Statistical Analysis of Quantum Phase Estimation and Quantum Annealing

by

Hongzhi Liu

A dissertation submitted in partial fulfillment

of the requirements for the degree of

Doctor of Philosophy

(Statistics)

at the

UNIVERSITY OF WISCONSIN-MADISON

2024

Date of Final Oral Exam: 07/29/2024
The dissertation is approved by the following members of the Final Oral Committee:
Yazhen Wang, Professor, Statistics
Miaoyan Wang, Associate Professor, Statistics
Jiwei Zhao, Associate Professor, Biostatistics and Medical Informatics
Yinqiu He, Assistant Professor, Statistics
Yiqiao Zhong, Assistant Professor, Statistics

Acknowledgments

First and foremost, I would like to convey my deepest gratitude to my advisor, Professor Yazhen Wang, for his treasured support and insightful guidance throughout my doctoral studies. He has consistently offered me precious advice on both my research and my daily life. His patience and encouragement have made a profound impact on the completion of my program.

I am grateful to my dissertation committee members: Professor Jiwei Zhao, Professor Miaoyan Wang, Professor Yinqiu He, and Professor Yiqiao Zhong, for their enlightening feedback and constructive suggestions in the development of the thesis.

I would like to express my sincere thanks to the faculty and staff in the Department of Statistics at the University of Wisconsin-Madison for their extensive help. I am also deeply appreciative of my friends, whose unwavering support and motivating suggestions have been crucial to my PhD journey.

Contents

Abstract

Quantum computing is a burgeoning research area that involves developing quantum devices that exploit the unique properties of quantum mechanics and designing quantum algorithms that run on these devices to address tough problems that even the most powerful classical supercomputers cannot efficiently solve. Extensive theoretical work has shown that a range of quantum algorithms can provide substantial, even exponential, speedup over the best-known classical algorithms. Since the quantum mechanics behind quantum computing is essentially stochastic, quantum computation has inherent randomness. As a result, statistical techniques can be employed to design quantum algorithms and investigate their properties. This thesis first introduces several basic concepts of quantum computing and then delves into two categories of quantum algorithms: quantum phase estimation and quantum annealing.

Quantum phase estimation refers to estimating the eigenphase ϕ in the eigenvalue $e^{i\phi}$ corresponding to an eigenvector $|\phi\rangle$ of a given unitary matrix U. It serves as a fundamental subroutine in various quantum algorithms. This thesis introduces Kitaev's phase estimation algorithm and the Bayesian phase estimation algorithm, which are sampling-based quantum algorithms for estimating the unknown phase, and examines their statistical properties.

Quantum annealing is a meta-heuristic quantum algorithm designed to tackle global optimization problems, particularly combinatorial minimization problems whose solution space can be vast and filled with numerous local minima. This thesis first presents the simulated annealing algorithm, which serves as the classical counterpart to quantum an-

nealing, and then introduces the quantum annealing algorithm, discusses its asymptotic convergence property, and provides a lower bound on the probability of successfully solving the concerned optimization problem at the final annealing time.

Chapter 1

Introduction

1.1 Overview

Quantum computing is a burgeoning research area that involves developing quantum devices that exploit the unique properties of quantum mechanics and designing quantum algorithms that run on these devices to address tough problems that even the most powerful classical supercomputers cannot efficiently solve. Extensive theoretical work has shown that a range of quantum algorithms can provide substantial, even exponential, speedup over the best-known classical algorithms (Shor, 1994; Grover, 1996; Nielsen & Chuang, 2010; Wang & Liu, 2022). Since the quantum mechanics behind quantum computing is essentially stochastic, quantum computation has inherent randomness (Nielsen & Chuang, 2010; Wang & Liu, 2022; Griffiths & Schroeter, 2019). As a result, statistical techniques can be employed to design quantum algorithms and investigate their properties. This thesis first introduces several basic concepts of quantum computing in Chapter 1 and then delves into two categories of quantum algorithms: quantum phase estimation and quantum annealing.

Quantum phase estimation refers to estimating the eigenphase ϕ in the eigenvalue $e^{i\phi}$ corresponding to an eigenvector $|\phi\rangle$ of a given unitary matrix U. It serves as a fundamental subroutine in a variety of efficient quantum algorithms (Shor, 1994; Temme, Osborne, Vollbrecht, Poulin, & Verstraete, 2011; Wang, 2012; Ozols, Roetteler, & Roland, 2013). At

present, there are mainly two types of quantum algorithms for addressing this problem: one is based on quantum Fourier transform, and the other involves sampling from a quantum circuit and then postprocess the gathered data. This thesis concentrates on the second approach. In Chapter 2, we introduce Kitaev's phase estimation algorithm and discuss its computational complexity. In Chapter 3, we introduce the Bayesian phase estimation algorithm and examine its statistical properties (Kitaev, Shen, & Vyalyi, 2002; Svore, Hastings, & Freedman, 2013; Wiebe & Granade, 2016).

Quantum annealing is a meta-heuristic quantum algorithm designed to tackle global optimization problems, particularly combinatorial minimization problems whose solution space can be vast and filled with numerous local minima. In Chapter 4, we begin by presenting the simulated annealing algorithm, which serves as the classical counterpart to quantum annealing. Then, we introduce the quantum annealing algorithm, discuss its asymptotic convergence property, and provide a lower bound on the probability of successfully solving the concerned optimization problem at the final annealing time (Kadowaki & Nishimori, 1998; Hauke, Katzgraber, Lechner, Nishimori, & Oliver, 2020; Rajak, Suzuki, Dutta, & Chakrabarti, 2023; Wang & Liu, 2022; Wang, Wu, & Liu, 2023).

1.2 Basic Concepts of Quantum Computing

1.2.1 Qubit

In classical computing, information is represented by bits, whereas quantum computing utilizes qubits as the basic units of information (Kitaev et al., 2002). Unlike a classical bit whose state is restricted to being either 0 or 1, a qubit has infinitely many possible states. It can be in a *superposition* $|\phi\rangle$ of two orthonormal *basis states* $|0\rangle$ and $|1\rangle$ (denoted in *Dirac notation* and pronounced as "ket 0" and "ket 1", respectively) (Nielsen & Chuang, 2010), where the basis states can be represented in vector form as

$$|0\rangle = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \text{ and } |1\rangle = \begin{bmatrix} 0 \\ 1 \end{bmatrix},$$

and the general state $|\phi\rangle$ of the qubit can be written as

$$|\phi\rangle = \alpha|0\rangle + \beta|1\rangle = \alpha \begin{bmatrix} 1 \\ 0 \end{bmatrix} + \beta \begin{bmatrix} 0 \\ 1 \end{bmatrix} = \begin{bmatrix} \alpha \\ \beta \end{bmatrix}, \text{where } \alpha, \beta \in \mathcal{C}, \|\alpha\|^2 + \|\beta\|^2 = 1.$$

There are multiple ways to physically implement qubits. For example, IBM has developed a quantum chip with 127 superconducting quantum qubits, where related studies suggest that such quantum chips hold potential for solving real-world problems (Kim et al., 2023). For more details about physically implementing qubits, see Clarke and Wilhelm (2008), Qiang et al. (2018), and Lu (2021).

1.2.2 Quantum Entanglement and Quantum Register

A quantum computer contains multiple qubits, and these qubits could have correlations. Manipulating certain qubits within a group of qubits can affect the states of the remaining qubits, meaning that the overall state of the group of qubits cannot be described solely by the individual states of each qubit. This effect is known as *quantum entanglement* (Horodecki, Horodecki, Horodecki, & Horodecki, 2009). A set of entangled qubits form a *quantum register*. The state $|\phi\rangle$ of a two-qubit quantum register can be represented by a linear combination of 2^2 basis states (Nielsen & Chuang, 2010), i.e. $|\phi\rangle = \alpha_{00}|00\rangle + \alpha_{01}|01\rangle + \alpha_{10}|10\rangle + \alpha_{11}|11\rangle$, where $\alpha_{00}, \alpha_{01}, \alpha_{10}, \alpha_{11} \in \mathcal{C}, \|\alpha_{00}\|^2 + \|\alpha_{01}\|^2 + \|\alpha_{10}\|^2 + \|\alpha_{11}\|^2 = 1$, and

$$|00\rangle = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix}, |01\rangle = \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \end{bmatrix}, |10\rangle = \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \end{bmatrix}, |11\rangle = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}.$$

More generally, the state $|\phi\rangle$ of a quantum register with n qubits can be represented by a 2^n-dimensional unit-length complex vector (Nielsen & Chuang, 2010), i.e.

$$|\phi\rangle = \sum_{s \in \{0,1\}^n} \alpha_s |s\rangle, \alpha_s \in \mathcal{C}, \sum_{s \in \{0,1\}^n} \|\alpha_s\|^2 = 1,$$

where each s represents a n-bit string of binary values, $|s\rangle$ is the basis state associated with s, and α_s is the coefficient corresponding to $|s\rangle$.

1.2.3 Measurement

The information embedded in the quantum state of a quantum register can be extracted by measurement (Wang & Liu, 2022). The outcome of a measurement is a binary string associated with a basis state, and due to the inherent characteristics of quantum mechanics, such a measurement will cause the state of the quantum register to irreversibly collapse into this basis state (Griffiths & Schroeter, 2019). The probability of getting a specific basis state is determined by the coefficient associated with it (Nielsen & Chuang, 2010): when measuring a n-qubit quantum register with unknown quantum state $|\phi\rangle = \sum_{s\in\{0,1\}^n} \alpha_s|s\rangle$, the probability of obtaining a specific n-bit binary string s is given by

$$P(\text{obtain the specific binary string } s \in \{0,1\}^n) = \|\alpha_s\|^2. \tag{1.1}$$

After the measurement, the quantum register's original state $|\phi\rangle$ collapses into a basis state, indicating that repeated measurements will not yield a different outcome. Moreover, the no-cloning theorem in quantum mechanics says that one cannot clone an arbitrary unknown quantum state to get an independent and identical copy of it (Wootters & Zurek, 1982), which implies that acquiring the same quantum state $|\phi\rangle$ for further measurement may consume many computational resources.

One may choose to measure only a portion of the qubits within a quantum register instead of the whole register. Without loss of generality, suppose we want to measure the first m qubits of an n-qubit quantum register with state $|\phi\rangle = \sum_{s\in\{0,1\}^n} \alpha_s|s\rangle$, the probability of obtaining a specific outcome $s^* \in \{0,1\}^m$ is (Rieffel & Polak, 2000)

$$P(\text{obtain } s^* \in \{0,1\}^m) = \sum_{s\in\mathcal{S}_{\text{consistent}}} \|\alpha_s\|^2, \tag{1.2}$$

where $\mathcal{S}_{\text{consistent}} = \{s \in \{0,1\}^n : \text{the first } m \text{ components of } s = s^*\}$ is the set of basis

states whose first m components are consistent with s^*. For example, when measuring the first qubit of a two-qubit system with a state

$$|\phi\rangle = \frac{1}{\sqrt{30}}(|00\rangle + 2|01\rangle + 3|10\rangle + 4|11\rangle),$$

the probability of obtaining 1 is equal to $(\frac{3}{\sqrt{30}})^2 + (\frac{4}{\sqrt{30}})^2 = \frac{5}{6}$.

After measuring the first m qubits of the n-qubit quantum register, the state of the rest $n-m$ qubits becomes $|\psi\rangle = \sum_{s' \in \{0,1\}^{n-m}} \alpha_{s'} |s'\rangle$ (Nielsen & Chuang, 2010). Denote the outcome of the measurement by $s^* \in \{0,1\}^m$, we have

$$\alpha_{s'} = \frac{\alpha_{(s^*,s')}}{\sqrt{\sum_{s \in \mathcal{S}_{\text{consistent}}} \|\alpha_s\|^2}},$$

where $\alpha_{(s^*,s')}$ is the coefficient corresponding to the basis state whose first m qubits is s^* and last $n-m$ qubits is s', and $\mathcal{S}_{\text{consistent}} = \{s \in \{0,1\}^n : \text{the first } m \text{ components of } s = s^*\}$. For example, when measuring the first qubit of a two-qubit system with a state

$$|\phi\rangle = \frac{1}{\sqrt{30}}(|00\rangle + 2|01\rangle + 3|10\rangle + 4|11\rangle),$$

if the outcome of the measurement is 0, the state of the second qubit will become

$$(\frac{1}{\sqrt{30}} / \sqrt{\left|\frac{1}{\sqrt{30}}\right|^2 + \left|\frac{2}{\sqrt{30}}\right|^2})|0\rangle + (\frac{2}{\sqrt{30}} / \sqrt{\left|\frac{1}{\sqrt{30}}\right|^2 + \left|\frac{2}{\sqrt{30}}\right|^2})|1\rangle = \frac{1}{\sqrt{5}}|0\rangle + \frac{2}{\sqrt{5}}|1\rangle,$$

and if the outcome of the measurement is 1, the state of the second qubit will become

$$(\frac{3}{\sqrt{30}} / \sqrt{\left|\frac{3}{\sqrt{30}}\right|^2 + \left|\frac{4}{\sqrt{30}}\right|^2})|0\rangle + (\frac{4}{\sqrt{30}} / \sqrt{\left|\frac{3}{\sqrt{30}}\right|^2 + \left|\frac{4}{\sqrt{30}}\right|^2})|1\rangle = \frac{3}{5}|0\rangle + \frac{4}{5}|1\rangle.$$

1.2.4 Quantum Gate

The state of a quantum register can be operated by quantum gates. A quantum gate acting on an n-qubit quantum register can be mathematically represented by a $2^n \times 2^n$

unitary matrix, and any $2^n \times 2^n$ unitary matrix corresponds to a n-qubit gate (Nielsen & Chuang, 2010). Given a quantum register with state $|\phi\rangle$ and a quantum gate U acting on this register, the state after the gate operation, denoted by $U|\phi\rangle$, can be calculated by matrix multiplication. For example, if a one-qubit gate U acts on a qubit with state $|\phi\rangle$, where

$$|\phi\rangle = \begin{bmatrix} \alpha \\ \beta \end{bmatrix}, U = \begin{bmatrix} a & b \\ c & d \end{bmatrix},$$

the state after the operation can be computed by

$$U|\phi\rangle = \begin{bmatrix} a & b \\ c & d \end{bmatrix} \begin{bmatrix} \alpha \\ \beta \end{bmatrix} = \begin{bmatrix} a\alpha + b\beta \\ c\alpha + d\beta \end{bmatrix}.$$

Multiple quantum gates can simultaneously operate on distinct portions of a quantum register, and the overall effect of these gates can be represented by the tensor product of their matrix representations. Suppose the Hadamard gate H acts on the first qubit of a 2-qubit system and the Pauli-x gate σ^x acts in parallel on the second qubit, where

$$H = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}, \sigma^x = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix},$$

the overall effect of this operation is

$$H \otimes \sigma^x = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} & 1 \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \\ 1 \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} & -1 \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \end{bmatrix} = \frac{1}{\sqrt{2}} \begin{bmatrix} 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & -1 \\ 1 & 0 & -1 & 0 \end{bmatrix}.$$

In general, when a series of quantum gates $U_1, U_2, ..., U_m$ act simultaneously on different groups of qubits, the operation can be represented by tensor product $U_1 \otimes U_2 \otimes ... \otimes U_m$ (Nielsen & Chuang, 2010).

