligned} \tag{A.67}$$

Later, I discuss how to use the trace to express measurements and expectation values of density matrices.

Other frequently cited properties of the density matrix, which one can show by the representation's definition, is that

$$\rho^\dagger = \sum_i p(i)^* |\psi_i\rangle\langle\psi_i|^\dagger = \sum_i p(i)|\psi_i\rangle\langle\psi_i| = \rho \tag{A.68}$$

and

$$\forall|\phi\rangle \in \mathscr{H}, \langle\phi|\rho|\phi\rangle \geq 0 \tag{A.69}$$

since for any state $|\phi\rangle$ we find

$$\langle\phi|\rho|\phi\rangle = \sum_i p(i)\langle\phi|\psi_i\rangle\langle\psi_i|\phi\rangle = \sum_i p(i)|\langle\psi_i|\phi\rangle|^2 \geq 0. \tag{A.70}$$

Now that density matrices have been defined, one might be curious about whether one can easily determine if a density matrix represents a pure state or not. It would be even better if there was a method for evaluating the degree of purity.

We can start to answer these questions by first considering the ensembles representing pure states and maximally-mixed states.

Any pure state is represented as a normalized state $|\psi\rangle$, and thus the ensemble

$$\mathscr{E}_{\text{pure}} = \{p = 1, |\psi\rangle\} \tag{A.71}$$

and the density matrix

$$\rho_{\text{pure}} = |\psi\rangle\langle\psi|. \tag{A.72}$$

A maximally-mixed state for a Hilbert space $\mathscr{H}$ spanned by d basis states $\{|j\rangle\}$ is represented by the ensemble

$$\mathscr{E}_{\text{max-mix}} = \{p(j) = \frac{1}{d}, |j\rangle\} \tag{A.73}$$

and the density matrix

$$\rho_{\text{max-mix}} = \frac{1}{d}\sum_j |j\rangle\langle j| = \frac{1}{d}I. \tag{A.74}$$

That is, the maximally-mixed state has an equal probability of being in any of the basis states $\{|j\rangle\}$.

The purity of a density matrix ρ is defined as

$$P(\rho) = \text{tr}\left(\rho^2\right). \tag{A.75}$$

If we evaluate the purity of ρ_{pure} and $\rho_{\text{max-mix}}$ we find

$$P(\rho_{\text{pure}}) = 1 \tag{A.76}$$

and

$$P(\rho_{\text{max-mix}}) = \text{tr}\left(\frac{I}{d^2}\right) = \frac{1}{d}. \tag{A.77}$$

For a general density matrix ρ (Eq. A.62) we have

$$\begin{aligned} P(\rho) &= \sum_k \langle k| \left[\sum_i p(i) |\psi_i\rangle\langle\psi_i| \sum_j p(j) |\psi_j\rangle\langle\psi_j| \right] |k\rangle \\ &= \sum_{i,j} p(i)p(j) \left|\langle\psi_i|\psi_j\rangle\right| = \sum_i p(i)^2. \end{aligned} \tag{A.78}$$

Thus, we see that $\frac{1}{d} \leq P(\rho) \leq 1$.

A.3.2 Dynamics

Evolution of an ensemble of pure states as described by Eq. A.60 evolves by a unitary U according to

$$\mathscr{E} = \{p(i), |\psi_i\rangle\} \rightarrow \mathscr{E}' = \{p(i), U|\psi_i\rangle\}. \tag{A.79}$$

Thus, the evolution of a density operator from ρ to ρ' by a unitary U is described by

$$\rho' = U\rho U^\dagger. \tag{A.80}$$

Note that unitaries also preserve the normalization of a density matrix since

$$\mathrm{tr}\left(U\rho U^\dagger\right) = \mathrm{tr}\left(\rho U^\dagger U\right) = \mathrm{tr}(\rho). \tag{A.81}$$

A.3.3 Measurements

The previously described projective measurements are ideal measurements that take pure states to pure states. In this section, I now discuss a more general measurement called a Positive Operator-Valued Measure (POVM). A POVM may take pure states to mixed states.