Any quantum gate can be decomposed as a series of one- or two-qubit quantum gates,

and these gates can be further approximated using several gates from a one- or two-qubit gate set (Möttönen & Vartiainen, 2006). A quantum gate set is called a *universal gate set* if any quantum gate U can be arbitrarily closely approximated by a finite series of gates from this set (DiVincenzo, 1995). One example of a universal gate set is $\{H, S, T, CNOT\}$ (Nielsen & Chuang, 2010), where H is the Hadamard gate, T is referred to as phase shift gate or $\pi/8$-gate, CNOT is called controlled-NOT gate, and

$$H = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}, \; S = \begin{bmatrix} 1 & 0 \\ 0 & i \end{bmatrix}, \; T = \begin{bmatrix} 1 & 0 \\ 0 & e^{i\frac{\pi}{4}} \end{bmatrix}, \; CNOT = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}.$$

Solovay and Kitaev (1997) separately showed that any one-qubit gate can be approximated to ϵ error by a product of $O\left(\log^c(1/\epsilon)\right)$ gates from a universal gate set, where c is a constant, which implies that approximating one-qubit gate using a universal gate set is efficient. Dawson and Nielsen (2005) systematically extended Solovay and Kitaev's results to broader classes of quantum gates, including notable 2-qubit gates. However, the necessary length of the gate series from a universal gate set to approximate an arbitrary n-qubit gate to a desired precision increases at an exponential rate as n grows (Harrow, Recht, & Chuang, 2002), indicating that some high dimensional unitary matrices are hard to implement in practice. Consequently, quantum algorithms typically avoid quantum gates that cannot be efficiently approximated.

In addition to the gates mentioned earlier, we list several commonly used quantum

gates below:

$$\sigma^x = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \text{ (Pauli-X gate)}, \ \sigma^y = \begin{bmatrix} 0 & i \\ -i & 0 \end{bmatrix} \text{ (Pauli-Y gate)},$$

$$\sigma^z = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} \text{ (Pauli-Z gate)}, \ CZ = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & -1 \end{bmatrix} \text{ (Controlled Z-gate)}.$$

1.2.5 Quantum Algorithm

A quantum algorithm provides a well-designed systematic procedure for carrying out operations on a quantum computer. It embeds adequate information regarding the solution to a problem within the quantum state of a quantum register, where enough information hidden in the state can be effectively accessed through measurement (Wang, 2012). A quantum algorithm generally describes the initialization of a quantum circuit, a series of quantum gates acting on the circuit, measurements of the quantum circuit, and the way to map the outcomes of the measurements to a solution of the concerned problem. The special properties of quantum systems, such as quantum superposition and entanglement, enable many quantum algorithms to outperform state-of-the-art classical algorithms (Marinescu, 2011). For example, Grover's algorithm (Grover, 1996) achieves quadratic speedup for database search, and Shor's algorithm (Shor, 1994) realizes an exponential acceleration in integer factorization. For more details about quantum algorithms, see Nielsen and Chuang (2010), Montanaro (2016), and Wang and Liu (2022). In the next section, we will introduce an example of quantum algorithms.

Chapter 2

Kitaev's Phase Estimation Algorithm

2.1 Quantum Phase Estimation

Quantum phase estimation refers to estimating the eigenphase ϕ in the eigenvalue $e^{i\phi}$ corresponding to an eigenvector $|\phi\rangle$ of a given unitary matrix U. It serves as a fundamental subroutine in a variety of efficient quantum algorithms (Shor, 1994; Temme et al., 2011; Wang, 2012; Ozols et al., 2013), so an efficient and robust phase estimation algorithm is pursued to accelerate the speed of these quantum algorithms. There are two popular methods for estimating the quantum phase (Svore et al., 2013). The first approach involves performing an inverse quantum Fourier transform to approximate the eigenvalues of the unitary matrix U (Abrams & Lloyd, 1999), which realizes a Fourier transform with complexity $O(\log^2(N))$ as compared to the classical complexity of $O(N\log(N))$ (Musk, 2020). This approach is efficient but not robust against the inevitable noise from quantum mechanics, making it difficult to implement in near-term quantum devices. The other approach involves running and measuring a parameterized quantum circuit multiple times to generate samples that provide information about the unknown eigenphase, then post-processing the data in classical computers to infer the eigenphase (Wiebe & Granade,

2016). This technique could be relatively more robust under noise and shows a potential to reduce the consumption of expensive quantum computational resources at the cost of more inexpensive classical computational resources. We hereinafter focus on the second approach, and in the next section, we will introduce a famous phase estimation algorithm proposed by Kitaev (1995).

2.2 Kitaev's Phase Estimation Algorithm

This section reviews Kitaev's phase estimation algorithm, following the structure in (Kitaev et al., 2002; Svore et al., 2013). Since the eigenvalue $e^{i\phi}$ has period 2π in ϕ, without loss of generality, we assume that $\phi \in [0, 2\pi)$. Kitaev's phase estimation algorithm generates samples from the following quantum circuit:

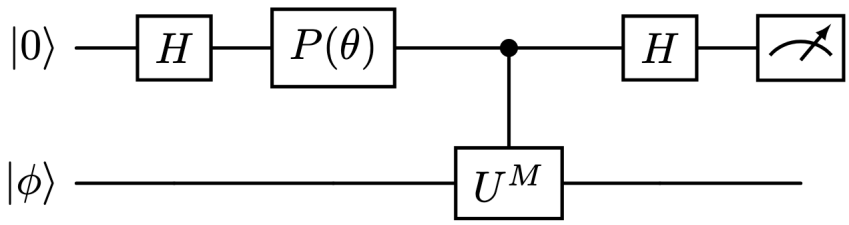

Figure 2.1: Circuit for Kitaev's phase estimation algorithm

where M and θ are hyperparameters of the circuit, $|\phi\rangle$ is the eigenvector associated with the concerned unknown eigenphase ϕ, U is the related unitary matrix, $H = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$ is the Hadamard gate, $P(\theta) = \begin{bmatrix} 1 & 0 \\ 0 & e^{i\theta} \end{bmatrix}$ represents the quantum phase gate, and the gate right after $P(\theta)$ is the controlled-U^M gate. A measurement on the first qubit will be conducted subsequent to the second Hadamard gate H, returning an outcome $E \in \{0, 1\}$. The binary distribution of the outcome E can be derived as follows:

1. The initial state of the quantum system is $|0\rangle \otimes |\phi\rangle$.

2. After applying the first Hadamard gate, the state of the first qubit becomes $H|0\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$.

3. After applying the $P(\theta)$ gate, the state of the first qubit becomes $\frac{1}{\sqrt{2}}(|0\rangle + e^{i\theta}|1\rangle)$.

4. After applying the controlled-U^M gate to the qubits, the state of the qubits becomes

$$\begin{aligned}\frac{1}{\sqrt{2}}\left[|0\rangle \otimes |\phi\rangle + e^{i\theta}|1\rangle \otimes (U^M|\phi\rangle)\right] &= \frac{1}{\sqrt{2}}\left(|0\rangle \otimes |\phi\rangle + e^{i\theta}|1\rangle \otimes (e^{iM\phi}|\phi\rangle)\right) \\ &= \frac{1}{\sqrt{2}}\left(|0\rangle + e^{i(M\phi+\theta)}|1\rangle\right) \otimes |\phi\rangle.\end{aligned}$$

5. After applying the second Hadamard gate, since

$$\frac{1}{\sqrt{2}} \cdot \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \begin{bmatrix} 1 \\ e^{i(M\phi+\theta)} \end{bmatrix} = \frac{1}{2} \begin{bmatrix} 1 + e^{i(M\phi+\theta)} \\ 1 - e^{i(M\phi+\theta)} \end{bmatrix},$$

the state of the quantum system becomes

$$\frac{1}{2}\left[(1 + e^{i(M\phi+\theta)})|0\rangle + (1 - e^{i(M\phi+\theta)})|1\rangle\right] \otimes |\phi\rangle.$$

6. Therefore, the probability that the outcome of the measurement is 0 equals

$$\begin{aligned}\left|\frac{1}{2}(1 + e^{i(M\phi+\theta)})\right|^2 &= \frac{1}{4}\left|1 + \cos(M\phi+\theta) + i\sin(M\phi+\theta)\right|^2 \\ &= \frac{1}{4}\left[(1 + \cos(M\phi+\theta))^2 + \sin^2(M\phi+\theta)\right] \\ &= \frac{1 + \cos(M\phi+\theta)}{2},\end{aligned}$$

and similarly, the probability of obtaining 1 is

$$\left|\frac{1}{2}\left(1 - e^{i(M\phi+\theta)}\right)\right|^2 = \frac{1 - \cos(M\phi+\theta)}{2}.$$

Thus, the probability of obtaining 0 and the probability of obtaining 1 can be denoted by

$$P_{M,\theta}(0|\phi) = \frac{1+\cos(M\phi+\theta)}{2},$$
$$P_{M,\theta}(1|\phi) = \frac{1-\cos(M\phi+\theta)}{2}, \text{ respectively.}$$

Let $\psi = \frac{\phi}{2\pi} \mod 1$. ψ is an unknown constant in $[0,1)$ which can be represented in binary expansion form, denoted by $\overline{.\alpha_1\alpha_2\alpha_3...}$,

$$\psi = \sum_{k=1}^{\infty} \frac{\alpha_k}{2^k}, \ \alpha_k \in \{0,1\}.$$

We further denote the sum of the first d terms of the binary expansion of ψ by $\psi_d = \overline{.\alpha_1\alpha_2...\alpha_d} = \sum_{k=1}^{d} \frac{\alpha_k}{2^k}$. For any given integer $m \geq 1$, Kitave's algorithm can provide an estimate $\widehat{\psi}_{m+2}$ for ψ_{m+2}, as detailed below (Kitaev et al., 2002; Svore et al., 2013):

1. Define $M_j = 2^{j-1}$. For each $j \in \{1,2,...,m\}$, run the circuit in Figure 2.1 s times with hyperparameters $(M,\theta) = (M_j, 0)$ and s times with hyperparameters $(M,\theta) = (M_j, \frac{\pi}{2})$, where s is a prespecified constant.

2. For each $j \in \{1,2,...,m\}$, calculate

$$P^*_{cos}(M_j) = \frac{N_c(0) - N_c(1)}{N_c},$$
$$P^*_{sin}(M_j) = \frac{N_s(1) - N_s(0)}{N_s},$$

   where $N_c = s$ represents the number of runs with $\theta = 0$, with $N_c(0)$ of them output 0 and $N_c(1)$ of them output 1, and $N_s = s$ represents the number of runs with $\theta = \pi/2$, with $N_s(0)$ of them output 0 and $N_s(1)$ of them output 1. Then, $\frac{P^*_{sin}(M_j)}{P^*_{cos}(M_j)}$ is an approximation for $\tan(M_j\phi)$, and an estimate for $(M_j\psi \mod 1)$ is thus given by

$$\rho_j = \begin{cases} \frac{\arctan\left(P^*_{sin}(M_i)/P^*_{cos}(M_i)\right)}{2\pi} \mod 1 & \text{if } P^*_{cos}(M_i) \geq 0 \\ \frac{\arctan\left(P^*_{sin}(M_i)/P^*_{cos}(M_i)\right)}{2\pi} + \frac{1}{2} & \text{if } P^*_{cos}(M_i) < 0. \end{cases}$$

3. An estimate
$$\widehat{\psi}_{m+2} \coloneqq \overline{.\beta_1\beta_2...\beta_{m+2}} = \sum_{k=1}^{m+2} \frac{\beta_k}{2^k}$$
for $\psi_{m+2} = \overline{.\alpha_1\alpha_2...\alpha_{m+2}}$ is then can be constructed as follows: choose proper $\beta_m, \beta_{m+1}, \beta_{m+2} \in \{0,1\}$ such that $\overline{.\beta_m\beta_{m+1}\beta_{m+2}}$ equals the value in $\{\frac{0}{8}, \frac{1}{8}, ..., \frac{7}{8}\}$ closest to ρ_m, and iteratively determine $\beta_j, j = m-1, m-2, ..., 1$ by set
$$\beta_j = \begin{cases} 0 & \text{if } |\overline{.0\beta_{j+1}\beta_{j+2}} - \rho_j| < \frac{1}{4} \\ 1 & \text{if } |\overline{.1\beta_{j+1}\beta_{j+2}} - \rho_j| < \frac{1}{4}. \end{cases}$$
If the estimate $\widehat{\psi}_{m+2}$ exactly matches ψ_{m+2}, $\widehat{\phi} \coloneqq 2\pi\widehat{\psi}_{m+2}$ is an estimator for the phase ϕ to precision $\frac{\pi}{2^{m+1}}$, i.e.
$$|\phi - \widehat{\phi}| \leq \frac{\pi}{2^{m+1}}.$$

2.3 Complexity of Kitaev's Phase Estimation Algorithm

As described in (Svore et al., 2013), given a fixed error probability δ, the total number of measurements required in Kitaev's algorithm to guarantee that $|\phi - \widehat{\phi}| < \frac{1}{2^m}$ with probability at least $1-\delta$ is $O(m\log(m))$. The proof of this statement can be phrased as follows:

1. First, recall that if $\widehat{\psi}_{m+3} = \psi_{m+3}$, $2\pi\widehat{\psi}_{m+3}$ is an estimate for ϕ to precision $\frac{\pi}{2^{m+2}} < \frac{1}{2^m}$. So we only need to show that the number of measurements required to guarantee that $\widehat{\psi}_{m+3} = \psi_{m+3}$ with probability $1-\delta$ is $O(m\log(m))$.

2. Since the total number of measurements for constructing $\widehat{\psi}_{m+3}$ is equal to $2(m+1)s$, it suffices to show that there exists $s = O(\log(m))$ such that $\widehat{\psi}_{m+3} = \psi_{m+3}$ with probability at least $1-\delta$.

3. The way we choose $\beta_j's$ assures that $\alpha_j = \beta_j$ for all $j \in \{1, 2, ..., m+3\}$ holds if $|\rho_j - (M_j\psi \mod 1)| < \frac{1}{16}$ simultaneously for all $j \in \{1, 2, ..., m+3\}$ (Kitaev et

al., 2002). Therefore, by Bonferroni's inequality and the properties of conditional probability, we only need to show that there exists $s = O(\log(m))$ such that the inequality $P(|\rho_j - (M_j\psi \mod 1)| > \frac{1}{16}) < \frac{\delta}{m+3}$ holds for all j, i.e. we need to prove the following lemma:

Lemma 2.1. *For any $\phi \in [0, 2\pi), \delta > 0$, $\exists s = O(\log(m))$, such that for any j,*

$$P(|\rho_j - (M_j\psi \mod 1)| > \frac{1}{16}) < \frac{\delta}{m+3}.$$

The proof of Lemma 2.1 is left in **Appendix**.

Svore et al. (2013) proposed a series of extensions of Kitaev's algorithm, where the basic version of their extensions only requires $O\left(m \log(\log(m))\right)$ measurements to output an estimate for the phase to precision $\frac{1}{2^m}$ with desired high probability, at the cost of more classical computation. Since the algorithms by Kitaev and Svore et al. are based on sampling from a low-depth quantum circuit, they are typically more robust to noise than Quantum Fourier Transform-based phase estimation algorithms. More enhanced extensions of Kitaev's algorithm can be found in (Kaftal & Demkowicz-Dobrzański, 2014; Mohammadbagherpoor et al., 2019)

Chapter 3

Bayesian Phase Estimation Algorithm

3.1 Bayesian Phase Estimation Algorithm

Wiebe and Granade (2016) proposed a phase estimation algorithm that combines Bayesian inference and rejection sampling, exhibiting notable efficiency and robustness (Paesani et al., 2017; Li et al., 2018; Gebhart, Smerzi, & Pezzè, 2021; Yamamoto, Duffield, Kikuchi, & Muñoz Ramo, 2024). This algorithm samples from a quantum circuit similar to the one used in Kitaev's phase estimation algorithm, as shown in Figure 3.1:

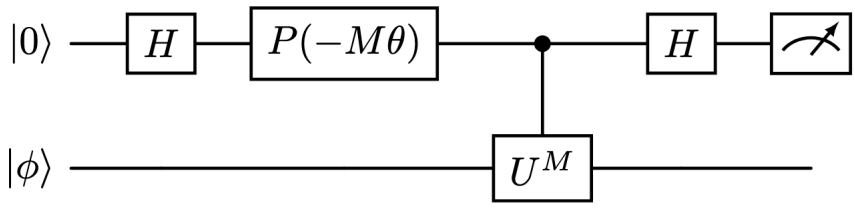

Figure 3.1: Circuit for Bayesian phase estimation algorithm

where $P(-M\theta) = \begin{bmatrix} 1 & 0 \\ 0 & e^{-iM\theta} \end{bmatrix}$. The binary distribution of the output E of this circuit is

given by (Paesani et al., 2017)

$$P_{M,\theta}(E|\phi) = \begin{cases} \frac{1+\cos(M[\phi-\theta])}{2} & \text{when } E=0 \\ \frac{1-\cos(M[\phi-\theta])}{2} & \text{when } E=1. \end{cases}$$

In Bayesian analysis, the unknown phase ϕ is viewed as a random variable. When adopting a Gaussian prior $\mathcal{N}(\mu, \sigma)$ for ϕ, starting with $(\mu, \sigma^2) = (\mu_0, \sigma_0^2)$, the Bayesian phase estimation algorithm iteratively update the hyperparameters (μ, σ^2) as follows:

1. Set the parameters θ_i and M_i of the circuit through a prespecified rule.

2. Run and measure the circuit to generate an observation $E_i \in \{0, 1\}$.

3. Sample m data points $\upsilon_{i,1}, \upsilon_{i,2}, ..., \upsilon_{i,m}$ from $\mathcal{N}(\mu_{i-1}, \sigma_{i-1}^2)$, where the sample size m is a customized constant.

4. (rejection filtering) For each $j \in \{1, 2, ..., m\}$, assign $\upsilon_{i,j}$ to set $\mathbf{\Phi}^i_{accept}$ with probability $P_{M_i,\theta_i}(E_i|\upsilon_{i,j})/\kappa$, where $\kappa \in (0, 1]$ is a scaling constant such that

$$P_{M_i,\theta_i}(E_i|\upsilon_{i,j})/\kappa \leq 1 \text{ for all } i, j.$$

5. Return $\mu_i = \mathbb{E}(\mathbf{\Phi}^i_{accept})$ and $\sigma_i^2 = \mathbb{V}(\mathbf{\Phi}^i_{accept})$.

Wiebe and Granade (2016) recommended setting the initial prior of ϕ to be $\mathcal{N}(\pi, \pi^2)$ and updating hyperparameters θ and M of the quantum circuit through the following rules:

$$\theta_i \sim \mathbb{N}(\mu_{i-1}, \sigma_{i-1}^2), M_i = \lceil \frac{1.25}{\sigma_{i-1}} \rceil.$$

Although the rejection filtering step is crucial when adopting other prior distributions like the wrapped-Gaussian distribution suggested in (Wiebe & Granade, 2016), it is avoidable when using the Gaussian prior since close-formed expressions for the posterior mean

and posterior variance can be derived. Denote the prior distribution of ϕ by $p(\phi)$, where

$$p(\phi) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{1}{2}(\frac{\phi-\mu}{\sigma})^2},$$

and let

$$A_0 = \int_{-\infty}^{\infty} \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{1}{2}(\frac{\phi-\mu}{\sigma})^2} \frac{1+\cos(M[\phi-\theta])}{2} d\phi$$
$$A_1 = \int_{-\infty}^{\infty} \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{1}{2}(\frac{\phi-\mu}{\sigma})^2} \frac{1-\cos(M[\phi-\theta])}{2} d\phi,$$

the posterior distribution of the phase ϕ given an output E of the quantum circuit in Figure 3.1 is

$$\tilde{p}(\phi|E;\theta,M) = \begin{cases} \frac{1}{2\sqrt{2\pi}\sigma A_0}(1+\cos(M[\phi-\theta]))e^{-\frac{1}{2}(\frac{\phi-\mu}{\sigma})^2} & \text{if } E=0 \\ \frac{1}{2\sqrt{2\pi}\sigma A_1}(1-\cos(M[\phi-\theta]))e^{-\frac{1}{2}(\frac{\phi-\mu}{\sigma})^2} & \text{if } E=1. \end{cases}$$

To find the explicit form of the posterior distribution $\tilde{p}(\phi|E;\theta,M)$, we now simplify A_0 and A_1:

$$\begin{aligned} A_0 &= \frac{1}{\sqrt{2\pi}\sigma} \int_{-\infty}^{\infty} e^{-\frac{1}{2}(\frac{\phi-\mu}{\sigma})^2} \frac{1+\cos(M[\phi-\theta])}{2} d\phi \\ &= \frac{1}{2\sqrt{2\pi}\sigma} \int_{-\infty}^{\infty} e^{-\frac{1}{2}(\frac{\phi-\mu}{\sigma})^2} d\phi + \frac{1}{2\sqrt{2\pi}\sigma} \int_{-\infty}^{\infty} \cos(M[\phi-\theta]) e^{-\frac{1}{2}(\frac{\phi-\mu}{\sigma})^2} d\phi \\ &= \frac{1}{2} + \frac{1}{2\sqrt{2\pi}\sigma} \int_{-\infty}^{\infty} \cos(M[\phi-\theta]) e^{-\frac{1}{2}(\frac{\phi-\mu}{\sigma})^2} d\phi \\ &= \frac{1}{2} + \frac{1}{2\sqrt{2\pi}\sigma} \int_{-\infty}^{\infty} \cos(M[\phi-\theta+\mu]) e^{-\frac{1}{2}(\frac{\phi}{\sigma})^2} d\phi \\ &= \frac{1}{2} + \frac{1}{2\sqrt{2\pi}} \int_{-\infty}^{\infty} \cos(M\sigma\phi - M(\theta-\mu)) e^{-\frac{1}{2}\phi^2} d\phi. \end{aligned}$$

Let

$$\alpha = M\sigma, \beta = -M(\theta-\mu),$$

We have

$$A_0 = \frac{1}{2} + \frac{1}{2\sqrt{2\pi}} \int_{-\infty}^{\infty} \cos(\alpha\phi+\beta) e^{-\frac{1}{2}\phi^2} d\phi.$$

Similarly,

$$A_1 = \frac{1}{2} - \frac{1}{2\sqrt{2\pi}} \int_{-\infty}^{\infty} \cos(\alpha\phi + \beta) e^{-\frac{1}{2}\phi^2} d\phi.$$

To continue the calculation, we need the following lemma, whose proof is left in **Appendix**:

Lemma 3.1.

$$\int_{-\infty}^{\infty} \cos(\alpha\phi + \beta) e^{-\frac{1}{2}\phi^2} d\phi = \sqrt{2\pi} \cos(\beta) e^{-\frac{1}{2}\alpha^2}$$
$$\int_{-\infty}^{\infty} \sin(\alpha\phi + \beta) e^{-\frac{1}{2}\phi^2} d\phi = \sqrt{2\pi} \sin(\beta) e^{-\frac{1}{2}\alpha^2}.$$

Applying Lemma 3.1, we have

$$\begin{aligned} A_0 &= \frac{1}{2} + \frac{1}{2\sqrt{2\pi}} \int_{-\infty}^{\infty} \cos(\alpha\phi + \beta) e^{-\frac{1}{2}\phi^2} d\phi \\ &= \frac{1}{2} + \frac{1}{2} \cos(\beta) e^{-\frac{1}{2}\alpha^2} \\ &= \frac{1}{2} + \frac{1}{2} \cos(M(\theta - \mu)) e^{-\frac{1}{2}M^2\sigma^2}, \end{aligned}$$

and

$$\begin{aligned} A_1 &= \int_{-\infty}^{\infty} \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{1}{2}(\frac{\phi-\mu}{\sigma})^2} \frac{1 - \cos(M[\phi + \theta])}{2} d\phi \\ &= \frac{1}{2} - \frac{1}{2} \cos(\beta) e^{-\frac{1}{2}\alpha^2} \\ &= \frac{1}{2} - \frac{1}{2} \cos(M(\theta - \mu)) e^{-\frac{1}{2}M^2\sigma^2}. \end{aligned}$$

Therefore, when the observed value $E = 0$, the posterior distribution of ϕ is

$$\begin{aligned} \tilde{p}(\phi | E = 0; \theta, M) &= \frac{1}{2\sqrt{2\pi}\sigma A_0} (1 + \cos(M[\phi - \theta])) e^{-\frac{1}{2}(\frac{\phi-\mu}{\sigma})^2} \\ &= \frac{1 + \cos(M[\phi - \theta])}{\sqrt{2\pi}\sigma(1 + \cos(M[\mu - \theta]) e^{-\frac{1}{2}M^2\sigma^2})} e^{-\frac{1}{2}(\frac{\phi-\mu}{\sigma})^2}, \end{aligned}$$

and when $E = 1$, the posterior distribution of ϕ is

$$\begin{aligned} \tilde{p}(\phi | E = 1; \theta, M) &= \frac{1}{2\sqrt{2\pi}\sigma A_1} (1 - \cos(M[\phi - \theta])) e^{-\frac{1}{2}(\frac{\phi-\mu}{\sigma})^2} \\ &= \frac{1 - \cos(M[\phi - \theta])}{\sqrt{2\pi}\sigma(1 - \cos(M[\mu - \theta]) e^{-\frac{1}{2}M^2\sigma^2})} e^{-\frac{1}{2}(\frac{\phi-\mu}{\sigma})^2}. \end{aligned}$$

The posterior mean $\tilde{\mu}_{E=0}$ when $E = 0$ then can be calculated as

$$\begin{aligned}
\tilde{\mu}_{E=0} &= \frac{1}{2\sqrt{2\pi}\sigma A_0}\int_{-\infty}^{\infty}\phi(1+\cos(M[\phi-\theta]))e^{-\frac{1}{2}(\frac{\phi-\mu}{\sigma})^2}d\phi \\
&= \frac{1}{2\sqrt{2\pi}\sigma A_0}\left(\int_{-\infty}^{\infty}\phi e^{-\frac{1}{2}(\frac{\phi-\mu}{\sigma})^2}d\phi + \int_{-\infty}^{\infty}\phi\cos(M[\phi-\theta])e^{-\frac{1}{2}(\frac{\phi-\mu}{\sigma})^2}d\phi\right) \\
&= \frac{\mu}{2A_0} + \frac{1}{2\sqrt{2\pi}\sigma A_0}\int_{-\infty}^{\infty}\phi\cos(M[\phi-\theta])e^{-\frac{1}{2}(\frac{\phi-\mu}{\sigma})^2}d\phi \\
&= \frac{\mu}{2A_0} + \frac{1}{2\sqrt{2\pi}A_0}\left(\sigma\int_{-\infty}^{\infty}\phi\cos(\alpha\phi+\beta)e^{-\frac{1}{2}\phi^2}d\phi + \mu\int_{-\infty}^{\infty}\cos(\alpha\phi+\beta)e^{-\frac{1}{2}\phi^2}d\phi\right) \\
&= \frac{\mu}{2A_0} + \frac{1}{2\sqrt{2\pi}A_0}\left(-\alpha\sigma\int_{-\infty}^{\infty}\sin(\alpha\phi+\beta)e^{-\frac{1}{2}\phi^2}d\phi + \sqrt{2\pi}\mu\cos(\beta)e^{-\frac{1}{2}\alpha^2}\right) \\
&= \frac{\mu}{2A_0} + \frac{1}{2\sqrt{2\pi}A_0}\left(-\sqrt{2\pi}\alpha\sigma\sin(\beta)e^{-\frac{1}{2}\alpha^2} + \sqrt{2\pi}\mu\cos(\beta)e^{-\frac{1}{2}\alpha^2}\right) \\
&= \frac{1}{2A_0}\left[\mu + (\mu\cos(\beta) - \alpha\sigma\sin(\beta))\,e^{-\frac{1}{2}\alpha^2}\right] \\
&= \frac{\mu + (\mu\cos(\beta) - \alpha\sigma\sin(\beta))\,e^{-\frac{1}{2}\alpha^2}}{1+\cos(\beta)e^{-\frac{1}{2}\alpha^2}} \\
&= \mu - \frac{\alpha\sigma\sin(\beta)}{e^{\frac{1}{2}\alpha^2}+\cos(\beta)} \\
&= \mu - \frac{M\sigma^2\sin(M(\mu-\theta))}{e^{\frac{M^2\sigma^2}{2}}+\cos(M(\mu-\theta))}
\end{aligned}$$