A POVM is described by a set of measurement operators $\{M_i\}$. These operators must be such that

$$\sum_i M_i^\dagger M_i = I. \tag{A.82}$$

Given such a set, the probability of measuring a density operator ρ and obtaining the result i after the measurement is

$$p(i) = \mathrm{tr}\left(M_i \rho M_i^\dagger\right). \tag{A.83}$$

Note that just as the probability $p(\lambda)$ as defined for projective measurements (Eq. A.55) could be shown to be restricted to lie in $[0,1]$, it is simple to show that the same holds for $p(i)$ of Eq. A.83 given the normalization constrain of the density matrix Eq.A.57 and measurement operator constraint Eq. A.82.

Immediately after the measurement, the density operator becomes

$$\rho' = \frac{M_i\, \rho\, M_i^\dagger}{p(i)}. \tag{A.84}$$

The expectation of measuring an operator A of a density matrix ρ is defined as

$$\begin{aligned}\mathbb{E}[A] \equiv \mathrm{tr}(\rho A) &= \sum_i \langle i|\rho A|i\rangle = \sum_i \sum_j p(j)\langle i||\psi_j\rangle\langle\psi_j|A|i\rangle \\ &= \sum_j p(j)\langle\psi_j|A\left(\sum_i |i\rangle\langle i|\right)|\psi_j\rangle = \sum_j p(j)\left\langle\psi_j\middle|A\middle|\psi_j\right\rangle. \end{aligned} \tag{A.85}$$

A.3.4 Quantum Channels

More general evolutions of density matrices are described by a *quantum channel*

$$\rho' = \mathscr{E}(\rho) = \sum_i E_i \rho E_i^\dagger, \tag{A.86}$$

where the set of operators $\{E_i\}$, called *Kraus operators*, are such that

$$\sum_i E_i^\dagger E_i = I. \tag{A.87}$$

Such a transformation turns out to be the most general map that is consistent with quantum mechanics that takes a valid density matrix to another valid density matrix (see Section 4.6 of [Wil13]).

This transformation can be interpreted replacing the state ρ with

$$\frac{E_i \rho E_i^\dagger}{\mathrm{tr}\left(E_i \rho E_i^\dagger\right)} \tag{A.88}$$

with probability

$$p(i) = \mathrm{tr}\left(E_i \rho E_i^\dagger\right). \tag{A.89}$$

A.3.4.1 Error Channels

Quantum channels are commonly used to describe quantum noise. In this section, I will outline some commonly considered quantum error-channels.

Digital computers sometimes experience bit flip errors ($0 \leftrightarrow 1$). The analogous quantum error-channel is also known as the *bit-flip channel*. On a single qubit, this channel acts as

$$\rho = (1-p)\rho + pX\rho X, \tag{A.90}$$

since $X|0\rangle = |1\rangle$ and $X|1\rangle = |0\rangle$. We see that this channel applies X on the qubit with probability p and otherwise leaves the qubit unchanged.

Unlike digital computers, qubits can also experience phase flips, which occur when Z is applied as an error with probability p, as well as both phase and bit flips, which occurs when the error is Y instead.

A symmetric application of these three errors (X, Y, and Z) is the often studied channel known as the symmetric depolarizing-channel, which acts on single qubits as

$$\rho = (1-p)\rho + \frac{p}{3}(X\rho X + Y\rho Y + Z\rho Z). \tag{A.91}$$

This channel can be interpreted as a process that leaves the system untouched with probability $1-p$ and applies either X, Y, or Z with an equal probability of $p/3$. Note that on two qubits, errors are chosen equally from the set $\{I,X,Y,Z\}^{\otimes 2} \setminus I \otimes I$.