Similarly, when $E = 1$, the posterior mean $\tilde{\mu}_{E=1}$ is equal to

$$\begin{aligned}
\tilde{\mu}_{E=1} &= \frac{1}{2\sqrt{2\pi}\sigma A_1}\int_{-\infty}^{\infty}\phi(1-\cos(M[\phi-\theta]))e^{-\frac{1}{2}(\frac{\phi-\mu}{\sigma})^2}d\phi \\
&= \frac{1}{2A_1}\left[\mu - (\mu\cos(\beta) - \alpha\sigma\sin(\beta))\,e^{-\frac{1}{2}\alpha^2}\right] \\
&= \frac{\mu - (\mu\cos(\beta) - \alpha\sigma\sin(\beta))\,e^{-\frac{1}{2}\alpha^2}}{1-\cos(\beta)e^{-\frac{1}{2}\alpha^2}} \\
&= \mu + \frac{\alpha\sigma\sin(\beta)}{e^{\frac{1}{2}\alpha^2}-\cos(\beta)} \\
&= \mu + \frac{M\sigma^2\sin(M(\mu-\theta))}{e^{\frac{M^2\sigma^2}{2}}-\cos(M(\mu-\theta))}.
\end{aligned}$$

Now, we calculate the posterior variance of ϕ. When $E = 0$, the posterior variance

when $E = 0$ can be expressed as

$$\tilde{\sigma}^2_{E=0} = \frac{1}{2\sqrt{2\pi}\sigma A_0} \int_{-\infty}^{\infty} \phi^2 (1 + \cos(M[\phi - \theta])) e^{-\frac{1}{2}(\frac{\phi-\mu}{\sigma})^2} d\phi - \tilde{\mu}^2_{E=0},$$

where

$$\begin{aligned}
&\frac{1}{\sqrt{2\pi}\sigma} \int_{-\infty}^{\infty} \phi^2 (1 + \cos(M[\phi - \theta])) e^{-\frac{1}{2}(\frac{\phi-\mu}{\sigma})^2} d\phi \\
&= \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} (\sigma\phi + \mu)^2 (1 + \cos(M[\sigma\phi - \theta + \mu])) e^{-\frac{1}{2}\phi^2} d\phi \\
&= \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} (\sigma^2\phi^2 + 2\mu\sigma\phi + \mu^2)(1 + \cos(\alpha\phi + \beta)) e^{-\frac{1}{2}\phi^2} d\phi \\
&= \frac{1}{\sqrt{2\pi}} \Bigg(\sigma^2 \int_{-\infty}^{\infty} \phi^2 e^{-\frac{1}{2}\phi^2} d\phi + 2\mu\sigma \int_{-\infty}^{\infty} \phi e^{-\frac{1}{2}\phi^2} d\phi + \mu^2 \int_{-\infty}^{\infty} e^{-\frac{1}{2}\phi^2} d\phi \\
&+ \sigma^2 \int_{-\infty}^{\infty} \phi^2 \cos(\alpha\phi + \beta) e^{-\frac{1}{2}\phi^2} d\phi + 2\mu\sigma \int_{-\infty}^{\infty} \phi \cos(\alpha\phi + \beta) e^{-\frac{1}{2}\phi^2} d\phi \\
&+ \mu^2 \int_{-\infty}^{\infty} \cos(\alpha\phi + \beta) e^{-\frac{1}{2}\phi^2} d\phi \Bigg) \\
&= \sigma^2 + \mu^2 - 2\alpha\mu\sigma \sin(\beta) e^{-\frac{1}{2}\alpha^2} + \mu^2 \cos(\beta) e^{-\frac{1}{2}\alpha^2} + \frac{\sigma^2}{\sqrt{2\pi}} \int_{-\infty}^{\infty} \phi^2 \cos(\alpha\phi + \beta) e^{-\frac{1}{2}\phi^2} d\phi \\
&= \sigma^2 + \mu^2 - 2\alpha\mu\sigma \sin(\beta) e^{-\frac{1}{2}\alpha^2} + \mu^2 \cos(\beta) e^{-\frac{1}{2}\alpha^2} + \sigma^2(1 - \alpha^2) \cos(\beta) e^{-\frac{1}{2}\alpha^2} \\
&= 2A_0(1 - \alpha^2)\sigma^2 + 2A_0\mu^2 + \alpha^2\sigma^2 - 2\alpha\mu\sigma \sin(\beta) e^{-\frac{1}{2}\alpha^2},
\end{aligned}$$

Thus,

$$\begin{aligned}
\tilde{\sigma}^2_{E=0} &= (1 - \alpha^2)\sigma^2 + \mu^2 + \frac{1}{2A_0}(\alpha^2\sigma^2 - 2\alpha\mu\sigma \sin(\beta) e^{-\frac{1}{2}\alpha^2}) - \tilde{\mu}^2_{E=0} \\
&= (1 - \alpha^2)\sigma^2 + \frac{1}{2A_0}\alpha^2\sigma^2 - \frac{1}{4A_0^2}\alpha^2\sigma^2 \sin^2(\beta) e^{-\alpha^2} \\
&= \left(1 - \alpha^2 + \frac{\alpha^2}{2A_0} - \frac{1}{4A_0^2}\alpha^2 \sin^2(\beta) e^{-\alpha^2}\right) \sigma^2 \\
&= \left(1 - \alpha^2 + \frac{\alpha^2}{1 + \cos(\beta) e^{-\frac{1}{2}\alpha^2}} - \frac{\alpha^2 \sin^2(\beta) e^{-\alpha^2}}{(1 + \cos(\beta) e^{-\frac{1}{2}\alpha^2})^2}\right) \sigma^2 \\
&= \left(1 - M^2\sigma^2 + \frac{M^2\sigma^2}{1 + \cos(M(\mu - \theta)) e^{-\frac{1}{2}M^2\sigma^2}} - \frac{M^2\sigma^2 \sin^2(M(\mu - \theta)) e^{-M^2\sigma^2}}{(1 + \cos(M(\mu - \theta)) e^{-\frac{1}{2}M^2\sigma^2})^2}\right) \sigma^2.
\end{aligned}$$

In the same manner, we can show that when $E = 1$, the posterior variance

$$\begin{aligned}\tilde{\sigma}^2_{E=1} &= (1-\alpha^2)\sigma^2 + \mu^2 + \frac{1}{2A_1}(\alpha^2\sigma^2 + 2\alpha\mu\sigma\sin(\beta)e^{-\frac{1}{2}\alpha^2}) - \tilde{\mu}^2_{E=1} \\ &= \left(1 - \alpha^2 + \frac{\alpha^2}{2A_1} - \frac{1}{4A_1^2}\alpha^2\sin^2(\beta)e^{-\alpha^2}\right)\sigma^2 \\ &= \left(1 - M^2\sigma^2 + \frac{M^2\sigma^2}{1-\cos(M(\mu-\theta))e^{-\frac{1}{2}M^2\sigma^2}} - \frac{M^2\sigma^2\sin^2(M(\mu-\theta))e^{-M^2\sigma^2}}{(1-\cos(M(\mu-\theta))e^{-\frac{1}{2}M^2\sigma^2})^2}\right)\sigma^2.\end{aligned}$$

3.2 Simulation Results of Bayesian Phase Estimation Algorithm

In this section, we perform several simulations to develop insights into the performance of the Bayesian phase estimation algorithm. The setting of the first simulation scheme is as follows:

- Recall that m represents the number of data points generated in the rejection filtering step at each iteration of the Bayesian phase estimation algorithm, we carry out simulations with three distinct values of m: $m = 50$, $m = 100$, and $m = 300$. Furthermore, we can immediately derive a simple analytical formula-based algorithm by replacing the rejection filtering step with the close-form expressions of the posterior mean and variance discussed in the previous section. We will also perform a simulation with this analytical formula-based algorithm alongside the other three scenarios.

- In each of the four scenarios, we generate 1000 8×8 unitary matrices, labeled as $U_j,\ j = 1, 2, ..., 1000$, randomly select an eigenvector $|\phi_{(j)}\rangle$ for each matrix U_j, and then compute the exact value of the phase $\phi_{(j)}$ corresponding to this eigenvector using classical algorithm.

- For each U_j, we run the Bayesian phase estimation algorithm with 120 iterations. The initial prior for phase $\phi_{(j)}$ is set to be $\mathcal{N}(\pi, \pi^2)$. Denote the posterior mean

after k iterations by $\hat{\phi}^k_{(j)}$, the absolute error $e^k_{(j)}, k = 0, 1, 2, ..., 120$, is defined by

$$e^0_{(j)} = |\pi - \phi_{(j)}| \mod 2\pi, e^k_{(j)} = |\hat{\phi}^k_{(j)} - \phi_{(j)}| \mod 2\pi, k = 1, 2, ..., 120.$$

- We then calculate the median absolute error and the mean absolute error as benchmarks of the algorithm. The median absolute error corresponding to the k_{th} iteration is given by

$$\text{Median Absolute Error}_k = \text{the median of } \{e^k_{(1)}, e^k_{(2)}, ..., e^k_{(1000)}\},$$

and the mean absolute error corresponding to the k_{th} iteration is given by

$$\text{Mean Absolute Error}_k = \frac{1}{1000}\sum_{j=1}^{1000} e^k_{(j)}.$$

- The trend of median absolute errors and the trend of mean absolute errors are then visualized in Figure 3.2 and Figure 3.3, respectively, where the y-coordinate in Figure 3.2 represents the median absolute error in log-scale, and the y-coordinate in Figure 3.3 represents the mean absolute error in log-scale.

As depicted in Figure 3.2, the median absolute error drops quickly at a nearly exponential rate as the number of iterations increases. The "m = 100", "m = 300", and "analytical formula" settings exhibit similar performance, but the "m = 50" setting shows a slower decrease in median absolute error, suggesting that an excessively low m may reduce the efficiency of the algorithm. However, as shown in Figure 3.3 and as mentioned in (Wiebe & Granade, 2016), the decrease in mean absolute error is notably slow; even after 120 iterations, the mean absolute errors still remain at a magnitude of 10^{-1}. Furthermore, even if it utilizes more computational resources, the "m=300" case performs worse than both the "m=50" and "m=100" cases, and the algorithm that relies on the analytical formula has the highest mean absolute error.

To explore why the rate of decrease in mean absolute errors is such slow, for each of

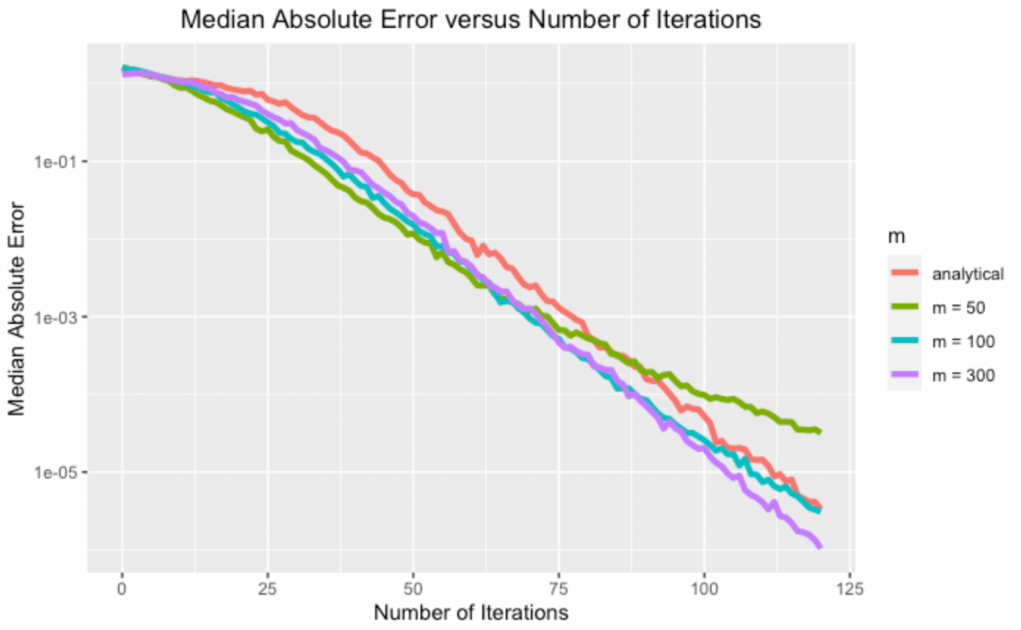

Figure 3.2: Median Absolute Error versus Number of Iterations

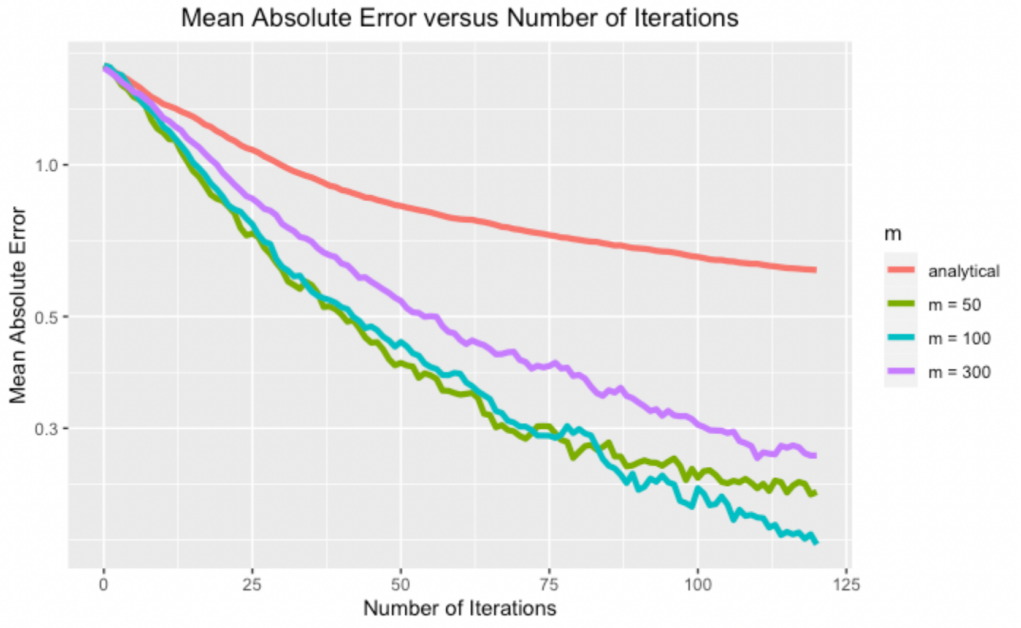

Figure 3.3: Mean Absolute Error versus Number of Iterations

the four setups (m = 50, m = 100, m = 300, analytical formula), we draw a histogram for the 1000 absolute errors corresponding to the 50_{th} iteration (i.e. $e^{50}_{(j)}, j = 1, 2, ..., 1000$), as well as a histogram for the 1000 absolute errors corresponding to the 100_{th} iteration (i.e. $e^{100}_{(j)}, j = 1, 2, ..., 1000$). The histograms are shown below (Figure 3.4-3.7):

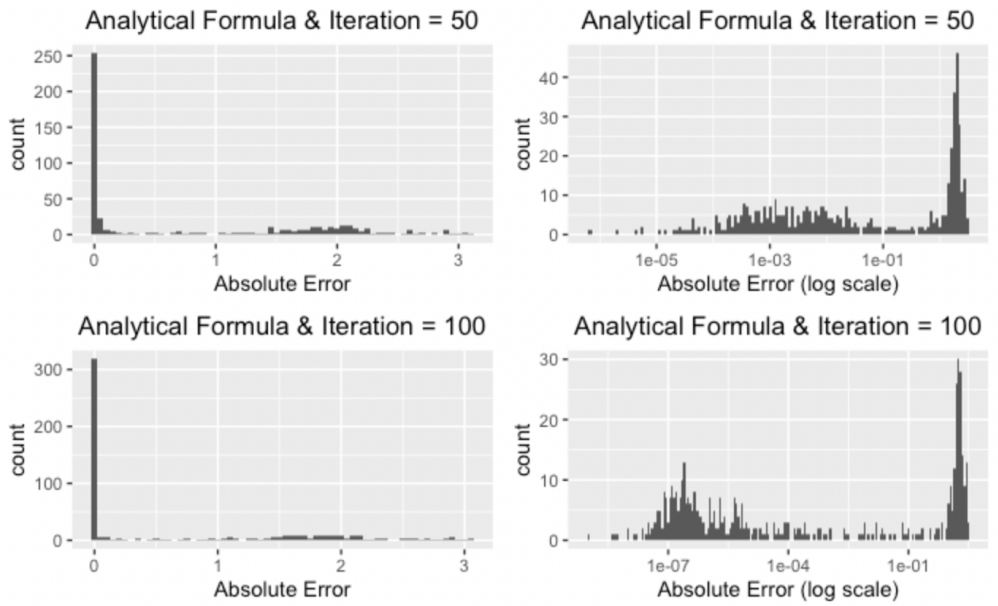

Figure 3.4: Absolute Errors of the "Analytical-Formula-Based" Scheme

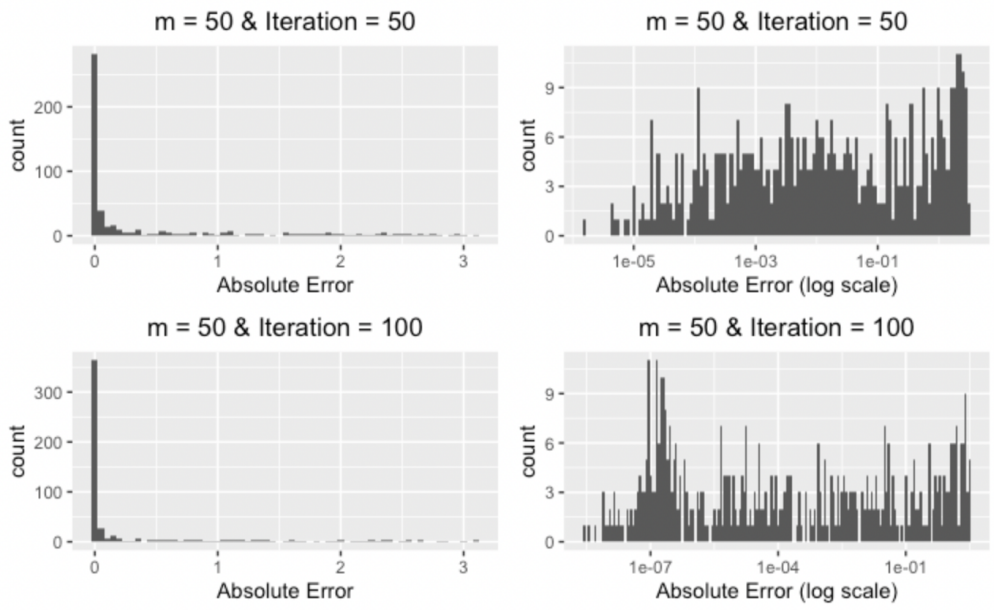

Figure 3.5: Absolute Errors of the "m = 50" Scheme

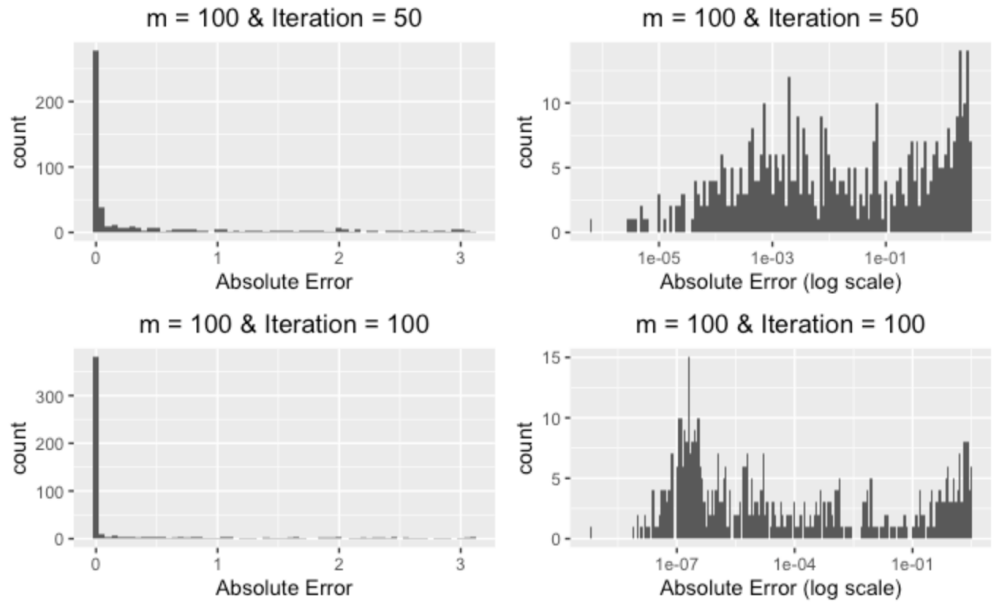

Figure 3.6: Absolute Errors of the "m = 100" Scheme

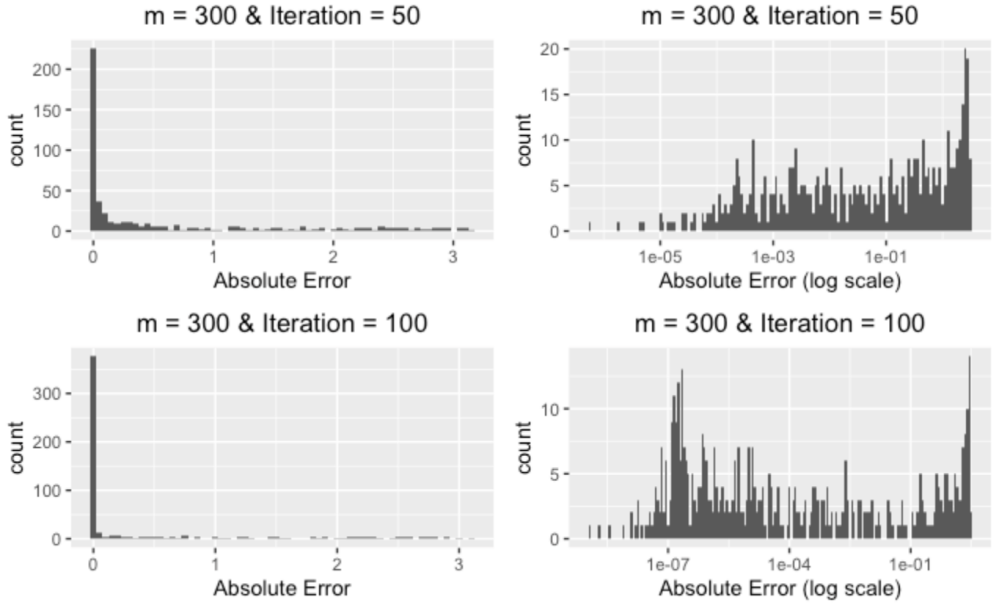

Figure 3.7: Absolute Errors of of the "m = 300" Scheme

These figures illustrate that there is a substantial number of absolute errors that are at a magnitude of 1, which significantly leverages the mean absolute errors. These large absolute errors diminish very slowly, especially in the "m = 300" case and the "analytical-formula-based" case, in which the number of absolute errors close to π at the 100th iteration remains comparable to that at the 50th iteration, resulting in the slow speed of decrease of the mean absolute errors. These simulation results indicate that in cases, the iteratively refined estimate μ_i for ϕ can be trapped at an incorrect value. To overcome this problem, Wiebe and Granade (2016) proposed a restart strategy involving monitoring the progress of the estimation and restarting the algorithm when it gets stuck. The main idea of their restart strategy is to examine the decay rate of the uncertainty σ, and when this rate is below a certain threshold, check whether μ is a relatively precise estimate for the phase ϕ. This strategy turns out to be effective (Wiebe & Granade, 2016; Paesani et al., 2017). For additional information regarding Bayesian-based quantum phase estimation algorithms, see (Li et al., 2018; Martínez-García, Vodola, & Müller, 2019; Gebhart et al., 2021; Yamamoto et al., 2024).

Chapter 4

Quantum Annealing

4.1 Simulated Annealing

4.1.1 Introduction to Simulated Annealing

This chapter discusses the quantum annealing algorithm. First, we introduce the simulated annealing algorithm which inspired the development of quantum annealing. Simulated annealing (Kirkpatrick, Gelatt Jr, & Vecchi, 1983; Černỳ, 1985) is a meta-heuristic algorithm designed to tackle global optimization problems, particularly combinatorial minimization problems whose solution space can be vast and filled with numerous local minima. The term "annealing" refers to a process in materials science and metallurgy involving heating a solid to a very high temperature to accelerate the movement of the atoms, allowing fast redistribution of the atoms, and then cooling the material according to a cooling schedule to "freeze" its structure. The solid will reach its lowest thermodynamic free energy configuration if it is heated to a sufficiently high temperature at the beginning of the annealing process and cooled slowly enough to allow it to reach thermal equilibrium at each temperature T_{ther}, where the thermal equilibrium is characterized by the Boltzmann distribution of the current state or configuration $\boldsymbol{s}$ of the solid (Metropolis, Rosenbluth, Rosenbluth, Teller, & Teller, 1953):

$$P(\boldsymbol{s} = s) = \frac{1}{Z(T_{ther})} e^{-\frac{E_s}{k_B T_{ther}}}, s \in \mathcal{S}, \tag{4.1}$$

where $\mathcal{S}$ is a set of all the possible configurations of the solid, E_s denotes the energy of the solid corresponding to the configuration s, k_B is a generic physical constant called Boltzmann constant, and

$$Z(T_{ther}) = \sum_{s' \in \mathcal{S}} e^{-\frac{E_{s'}}{k_B T_{ther}}}$$

is the normalization factor. Simulated annealing imitates the physical annealing process to solve optimization problems. We restrict our discussion to discrete optimization problems. Further noted that constrained optimization problems can be handled by adding penalty terms to the objective functions to convert to unconstrained optimization problems, we hereinafter focus on the following unconstrained combinatorial optimization problem:

$$\min_{s \in \mathcal{S}} \quad f(s), \tag{4.2}$$

where $\mathcal{S}$ is the search space analogous to the set of all the possible configurations of a solid, and $f(s)$ is the objective function analogous to the thermodynamic free energy of the solid. The classical version of the simulated annealing algorithm starts with an *initial solution* s_0, which can be viewed as the configuration of a solid before cooling, an *initial temperature* T_0, which is a control parameter corresponding to the initial temperature of the physical annealing process, and a neighborhood structure $\mathcal{N}$ defined on $\mathcal{S}$ such that

(i) The cardinality $|\mathcal{N}(s)|$ is a constant for all $s \in \mathcal{S}$.

(ii) For any $s, s' \in \mathcal{S}$, there exists $l \in \mathbb{Z}^+$, such that $s_l = s'$, $s \in \mathcal{N}(s_1)$, $s_{i-1} \in \mathcal{N}(s_i), i = 2, ..., l$.

The algorithm iteratively refines the current solution s by first uniformly randomly generating a candidate solution s' from $\mathcal{N}(s)$ and then determining whether to accept or not this candidate solution. The acceptance probability is given by the *Metropolis criterion* (Metropolis et al., 1953),

$$P(\text{accpect } s' \text{ as the next solution}) = \begin{cases} 1 & \text{if } f(s') \leq f(s) \\ \exp\left(-\frac{f(s')-f(s)}{T}\right) & \text{if } f(s') > f(s), \end{cases} \tag{4.3}$$

where T is the *temperature* of the current iteration. If the candidate solution is accepted, the algorithm transitions the current solution to the candidate solution; otherwise, the current solution remains unchanged. This procedure will repeated several times at temperature T until an equilibrium state, which is the stationary distribution of a reversible Markov chain, is ideally approximately reached

$$P(\text{the current solution} = s) = \frac{1}{Z(T)} e^{-\frac{f(s)}{T}}, s \in \mathcal{S}, \tag{4.4}$$

where

$$Z(T) = \sum_{s' \in \mathcal{S}} e^{-\frac{f(s')}{T}}$$

The temperature T is then lowered according to a *cooling schedule*. Along this line, the temperature T gradually decreases throughout the iterations, and the algorithm terminates when the temperature is close enough to 0.