Finally, the amplitude-damping channel corresponds to qubits spontaneously decaying to the ground-state $|0\rangle$ with probability p. The Kraus operators for this channel is

$$E_0 = \sqrt{p}\,|0\rangle\langle 1| \text{ and } E_1 = |0\rangle\langle 0| + \sqrt{1-p}\,|1\rangle\langle 1|. \tag{A.92}$$

Later, in Chapter **??**, I discuss a class of quantum error-correcting codes known as stabilizer codes. For those who are curious about the broader field of quantum error correction, see the textbook [LB13] as well as Preskill's lecture notes [Pre98]. For a book on how classical errors affect the architecture of modern CMOS processors, see [Muk08].

A.4 A Brief Comparison between Classical and Quantum Computation

I started this chapter by first considering the states of digital computers. I now end this chapter by making some comparisons between classical and quantum computation.

In classical (binary) computation, it is known that any Boolean function on n bits can be written as a circuit composed of only NAND gates ($00 \to 1$, $01 \to 1$, $10 \to 1$, and $11 \to 0$) and fanout operations [She13] (assuming bit preparation and readout). That is, $\{\text{NAND}, \text{fanout}\}$ is a universal set of gates for finite-state machines (digital computers) since the composition of gates from this set will take any state $z \in \{0,1\}^n$ to any other state $z' \in \{0,1\}^n$. Note that a NAND gate can be formed from two transistors (see, for example, Chapter 2 of Feynman's lecture notes on computation [Fey98]).

Given that $\{\text{NAND}, \text{fanout}\}$ is universal for classical computation, one might wonder whether there are simple sets of gates that can be composed to form any quantum circuit. Fortunately, universal sets of quantum gates that allow any unitary in $U(2^n)$ to be efficiently approximated do exist. The "standard set" of universal gates is $\{H, S, \text{CNOT}, T\}$ [NC11] (assuming state preparations and measurements in the computational basis). Other universal gate sets (making the same assumptions) include:

1. The three-qubit Deutsch gate [Deu89]
2. All two-qubit gates [DiV95]
3. CNOT and all single-qubit $U(2)$ [Bar+95]
4. H, T, and CNOT [Boy+99]
5. Controlled single-qubit rotation [RG02]
6. Toffoli and H [Shi03; Aha03]

For proofs and discussions on universal quantum-gate sets, see Chapter 4 of [NC11] and Chapter 6 of [Pre98].

The fanout gate is often implicitly assumed and is rather natural for digital circuits. It is achieved by splitting a wire to produce multiple bits from one. In QC, the analogous operation is copying a general quantum state. The *no-cloning theorem* [WZ82; Die82; Joz02] states that no unitary will allow such copying perfectly. That is, for any general state $|\psi\rangle$ and a fiducial state $|\phi\rangle$ (*e.g.*, $|0\rangle$), there is no unitary that takes $|\psi\rangle \otimes |\phi\rangle \to |\psi\rangle \otimes |\psi\rangle$. If cloning were possible, it can be shown that it would allow for superluminal communication [Die82]. For discussions about the no-cloning theorem see Box 12.1 of [NC11], Chapter 3 of [Wil13], and Chapter 4 of [Pre98].

Another important difference between classical and quantum circuits, which we have already implicitly discussed, is how measurements affect the state of a computer. In digital computers one can measure any bit as many times as one likes and not affect the state of the computer. However, as we have discussed in Section A.1.4, even for ideal pure states and ideal projective measurements a general measurement will collapse a state onto a subspace. Thus, we must be careful with what measurements we make during the execution of quantum algorithm as measurements *can* result in one loosing information that was encoded in the quantum state. Together, measurement collapse combined with no-cloning means that, in the middle of a quantum computation, one can't just arbitrarily read-off and see what the state is.

However, if done wisely, it is sometimes advantageous to make measurements while running a quantum algorithm. For example, some quantum error-correction protocols enact judicially chosen measurements to digitize error and determine recovery operations (*e.g.*, see Chapter **??**). Further, there is a quantum computational model entirely based on measurements known as *measurement-based quantum computation* (MQC or MBQC) [RB01; Joz05; Bri+09].