The Metropolis acceptance criterion (4.3) enables the algorithm to temporarily accept a "worse" solution, which corresponds to a larger objective function value compared to the current solution. This strategy helps the algorithm avoid being trapped in a local minima. When the temperature T is high, the algorithm is more likely to accept a worse solution, thus able to explore the solution space thoroughly. As the temperature decreases, the algorithm will increasingly favor a "better" solution and eventually freeze on a local minimum solution, where this local minimum solution is guaranteed to be a global minimum solution with probability 1 if the cooling schedule is sufficiently slow (Hajek, 1988; Connors & Kumar, 1989). The pseudo-code of the classical simulated annealing algorithm is shown in Algorithm 1 (Laarhoven & Aarts, 1987; Henderson, Jacobson, & Johnson, 2003; Delahaye, Chaimatanan, & Mongeau, 2019).

Algorithm 1 Simulated Annealing

▷ s_0: the initial solution
▷ T_0: the initial temperature
▷ T_k: the k_{th} temperature generated according to the cooling schedule
▷ L_k: the number of iterations executed at temperature T_k
▷ $\mathcal{N}(s)$: the neighborhood of s associated with the pre-specified neighborhood structure

1: $k \leftarrow 0, s \leftarrow s_0$
2: **while** stopping criterion is not met **do**
3: $count \leftarrow 0$
4: **for** $count < L_k$ **do**
5: Generate a candidate solution s' from the neighborhood $\mathcal{N}(s)$ of s.
6: **if** $f(s') \leq f(s)$ **then**
7: $s \leftarrow s'$
8: **else if** Uniform$(0,1) \leq \exp\left(-\frac{f(s')-f(s)}{T_k}\right)$ **then**
9: $s \leftarrow s'$
10: **end if**
11: $count \leftarrow count + 1$
12: **end for**
13: $k \leftarrow k + 1$
14: **end while**
15: **return** s

4.1.2 Mathematical Model of Simulated Annealing

This section presents mathematical formulations of various components of the simulated annealing algorithm, including neighborhood structure, including *neighborhood structure*, *generation mechanism, acceptance criterion*, *cooling schedule*, and *Markov chain length*. Additionally, it discusses practical insights regarding the selection of these components and provides findings related to the convergence properties of the algorithm.

- *neighborhood structure*: A *neighborhood structure* $\mathcal{N}$ is a set-valued function which maps each configuration $s \in \mathcal{S}$ to a subset of $\mathcal{S}\backslash\{s\}$. In general, a neighborhood structure is required to satisfy the following two conditions:

  (N1) For any $s, s' \in \mathcal{S}$, $s \in \mathcal{N}(s')$ if and only if $s' \in \mathcal{N}(s)$.

  (N2) For any $s, s' \in \mathcal{S}$, there exists $l \in \mathbb{Z}^+$, such that $s_l = s'$, $s \in \mathcal{N}(s_1)$, $s_{i-1} \in \mathcal{N}(s_i), i = 2, ..., l$.

A neighborhood structure is typically defined by manually describing a rule for generating a *neighborhood* $\mathcal{N}(s)$ from each configuration s, like randomly modifying one component or swapping two components of s. The selection of the neighborhood structure is tied closely to the category of the combinatorial optimization problem. Different neighborhood structures can be employed for a particular type of combinatorial optimization problem, and the selection of these neighborhood structures can substantially impact the performance of the simulated annealing algorithm (Fleischer & Jacobson, 1999). More results for neighborhood structure selection can be found in Henderson et al. (2003), Ventresca and Tizhoosh (2007), and Xinchao (2011).

- *generation mechanism*: A *generation mechanism* $\mathcal{G}$ describes how the simulated annealing algorithm generates a candidate solution from the current solution. It is a distribution-valued function defined on $\mathcal{S}$ mapping each configuration s to a distribution $\mathcal{G}(s)$ on $\mathcal{N}(s)$. We denote the probability that generating $s' \in \mathcal{N}(s)$ from s at temperature T according to the generation mechanism $\mathcal{G}$ by $G_T(s, s')$,

$$G_T(s, s') \coloneqq P_T(\text{generate } s' | \text{the current solution is } s).$$

A common choice of generation mechanism is for any configuration $s \in \mathcal{S}$, uniformly randomly generate a candidate solution s' from its neighborhood $\mathcal{N}(s)$, i.e. for all temperature T,

$$G_T(s, s') = \begin{cases} \frac{1}{|\mathcal{N}(s)|} & \text{if } s' \in \mathcal{N}(s) \\ 0 & \text{if } s' \notin \mathcal{N}(s). \end{cases} \tag{4.5}$$

Nevertheless, other generation mechanisms can be employed as long as the corresponding generation probability functions satisfy the following two conditions (Faigle & Kern, 1991):

(G1) For all $T \geq 0$, for any $s, s' \in \mathcal{S}$, $G_T(s, s') > 0$ if and only if $G_T(s', s) > 0$.

(G2) For all $T \geq 0$, for any $s, s' \in \mathcal{S}$, there exists $l \in \mathbb{Z}^+$ and $s_1, s_2, ..., s_l \in \mathcal{S}$, such

that

$$G_T(s, s_1)G_T(s_1, s_2)...G_T(s_l, s') > 0.$$

See Fox (1993) and Xinchao (2011) for more generation mechanisms.

- *acceptance criterion*: A *acceptance criterion* is a criterion to determine whether to accept a generated candidate solution as the current solution for the next iteration. The associated *acceptance probability function* of accepting the candidate solution s' at temperature T is denoted by $A_T(s, s')$,

$$A_T(s, s') \coloneqq P_T(\text{accept } s' | \text{the current solution is } s).$$

The acceptance probability function commonly needs to satisfy the following conditions:

(A1) $\forall T > 0$, $\forall s, s' \in \mathcal{S}$,

$$\begin{cases} A_T(s, s') \in (0, 1] & \text{if } f(s') > f(s) \\ A_T(s, s') = 1 & \text{if } f(s') \leq f(s), \end{cases}$$

(A2) $\forall s, s' \in \mathcal{S}$ with $f(s) < f(s')$,

$$\lim_{T \downarrow 0} A_T(s, s') = 0,$$

(A3) (Multiplicativity) $\forall T > 0$, $\forall s, s', s'' \in \mathcal{S}$ with $f(s) \leq f(s') \leq f(s'')$,

$$A_T(s, s')A_T(s', s'') = A_T(s, s'').$$

When the temperature $T > 0$ stays invariant during the annealing process, it's easy to check that the condition (A1) along with conditions (G1)-(G2) guarantees that the algorithm behaves as an ergodic homogeneous Markov chain with transition

probability

$$P_T(s, s') = \begin{cases} G_T(s, s')A_T(s, s') & \text{if } s \neq s' \\ 1 - \sum_{s'' \in \mathcal{N}(s)} G_T(s, s')A_T(s, s') & \text{if } s = s'. \end{cases} \tag{4.6}$$

The ergodicity ensures that the sequence of solutions of the algorithm will converge in distribution to the unique *stationary distribution* π_T of the Markov chain (Levin & Peres, 2017). When adopting the generation mechanism in (4.5), the stationary distribution corresponding to the Metropolis criterion (4.3) is just the Boltzmann distribution mentioned in (4.4). The conditions (A2)-(A3) further guarantee that the equilibrium distribution π_T converges in distribution to a uniform distribution defined on the set of all optimal solutions $\mathcal{S}_{opt} \coloneqq \{s : s \text{ is an optimal solution}\}$:

$$\lim_{T \downarrow 0} \pi_T(s) = \begin{cases} \frac{1}{|\mathcal{S}_{opt}|} & \text{if } s \in \mathcal{S}_{opt} \\ 0 & \text{if } s \notin \mathcal{S}_{opt} \end{cases}$$

Beyond the widely employed Metropolis criterion (4.3), alternative acceptance criteria exist that could enhance the finite-time performance of the simulated annealing algorithm for specific categories of combinatorial optimization problems. (Dueck & Scheuer, 1990; Moscato & Fontanari, 1990; Penna, 1995). See Anily and Federgruen (1987), Ogbu and Smith (1990), Schuur (1997), and Franz, Hoffmann, and Salamon (2001) for more discussion about acceptance criterion.

- *cooling schedule*: A *cooling schedule* involves three components: an *initial temperature* T_0, a schedule for temperature decrement, and a *stopping criterion*. We first talk about how to select the *initial temperature* T_0 and then discuss other components.

  The algorithm becomes inefficient when the initial temperature is either too high or too low. One way to experimentally find an appropriate initial temperature T_0 involves running the algorithm at a fixed temperature T and calculate the *acceptance*

ratio $\chi(T)$ associated with T, where

$$\chi(T) = \frac{\textit{number of accepted transitions from current solution to candidate solution}}{\textit{total number of iterations}}.$$

If this ratio is larger than a pre-specified threshold χ_0, adjust T by multiplying it by a factor greater than one and recalculate the acceptance ratio at this adjusted temperature. By iterating this process until the acceptance ratio is larger than χ_0, the concluding temperature can be utilized as the initial solution (Kirkpatrick et al., 1983). See Ben-Ameur (2004) for a detailed discussion of initial temperature selection.

A theoretically important cooling schedule is the *logarithmic cooling schedule*:

$$T_k = \frac{d}{\log(k+1)},\ k \geq 1,$$

where $d > 0$ is a constant, and k represents the iteration number. Hajek (1988) proved that, with a neighborhood structure satisfying (N1)-(N2) as well as having a constant neighborhood size $|\mathcal{N}(s)|$, the uniform generation mechanism (4.5), and the Metropolis acceptance criterion (4.3), the simulated annealing algorithm converges to a global optimal solution with probability 1 if and only if d equals a problem-dependent constant d^*. However, this cooling schedule is excessively slow, so in practice, other cooling schedules that don't have a theoretical guarantee for convergence but are much more efficient should be applied.

One commonly used cooling schedule in practice is the *geometric cooling schedule* (Strenski & Kirkpatrick, 1991): Starting from an initial temperature T_0, iteratively reduce the temperature by multiplying it by a factor $\alpha \in (0, 1)$, i.e. $T_k = \alpha^k T_0$, until it is close enough to 0. Another widely adopted cooling schedule is the *linear cooling schedule* (Kirkpatrick et al., 1983), which appoints a total number of temperature changes n, and T_k is defined by $T_k = (1 - \frac{k}{n})T_0, k = 1, 2, ..., n$. In addition to these strategies, researchers have proposed various cooling schedules, including adaptive

ones that heuristically determine T_k based on information gathered during the algorithm's execution. For example, Blum, Dan, and Seddighin (2021) designed a procedure that learns a problem-dependent finite sequence of temperatures $\{T_k\}_{k=1}^{n}$ by sampling instantiations from a family of similar combinatorial optimization problems. They demonstrated that, under certain conditions, this method exhibits remarkable efficiency regarding the likelihood of finding a global optimal solution in a limited amount of iterations. For more discussion about cooling schedules, see (Nourani & Andresen, 1998; Henderson et al., 2003; Karabin & Stuart, 2020).

- *Markov chain length*: At each temperature T_k, the algorithm usually performs several iterations until it ideally converges to the unique stationary distribution of the corresponding Markov chain. We label the number of iterations at temperature T_k as L_k, referring to it as the *Markov chain length*. When utilizing the uniform generation mechanism (4.5), Sanvicente-Sánchez and Frausto-Solís (2004) suggested that the Markov chain length L_k should be a multiple of the cardinality $|\mathcal{N}(s)|$ of the neighborhood of the current solution s. This relationship can be formulated as $L_k = c|\mathcal{N}(s)|$, with c recommended to lie between $[1, 4.6]$. Frausto-Solís, Sanvicente-Sánchez, and Imperial-Valenzuela (2006) proposed that, when using the geometric cooling schedule, exponentially increasing the length of the Markov chain, i.e. $L_k = \beta L_{k-1}$ with β being a constant slightly greater than 1, may improve the algorithm's efficiency. For more information, see (Henderson et al., 2003).

4.2 Quantum Annealing

In this section, we will discuss the quantum annealing algorithm, incorporating some notations and expressions from (Wang & Liu, 2022; Wang et al., 2023) in which I am one of the authors. Quantum annealing is the quantum analog of classical annealing, with thermodynamics replaced by quantum dynamics. Similar to the simulated annealing

algorithm, it is used to solve combinatorial optimization problems in the form of

$$\min_{s \in \mathcal{S}} \quad f(s).$$

This combinatorial optimization problem can be embedded in a specially designed quantum system. The basis states of this quantum system have a one-to-one relationship with the solutions of the combinatorial optimization problem, and the system's total energy at each basis state matches the value of the objective function at the corresponding solution. Therefore, finding an optimal solution to the original combinatorial optimization problem can be accomplished by finding a basis state, known as *ground state*, at which the quantum system has the lowest energy among all possible energies (Kadowaki & Nishimori, 1998; Rajak et al., 2023; Wang & Liu, 2022). The total energy of this quantum system is characterized by an operator called the Hamiltonian, denoted by $\mathbf{H}_q$. The Hamiltonian can be represented by a Hermitian matrix, which is a matrix whose conjugate transpose is equal to itself. The eigenstates of the Hamiltonian constitute the basis of the quantum system, with each eigenvalue reflecting the system's total energy in the respective eigenstate (Morita & Nishimori, 2008; Griffiths & Schroeter, 2019).

To find the ground state of $\mathbf{H}_q$, the (commonly referred) quantum annealing algorithm introduces an auxiliary Hamiltonian $\mathbf{H}_a$, where $\mathbf{H}_a$ is selected to be non-commutable with $\mathbf{H}_q$, i.e., $\mathbf{H}_q\mathbf{H}_a \neq \mathbf{H}_a\mathbf{H}_q$, and its ground state should be easy to prepare in practice (Kadowaki & Nishimori, 1998; Hauke et al., 2020). The quantum annealing system is initialized in the ground state of $\mathbf{H}_a$, and the Hamiltonian $\mathbf{H}_{QA}$ of the system is then gradually transitioned from the initial Hamiltonian $\mathbf{H}_a$ to the target Hamiltonian $\mathbf{H}_q$, which can be formulated as

$$\mathbf{H}_{QA}(t) = A(t)\mathbf{H}_q + B(t)\mathbf{H}_a, \ t \in [0, t_f], \tag{4.7}$$

where t_f is the duration of the annealing process, and the annealing schedules $A(t)$ and $B(t)$ are smooth functions such that $A(t)$ is increasing with $A(0) = 0, A(1) = 1$, and $B(t)$

is decreasing with $B(0) = 1, B(1) = 0$ (Hauke et al., 2020; Wang et al., 2023). As a spontaneous physical process, the state of the quantum annealing system automatically evolves, where the dynamic of the evolution is governed by a fundamental differential equation in quantum mechanics referred to as the Schrödinger equation, formulated as

$$i\hbar\frac{\partial}{\partial t}|\psi(t)\rangle = \mathbf{H}_{QA}(t)|\psi(t)\rangle,$$

where $|\psi(t)\rangle$ represents the state of the system at time t, and $\hbar = \frac{6.62607015\times10^{-34}}{2\pi}$ is the reduced Planck's constant (Schrödinger, 1926; Griffiths & Schroeter, 2019). Recall that the quantum system is initialized in the ground state $|\psi(0)\rangle$ of $\mathbf{H}_{QA}(0) = \mathbf{H}_a$. The adiabatic theorem, first established by Born and Fock (1928), guarantees that if the Hamiltonian $\mathbf{H}_{QA}(t)$ is not degenerate and the change of $\mathbf{H}_{QA}(t)$ is sufficiently slow, the final state $|\psi(t_f)\rangle$ of the quantum system will be nearly identical to the ground state of $\mathbf{H}_{QA}(t_f) = \mathbf{H}_q$ (Morita & Nishimori, 2008; Albash & Lidar, 2018). Consequently, with high probability, measuring the quantum system at the end of the process will yield an optimal solution to the combinatorial optimization problem (Jansen, Ruskai, & Seiler, 2007).

In contrast to many quantum algorithms, such as Shor's algorithm and the quantum phase estimation algorithm, quantum annealing does not rely on quantum gates. As a result, it can be effectively implemented on devices specifically constructed for this method (Johnson et al., 2011; Yarkoni, Raponi, Bäck, & Schmitt, 2022). This kind of special-purpose quantum device is called a quantum annealer. The large quantum annealing devices available today are designed and manufactured by the D-Wave company. Its Advantage$^{\text{TM}}$ quantum annealer, which was released in 2020, has at least five thousand superconducting qubits, and the next-generation device under development is expected to have over seven thousand qubits (Tasseff et al., 2022). Since D-Wave launched its first commercial quantum annealer in 2011, researchers have conducted extensive studies on these quantum annealers. Most studies suggest that D-Wave machines have not achieved quantum supremacy over state-of-the-art classical heuristic algorithms, but some research results indicate that the new generation device can outperform the simulated annealing

algorithm in specially designed benchmark problems tailored to the architecture of the quantum annealers (Boixo et al., 2014; Lanting et al., 2014; Albash, Vinci, Mishra, Warburton, & Lidar, 2015; Adame & McMahon, 2020; King et al., 2022; Willsch et al., 2022; Tasseff et al., 2022).

Due to technical restrictions, current quantum annealers can only handle special categories of combinatorial optimization problems that can be formulated in terms of classical Ising models (Kadowaki & Nishimori, 1998; Dattani, Szalay, & Chancellor, 2019; Hauke et al., 2020), where many practically important optimization problems, such as the travelling salesman problem and the Boolean satisfiability problem, can be represented as these models (Martoňák, Santoro, & Tosatti, 2004; Battaglia, Santoro, & Tosatti, 2005; Su, Tu, & He, 2016; Yarkoni et al., 2022). We describe the Ising model using a graph $\mathcal{G}$ whose site and edge sets are denoted by $\mathcal{V}$ and $\mathcal{E}$, respectively. Each site is associated with a value in $\{+1, -1\}$, and each edge specifies the interaction (or coupling) between the two sites connected by the edge. Denote by d the total number of sites in $\mathcal{G}$, and denote a configuration or state of the model by $s = (s_1, s_2, ..., s_d) \in \{-1, 1\}^d$. The classical Ising model has the following energy function (Wang & Liu, 2022; Wang et al., 2023):

$$\mathbf{H}_I^c(s) = -\sum_{\langle i,j\rangle \in \mathcal{E}} J_{ij} s_i s_j - \sum_{i \in \mathcal{V}} h_i s_i. \tag{4.8}$$

We refer to a set of fixed values $\{J_{ij}, h_i\}$ as one instance of the Ising model. The optimization problem associated with the Ising model involves finding a state that corresponds to the lowest energy of $\mathbf{H}_I^c(s)$.

The quantum Hamiltonian associated with the classical Ising model, i.e. the $\mathbf{H}_q$ term in Equation (4.7) is as follows:

$$\mathbf{H}_I^q = -\sum_{\langle i,j\rangle \in \mathcal{E}} J_{ij} \boldsymbol{\sigma}_i^z \boldsymbol{\sigma}_j^z - \sum_{i \in \mathcal{V}} h_i \boldsymbol{\sigma}_i^z, \tag{4.9}$$

where $\mathcal{G}$ is the graph specified in the definition of the classical Ising model with site set $\mathcal{V}$ and edge set $\mathcal{E}$, $\boldsymbol{\sigma}_i^z \boldsymbol{\sigma}_j^z$ denotes the tensor product of Pauli-z matrices along with identity

matrices in such a way that

$$\boldsymbol{\sigma}_i^z \boldsymbol{\sigma}_j^z = I_1 \otimes \cdots \otimes I_{i-1} \otimes \sigma_i^z \otimes I_{i+1} \otimes \cdots \otimes I_{j-1} \otimes \sigma_j^z \otimes I_{j+1} \otimes \cdots \otimes I_d,$$

and $\boldsymbol{\sigma}_i^z$ stands for the following tensor product of matrices:

$$\boldsymbol{\sigma}_i^z = I_1 \otimes \cdots \otimes I_{i-1} \otimes \sigma_i^z \otimes I_{i+1} \otimes \cdots \otimes I_d,$$

Each interaction strength coefficient J_{ij} is implemented in D-wave devices by a tunable coupler between the qubit on site i and the qubit on site j, and h_i is implemented by applying external flux to qubit on site $i \in \mathcal{V}$ (Boixo et al., 2014).

The commonly referred quantum annealing model is the transverse-field Ising model (Kadowaki & Nishimori, 1998; Wang et al., 2023):

$$\mathbf{H}_{QA}(t) = A(t)\mathbf{H}_I^q + B(t)\mathbf{H}_X, \ t \in [0, t_f], \tag{4.10}$$

where the transverse field Hamiltonian $\mathbf{H}_X$ is as follows:

$$\mathbf{H}_X = -\sum_{i \in \mathcal{V}} \boldsymbol{\sigma}_i^x,$$

with

$$\boldsymbol{\sigma}_i^x = I_1 \otimes \cdots \otimes I_{i-1} \otimes \sigma_i^x \otimes I_{i+1} \otimes \cdots \otimes I_d,$$

where σ_i^x is the Pauli-x matrix. As described before, t_f is the duration of the annealing process, and the annealing schedules $A(t)$ and $B(t)$ are smooth functions such that $A(t)$ is increasing with $A(0) = 0, A(1) = 1$, and $B(t)$ is decreasing with $B(0) = 1, B(1) = 0$. The quantum annealing system is initialized to the ground state $(1, 1, ..., 1)^T$ of the transverse field Hamiltonian $\mathbf{H}_X$, and the Hamiltonian of the system is slowly driven from $\mathbf{H}_X$ to the final target Hamiltonian $\mathbf{H}_I^q$. Ideally, the quantum state of the system will end up with the ground state of $\mathbf{H}_I^q$. The eigenvalues of $\mathbf{H}_I^q$ are exactly all the 2^d configurations of the

classical Hamiltonian $\mathbf{H}_I^c(s)$ in (4.8). The relationship between the classical Hamiltonian $\mathbf{H}_I^c(s)$ and the quantum Hamiltonian $\mathbf{H}_I^q$ is described in the following theorem, whose proof can be found in (Wang et al., 2023):

Theorem 4.1. *The eigenvalues of the quantum Hamiltonian $\mathbf{H}_I^q$ in (4.9) are given by the 2^d values of the classical Hamiltonian $\mathbf{H}_I^c(s)$ in (4.8) evaluated at 2^d configurations $\mathbf{s} \in \{+1, -1\}^d$. In particular, the minimum of $\mathbf{H}_I^c(s)$ over $\mathbf{s} \in \{+1, -1\}^d$ is equal to the smallest eigenvalue of $\mathbf{H}_I^q$.*

Theorem 4.1 shows that finding the minimal energy of the classical Ising model described by $\mathbf{H}_I^c(s)$ is mathematically identical to finding the minimal energy of the quantum Hamiltonian $\mathbf{H}_I^q(s)$. Thus, at the end of the quantum annealing process, measuring the quantum system renders a solution to the combinatorial minimization problem with the objective function $\mathbf{H}_I^c(s)$. Like the simulated annealing algorithm, each run of quantum annealing can produce a solution to the optimization problem that is optimal with some probability, and running quantum annealing many times enables us to solve the optimization problem.

Several techniques were proposed to enhance the performance of the quantum annealing algorithm, such as properly designing the annealing schedules $A(t)$ and $B(t)$ as well as replacing the simple transverse field Hamiltonian $\mathbf{H}_X$ by more complicated Hamiltonians (Perdomo-Ortiz, Venegas-Andraca, & Aspuru-Guzik, 2011; Chancellor, 2017; Nishimori & Takada, 2017; Susa, Yamashiro, Yamamoto, & Nishimori, 2018; Susa, Yamashiro, Yamamoto, Hen, et al., 2018; Hauke et al., 2020). For example, Adame and McMahon (2020) proposed a time-shift-based protocol for inhomogeneous driving where the quantum annealing Hamiltonian is given by

$$\mathbf{H}_{QA}(t) = -\sum_{\langle i,j\rangle\in\mathcal{E}} \sqrt{A_i(t)A_j(t)}J_{ij}\boldsymbol{\sigma}_i^z\boldsymbol{\sigma}_j^z - \sum_{i\in\mathcal{V}} A_i(t)h_i\boldsymbol{\sigma}_i^z - \sum_{i\in\mathcal{V}} B_i(t)h_i\boldsymbol{\sigma}_i^x,$$

and the experimental implementation of this protocol in D-wave devices demonstrates a notable improvement in performance as compared to the simple transverse-field Ising

model (4.10).

In the next section, we will introduce some theoretical results regarding the probability of successfully finding the ground state via the quantum annealing algorithm.

4.3 Bounds on Quantum Annealing

A theoretical base of quantum annealing is the adiabatic theorem. Let $u = \frac{t}{t_f}, t \in [0, t_f]$. Denote the state of the quantum annealing system at time $t = ut_f$ by $|\psi(u)\rangle$ and the instantaneous ground state of $\mathbf{H}_{QA}(ut_f)$ by $|\lambda_0(u)\rangle$. Assume that $|\lambda_0(u)\rangle$ is non-degenerate over $u \in [0, 1]$, denote the *energy gap* of the system by $\Delta\lambda(u) = \lambda_1(u) - \lambda_0(u)$, where $\lambda_1(u)$ is the energy of the instantaneous first excited state of the quantum system at time ut_f, and define $\Delta_{\min}\lambda = \min_{u\in[0,1]} \Delta\lambda(u)$ as the *minimum energy gap*, the adiabatic theorem says the quantum annealing will yield the ground state of $\mathbf{H}_q^I$ with sufficiently high probability if (Born & Fock, 1928; McGeoch, 2014)

$$t_f \gg \frac{\left|\left|\frac{dH(u)}{du}\right|\right|_{L_2}}{(\Delta_{\min}\lambda)^2}.$$

More convergence conditions for the quantum annealing algorithm can be found in (Morita & Nishimori, 2008; Cheung, Høyer, & Wiebe, 2011; Kimura & Nishimori, 2022). Lower bounds for the annealing time can be found in (García-Pintos, Brady, Bringewatt, & Liu, 2023).

The convergence conditions demonstrate the asymptotic behavior of the quantum annealing process. The following theorem, which characterizes the finite-time behavior of the algorithm, provides a probability bound on successfully solving the optimization problem at the final annealing time t_f using quantum annealing, whose proof can be found in (Wang et al., 2023):

Theorem 4.2. *Suppose that the quantum system associated with quantum annealing is driven by $\mathbf{H}_{QA}(t)$ as defined in (4.10). Then the probability that the lowest energy of $\mathbf{H}_I^c$ in (4.8) is obtained by measuring the system at the end of quantum annealing is bounded*

from below by

$$\max\left\{\left[\left(1-\int_0^1\left\|\frac{d}{du}(|v_1(u)\rangle,...,|v_r(u)\rangle)\right\|du\right)_+\right]^2,\{v_j\}_{1\le j\le r},1\le r\le\zeta\right\},\qquad(4.11)$$

where ζ *denotes the number of ground states for the quantum Hamiltonian* $\mathbf{H}_I^q$ *in (4.9),* $(x)_+$ *denotes the positive part of* x*—namely,* $(x)_+$ *is equal to* x *if* $x\ge 0$ *and zero otherwise, and* $(|v_1(u)\rangle,...,|v_r(u)\rangle)$ *is a matrix formed by* r *column vectors* $|v_1(u)\rangle,...,|v_r(u)\rangle$ *that are defined as follows. Denote by* $\xi_1(u)\le\xi_2(u)\le...\le\xi_{2^d}(u)$ *the* 2^d *instantaneous eigenvalues of* $\mathbf{H}_{QA}(ut_f)$ *listed in an increasing order along with the corresponding* 2^d *normalized eigenvectors* $v_1(u),v_2(u),...,v_{2^d}(u)$*, where for any eigenvalue with a multiplicity greater than 1, the eigenvalue is repeated in the list with the number of repetitions equal to its multiplicity, and the multiple eigenvectors corresponding to the same eigenvalue are ordered as a group—that is, their positions in the list are exchangeable, and the maximum in (4.11) is taken over* $1\le r\le\zeta$ *and possible group orderings of* $v_j(u)$*.*