Although measurements collapse quantum states, one may wonder whether it is possible to extract more information out of n qubits than n bits. Holevo's theorem [Hol73] proves that this is impossible. One may obtain at most 1 bit of information for every 1 qubit measured. Thus, the notion often heard in popular science reporting that quantum computation is really a form of massively-parallel computation is simplistic, at best.

The unique state-space and evolutions allowed by quantum mechanics result in quantum computers being able to manipulate probability distributions of states in ways seemingly unavailable

to classical computers. Even without access to large-scale reliable quantum computes, researchers have developed numerous quantum algorithms that outperform (sometimes exponentially so) certain best known classical algorithms. In Chapter 2, I will discuss a quantum algorithm that can serve as framework for finding approximate solutions to classical combinational satisfaction problems. For a comprehensive collection of quantum algorithms, see the site maintained by Jordan [Jor09]. For introductions to quantum algorithms see [Sho00; MM11; Mon15; Col+18].

Finally, unlike modern digital computers, which are predominately based on complementary metal-oxide-semiconductor (CMOS) technology, there is currently no dominent substrate for quantum computation. Technologies that are currently being explored include ion traps [BW08; Sin+10; Hum+18], superconductors [MSS00; Wen16; Hum+18], semiconductors [LD97; Zwa+12; GZ18; Hum+18], neutral atoms [Bre+98; Jak+99; Mal+15], nitrogen vacancies in diamonds [Dut+07; Yao+12; CH13], and optics [Kok+05; Bar+14]. Each of these quantum-computing substrates has its own advantages and disadvantages (see the works cited in the previous sentence).

The field of quantum computation is still rapidly evolving. In the following chapters, I present my contributions to the study of quantum algorithms and the development of large-scale, reliable quantum-computing architectures.

Appendix B

Additional Proofs and Code

B.1 Proofs for Section 3.2

Proof of Lemma 3.1. Observe that for standard basis vectors $|i\rangle, |j\rangle, |k\rangle, |l\rangle \in \mathbb{C}^2$, we have

$$\begin{aligned}
\langle ij|X \otimes X|kl\rangle &= (i \oplus k)(j \oplus l), && \text{(B.1)}\\
\langle ij|Y \otimes Y|kl\rangle &= (-1)^{\delta_{kl}}(i \oplus k)(j \oplus l), && \text{(B.2)}\\
\langle ij|Z \otimes Z|kl\rangle &= (-1)^{k \oplus l}\delta_{ik}\delta_{jl}, && \text{(B.3)}
\end{aligned}$$

were δ_{ij} is the usual Kronecker delta. Denoting an arbitrary product state as $|\psi\rangle = ac|00\rangle + ad|01\rangle + bc|10\rangle + bd|11\rangle$ for $|a|^2 + |b|^2 = |c|^2 + |d|^2 = 1$, we have

$$\begin{aligned}
\langle \psi|H|\psi\rangle &= \alpha(a^*c^*bd + acb^*d^* + a^*d^*bc + adb^*c^*) + \\
&\quad \beta(-a^*c^*bd - acb^*d^* + a^*d^*bc + adb^*c^*) + \\
&\quad \gamma(|a|^2|c|^2 + |b|^2|d|^2 - |a|^2|d|^2 - |b|^2|c|^2) && \text{(B.4)}\\
&= 2\,\mathrm{Re}[acb^*d^*](\alpha - \beta) + 2\,\mathrm{Re}[adb^*c^*](\alpha + \beta) + \\
&\quad \gamma(|a|^2|c|^2 + |b|^2|d|^2 - |a|^2|d|^2 - |b|^2|c|^2) \\
&\leq 2|a||b||c||d|(|\alpha + \beta| + |\alpha - \beta|) + \\
&\quad |\gamma|\Big|(|a|^2 - |b|^2)(|c|^2 - |d|^2)\Big|.
\end{aligned}$$

where the last inequality follows from the triangle inequality. Let us simplify the notation above by assuming without loss of generality $a, b, c, d \in \mathbb{R}^+$. We may also assume without loss of generality that $a \geq b$ and $c \geq d$ (since this maximizes the upper bound). Thus:

$$\langle \psi|H|\psi\rangle \leq 2abcd(|\alpha + \beta| + |\alpha - \beta|) + |\gamma|(a^2 - b^2)(c^2 - d^2).$$

Note now for any $\alpha, \beta \in \mathbb{R}$, $|\alpha + \beta| + |\alpha - \beta| = ||\alpha| + |\beta|| + ||\alpha| - |\beta||$. Assume first $|\alpha| \geq |\beta|$. Then

$$\langle \psi|H|\psi\rangle \leq 4abcd|\alpha| + |\gamma|(a^2 - b^2)(c^2 - d^2). \quad \text{(B.5)}$$

Let $p = 4abcd$ and $q = (a^2-b^2)(c^2-d^2)$. Note $p,q \geq 0$. Also, we claim $p+q \leq 1$; this will imply $\langle\psi|H|\psi\rangle \leq \max(|\alpha|,|\gamma|)$. To see this claim, note

$$p+q = (ac+bd)^2 - (ad-bc)^2 \leq (ac+bd)^2 \leq 1,$$

where the last inequality follows from the Cauchy-Scharz inequality. The case of $|\beta| \geq |\alpha|$ follows analogously with $|\alpha|$ in Equation (B.5) replaced with $|\beta|$. We hence have $\langle\psi|H|\psi\rangle \leq \max(|\alpha|,|\beta|,|\gamma|) = \|(|\alpha|,|\beta|,|\gamma|)\|_\infty$.

We now show matching lower bounds, i.e. that $|\alpha|$, $|\beta|$, and $|\gamma|$ are attainable. Returning to Equation (B.4):

- For $|\alpha|$: If $\alpha \geq 0$, set $a = b = c = d = 1/\sqrt{2}$, and if $\alpha < 0$, set $a = b = c = 1/\sqrt{2}$ and $d = -1/\sqrt{2}$.
- For $|\beta|$: If $\beta \geq 0$, set $a = i/\sqrt{2}$, $c = i/\sqrt{2}$, $b = d = 1\sqrt{2}$, and if $\beta < 0$, set $a = -i/\sqrt{2}$, $c = i/\sqrt{2}$, $b = d = 1\sqrt{2}$.
- For $|\gamma|$: If $\gamma \geq 0$, set $a = c = 1$ and $b = d = 0$. and of $\gamma < 0$, set $a = d = 1$, $b = c = 0$.

□

Proof of Lemma 3.2. Denoting an arbitrary two-qubit state as $|\psi\rangle = a|00\rangle + b|01\rangle + c|10\rangle + d|11\rangle$ for $|a|^2 + |b|^2 = |c|^2 + |d|^2 = 1$, we have via Equations (B.1)-(B.3) that

$$\begin{aligned}
\langle\psi|X\otimes X|\psi\rangle &= a^*d + ad^* + b^*c + bc^*, \\
\langle\psi|Y\otimes Y|\psi\rangle &= -a^*d - ad^* + b^*c + bc^*, \\
\langle\psi|Z\otimes Z|\psi\rangle &= |a|^2 - |b|^2 - |c|^2 + |d|^2.
\end{aligned}$$

Thus,

$$\begin{aligned}
\langle\psi|H|\psi\rangle &= \alpha(2\mathrm{Re}(ad^*) + 2\mathrm{Re}(bc^*)) + \beta(-2\mathrm{Re}(ad^*) + 2\mathrm{Re}(bc^*)) + \gamma(|a|^2 + |d|^2 - |b|^2 - |c|^2) \\
&= 2\mathrm{Re}[ad^*](\alpha-\beta) + 2\mathrm{Re}[bc^*](\alpha+\beta) + (|a|^2 + |d|^2 - |b|^2 - |c|^2)\gamma.
\end{aligned}$$

Observe that since the coefficient of γ depends on only *absolute values* of a,b,c,d, we can assume without loss of generality that the optimal assignment has $a,b,c,d \geq 0$ and satisfies

$$\langle\psi|H|\psi\rangle = 2ad|\alpha-\beta| + 2bc|\alpha+\beta| + (a^2+d^2-b^2-c^2)\gamma.$$

By applying the Arithmetic-Geometric mean inequality, we hence have

$$\begin{aligned}
\langle\psi|H|\psi\rangle &\leq (a^2+d^2)|\alpha-\beta| + (b^2+c^2)|\alpha+\beta| + (a^2-b^2-c^2+d^2)\gamma \\
&= (a^2+d^2)(|\alpha-\beta|+\gamma) + (b^2+c^2)(|\alpha+\beta|-\gamma) \\
&\leq \max(|\alpha-\beta|+\gamma, |\alpha+\beta|-\gamma),
\end{aligned}$$

where the last statement follows since $a^2+b^2+c^2+d^2=1$. The matching lower bound is obtained as follows. To achieve $|\alpha-\beta|+\gamma$ when $\alpha \geq \beta$, set $a=d=1/\sqrt{2}$, and when $\alpha \leq \beta$, set $a=1/\sqrt{2}, d=-1/\sqrt{2}$. Similarly, to achieve $|\alpha+\beta|-\gamma$ when $\alpha \geq -\beta$, set $b=c=1/\sqrt{2}$, and when $\alpha \leq -\beta$, set $b=1/\sqrt{2}, c=-1/\sqrt{2}$.

□

B.2 Lemmas and Mathematica code

In Section 3.2.2 we use the following lemma, which is stated as given in [BOFV14]. Below, ${}_2F_1(a,b;c;z)$ is the hypergeometric function, defined for $|z|<1$ as

$$ {}_2F_1(a,b;c;z) = \sum_{n=0}^{\infty} \frac{(a)_n (b)_n}{(c)_n} \frac{z^n}{n!}, $$

where for $n \geq 0$, we have Pochhammer symbol $(x)_n = \Gamma(x+n)/\Gamma(x) = x(x+1)\cdots(x+n-1)$ for Γ the Gamma function.

Lemma B.1 (Briët, de Oliveira Filho and Vallentin [BOFV14]). *Let u,v be unit vectors in $\mathbb{R}^n$ and let $Z \in \mathbb{R}^{r\times n}$ be a random matrix whose entries are distributed independently according to the standard normal distribution with mean 0 and variance 1. Then,*

$$ \mathbb{E}\left[\frac{Zu}{\|Zu\|_2}\cdot\frac{Zv}{\|Zv\|_2}\right] = \frac{2}{r}\left(\frac{\Gamma((r+1)/2)}{\Gamma(r/2)}\right)^2 (u\cdot v)\, {}_2F_1\left(1/2,1/2;r/2+1;(u\cdot v)^2\right). \tag{B.6} $$

Mathematica code Below, we give the Mathematica code used to numerically calculate the approximation ratios of Theorem 3.2:

```
g[r_] := 2/r (Gamma[(r + 1)/2]/Gamma[r/2])^2
F[r_, t_] := g[r] t Hypergeometric2F1[1/2, 1/2, r/2 + 1, t^2]
ApproxRatio[r_] := Min[Select[ Table[(1 - F[r, t])/(1 - r t),
                                    {t, -1, 1/r, 0.01}], # > 0 &]]
ApproxRatio[1]
ApproxRatio[2]
ApproxRatio[3]
```

DISTRIBUTION:

1	MS 0359	D. Chavez, LDRD Office, 1911
1	MS 0899	Technical Library, 9536 (electronic copy)


Sandia National Laboratories