Furthermore, assume that the energy gap $\Delta\lambda(u)$ *is bounded from zero uniformly over* $u\in[0,1]$*. Then the probability that the quantum annealing procedure can find the lowest energy of* $\mathbf{H}_I^c$ *in (4.8) is bounded from below by*

$$1-2^d\zeta\max_{u\in[0,1]}\left\{\frac{1}{\Delta\lambda(u)}\left\|\frac{d\mathbf{H}_{QA}(ut_f)}{du}\right\|\right\}^2,\qquad(4.12)$$

where $\|\cdot\|$ *denotes the matrix spectral norm.*

Theorem 4.2 indicates that by choosing appropriate $A(t)$ and $B(t)$, we can ensure that the probability lower bound in (4.12) is positive and thus guarantee that quantum annealing can find the lowest energy of $\mathbf{H}_I^c$ with some probability. Indeed, let

$$\aleph=\max_{u\in[0,1]}\left\{\frac{1}{\Delta\lambda(u)}\left\|\frac{d\mathbf{H}_{\mathrm{QA}}(ut_f)}{du}\right\|\right\}^2,$$

note that

$$\frac{d\mathbf{H}_{\mathrm{QA}}(ut_f)}{du}=\frac{dA(ut_f)}{du}\mathbf{H}_I^q+\frac{dB(ut_f)}{du}\mathbf{H}_X,$$

which depends on u only through the derivatives of annealing schedules $A(t)$ and $B(t)$, we have

$$\begin{aligned}\aleph &\leq \max_{u\in[0,1]} \left\{ (\Delta\lambda(u))^{-2} \left[\left| \frac{dA(ut_f)}{du} \right| \|\mathbf{H}_I^q\| + \left| \frac{dB(ut_f)}{du} \right| \|\mathbf{H}_X\| \right]^2 \right\} \\ &\leq (\Delta_{\min}\lambda)^{-2} [\|\mathbf{H}_I^q\| + \|\mathbf{H}_X\|]^2 \max_{u\in[0,1]} \left\{ \left| \frac{dA(ut_f)}{du} \right| \bigvee \left| \frac{dB(ut_f)}{du} \right| \right\},\end{aligned}$$

where $\bigvee$ stands for the maximum. For a given quantum annealing setup, we have fixed $d, \zeta, \|\mathbf{H}_I^q\|, \|\mathbf{H}_X\|$, and $\Delta_{\min}\lambda$. Hence, if for $u \in [0,1]$,

$$\left| \frac{dA(ut_f)}{du} \right| \bigvee \left| \frac{dB(ut_f)}{du} \right| < \frac{\Delta_{\min}\lambda}{2^{d/2}\sqrt{\zeta}[\|\mathbf{H}_I^q\| + \|\mathbf{H}_X\|]},$$

then the probability lower bound

$$1 - 2^d \zeta \max_{u\in[0,1]} \left\{ \frac{1}{\Delta\lambda(u)} \left\| \frac{d\mathbf{H}_{\mathrm{QA}}(ut_f)}{du} \right\| \right\}^2$$

is greater than 0, which partially reflecting the finite-time behavior of the quantum annealing algorithm.

Appendix A

Appendix

A.1 Proof of Lemma 2.1

We now prove the Lemma 2.1. Since for any $\delta > 0$, there exists $\delta' > 0$ such that $\frac{\delta'}{m} = \frac{\delta}{m+3}$, we reformulate the lemma as follows:

Lemma A.1. *For any $\phi \in [0, 2\pi), \delta > 0$, $\exists s = O(\log(m))$, such that for all j,*

$$P(|\rho_j - (M_j\psi \mod 1)| > \frac{1}{16}) < \frac{\delta}{m}.$$

Proof. We first consider the case that $j = 1$. Let $c = \frac{\pi}{8}$, $P^*_{sin} = P^*_{sin}(M_1)$, and $P^*_{cos} = P^*_{cos}(M_1)$. When $\phi \in [0, \frac{\pi}{4})$,

$$\begin{aligned}
P(|\rho_j - (M_j\psi \mod 1)| > \frac{1}{16}) &= P(|\arctan(\frac{P^*_{sin}}{P^*_{cos}}) - \phi| > c) \\
&\leq P(|\frac{P^*_{sin}}{P^*_{cos}} - \tan(\phi)| > c) \\
&= P(|\frac{P^*_{sin}}{P^*_{cos}} - \frac{\sin(\phi)}{P^*_{cos}} + \frac{\sin(\phi)}{P^*_{cos}} - \frac{\sin(\phi)}{\cos(\phi)}| > c) \\
&\leq P(|\frac{P^*_{sin} - \sin(\phi)}{P^*_{cos}}| > \frac{c}{2} \cup |\frac{\sin(\phi)\cos(\phi) - P^*_{cos}\sin(\phi)}{P^*_{cos}\cos(\phi)}| > \frac{c}{2}) \\
&\leq P(|\frac{P^*_{sin} - \sin(\phi)}{P^*_{cos}}| > \frac{c}{2}) + P(|\frac{\sin(\phi)\cos(\phi) - P^*_{cos}\sin(\phi)}{P^*_{cos}\cos(\phi)}| > \frac{c}{2}),
\end{aligned}$$

$$
\begin{aligned}
P(|\frac{P^*_{sin}-\sin(\phi)}{P^*_{cos}}|>\frac{c}{2}) &= P(|\frac{P^*_{sin}-\sin(\phi)}{P^*_{cos}}|>\frac{c}{2}, |P^*_{cos}-\cos(\phi)|<\frac{1}{2}\cos(\phi)) \\
&\quad + P(|\frac{P^*_{sin}-\sin(\phi)}{P^*_{cos}}|>\frac{c}{2}, |P^*_{cos}-\cos(\phi)|\geq\frac{1}{2}\cos(\phi)) \\
&\leq P(|\frac{P^*_{sin}-\sin(\phi)}{\frac{1}{2}\cos(\phi)}|>\frac{c}{2}) + P(|P^*_{cos}-\cos(\phi)|\geq\frac{1}{2}\cos(\phi)) \\
&= P(|P^*_{sin}-\sin(\phi)|>\frac{c}{4}\cos(\phi)) + P(|P^*_{cos}-\cos(\phi)|\geq\frac{1}{2}\cos(\phi)),
\end{aligned}
$$

so by Hoeffding's inequality,

$$
\begin{aligned}
P(|\frac{P^*_{sin}-\sin(\phi)}{P^*_{cos}}|>\frac{c}{2}) &\leq P(|P^*_{sin}-\sin(\phi)|>\frac{c}{4}\cos(\phi)) + P(|P^*_{cos}-\cos(\phi)|\geq\frac{1}{2}\cos(\phi)) \\
&\leq 2e^{-2(\frac{c}{4}\cos(\phi))^2 s} + 2e^{-2(\frac{1}{2}\cos(\phi))^2 s} \\
&= 2e^{-\frac{c^2}{8}\cos^2(\phi)s} + 2e^{-\frac{1}{2}\cos^2(\phi)s} \\
&\leq 2e^{-\frac{c^2}{16}s} + 2e^{-\frac{1}{4}s},
\end{aligned}
$$

where the last inequality is from the assumption that $\phi \in [0, \frac{\pi}{4})$.

Similarly,

$$
\begin{aligned}
P(|\frac{\sin(\phi)\cos(\phi)-P^*_{cos}\sin(\phi)}{P^*_{cos}\cos(\phi)}|>\frac{c}{2}) &\leq P(|\frac{\sin(\phi)(P^*_{cos}-\cos(\phi))}{\frac{1}{2}\cos^2(\phi)}|>\frac{c}{2}) \\
&\quad + P(|P^*_{cos}-\cos(\phi)|\geq\frac{1}{2}\cos(\phi)) \\
&= P(|P^*_{cos}-\cos(\phi)|>\frac{c}{4}\frac{\cos^2(\phi)}{\sin(\phi)}) \\
&\quad + P(|P^*_{cos}-\cos(\phi)|\geq\frac{1}{2}\cos(\phi)) \\
&\leq 2e^{-2(\frac{c}{4}\frac{\cos^2(\phi)}{\sin(\phi)}))^2 s} + 2e^{-2(\frac{1}{2}\cos(\phi))^2 s} \\
&= 2e^{-\frac{c^2}{8}\cos^2(\phi)\cot^2(\phi)s} + 2e^{-\frac{1}{2}\cos^2(\phi)s} \\
&\leq 2e^{-\frac{c^2}{16}s} + 2e^{-\frac{1}{4}s}.
\end{aligned}
$$

Combining the bounds above together, we have

$$\begin{aligned} P(|\rho_j - (M_j\psi \mod 1)| > \frac{1}{16}) &= P(|\arctan(\frac{P^*_{sin}}{P^*_{cos}}) - \phi| > c) \\ &\leq 4e^{-\frac{c^2}{16}s} + 4e^{-\frac{1}{4}s} \\ &\leq 8e^{-\frac{c^2}{16}s}. \end{aligned}$$

Therefore, when

$$s = \frac{16}{c^2}\log(\frac{8m}{\delta}) = \frac{1024}{\pi^2}\log(\frac{8m}{\delta}) = O(\log(m)),$$

$P(|\rho_j - (M_j\psi \mod 1)| > \frac{1}{16}) < \frac{\delta}{m}$ holds.

The case that $\phi \in [\frac{\pi}{4}, \frac{\pi}{2})$ can be handled by utilizing the equation $\arctan(x) = \frac{\pi}{2} - \arctan(\frac{1}{x})$:

$$\begin{aligned} P(|\rho_j - (M_j\psi \mod 1)| > \frac{1}{16}) &= P(|\arctan(\frac{P^*_{sin}}{P^*_{cos}}) - \phi| > c) \\ &= P(|(\frac{\pi}{2} - \arctan(\frac{P^*_{cos}}{P^*_{sin}}) - \phi| > c) \\ &= P(|\arctan(\frac{P^*_{cos}}{P^*_{sin}}) - (\frac{\pi}{2} - \phi)| > c) \\ &\leq P(|\frac{P^*_{cos}}{P^*_{sin}} - \cot(\phi)| > c). \end{aligned}$$

In the same manner, we can show that there exists $s = O(\log(m))$ such that $P(|\rho_j - (M_j\psi \mod 1)| > \frac{1}{16}) < \frac{\delta}{m}$ holds for any $\phi \in [0, 2\pi)$. The choice of s is independent of the phase ϕ, so the proof for $j = 1$ case can be immediately extended to $j > 1$ cases. Therefore, the statement in this lemma holds. □

A.2 Proof of Lemma 3.1

We now provide a proof for Lemma 3.1:

Lemma 3.1.

$$\int_{-\infty}^{\infty} \cos(\alpha\phi+\beta)e^{-\frac{1}{2}\phi^2}d\phi = \sqrt{2\pi}\cos(\beta)e^{-\frac{1}{2}\alpha^2}$$
$$\int_{-\infty}^{\infty} \sin(\alpha\phi+\beta)e^{-\frac{1}{2}\phi^2}d\phi = \sqrt{2\pi}\sin(\beta)e^{-\frac{1}{2}\alpha^2}.$$

Proof. Let $f(\phi) = e^{-\frac{1}{2}\phi^2}$, the derivative $f'(\phi)$ satisfies:

$$f'(\phi) = -\phi e^{-\frac{1}{2}\phi^2} = -\phi f(\phi).$$

Taking Fourier transform on both sides, we have

$$\int_{-\infty}^{\infty} e^{-i\omega\phi}f'(\phi)d\phi = -\int_{-\infty}^{\infty} \phi e^{-i\omega\phi}f(\phi)d\phi.$$

Since

$$\int_{-\infty}^{\infty} e^{-i\omega\phi}f'(\phi)d\phi = i\omega\int_{-\infty}^{\infty} e^{-i\omega\phi}f(\phi)d\phi$$

and

$$\int_{-\infty}^{\infty} \phi e^{-i\omega\phi}f(\phi)d\phi = i\frac{d}{d\omega}\int_{-\infty}^{\infty} e^{-i\omega\phi}f(\phi)d\phi,$$

we have

$$\omega\hat{f}(\omega) = -\hat{f}'(\omega),$$

where

$$\hat{f}(\omega) = \int_{-\infty}^{\infty} e^{-i\omega\phi}f(\phi)d\phi.$$

Solving this differential equation, we can get

$$\hat{f}(\omega) = Ce^{-\frac{1}{2}\omega^2},$$

where C is a constant. When $\omega = 0$,

$$\hat{f}(0) = C = \int_{-\infty}^{\infty} e^{-\frac{1}{2}\phi^2}d\phi = \sqrt{2\pi}.$$

Therefore,

$$\hat{f}(\omega) = \sqrt{2\pi} e^{-\frac{1}{2}\omega^2},$$

which follows that

$$\hat{f}(-\alpha) = \int_{-\infty}^{\infty} e^{i\alpha\phi} f(\phi) d\phi = \sqrt{2\pi} e^{-\frac{1}{2}\alpha^2},$$

and it implies

$$\begin{aligned}
\int_{-\infty}^{\infty} e^{i(\alpha\phi+\beta)} f(\phi) d\phi &= \int_{-\infty}^{\infty} \cos(\alpha\phi+\beta) e^{-\frac{1}{2}\phi^2} d\phi + i \int_{-\infty}^{\infty} \sin(\alpha\phi+\beta) e^{-\frac{1}{2}\phi^2} d\phi \\
&= \sqrt{2\pi} e^{-\frac{1}{2}\alpha^2} \cdot e^{i\beta} \\
&= \sqrt{2\pi} \cos(\beta) e^{-\frac{1}{2}\alpha^2} + i\sqrt{2\pi} \sin(\beta) e^{-\frac{1}{2}\alpha^2}.
\end{aligned}$$

By comparing the real parts and the imaginary parts, we have

$$\begin{aligned}
\int_{-\infty}^{\infty} \cos(\alpha\phi+\beta) e^{-\frac{1}{2}\phi^2} d\phi &= \sqrt{2\pi} \cos(\beta) e^{-\frac{1}{2}\alpha^2} \\
\int_{-\infty}^{\infty} \sin(\alpha\phi+\beta) e^{-\frac{1}{2}\phi^2} d\phi &= \sqrt{2\pi} \sin(\beta) e^{-\frac{1}{2}\alpha^2}.
\end{aligned}$$

□

References

Abrams, D. S., & Lloyd, S. (1999). Quantum algorithm providing exponential speed increase for finding eigenvalues and eigenvectors. *Physical Review Letters*, *83*(24), 5162.

Adame, J. I., & McMahon, P. L. (2020). Inhomogeneous driving in quantum annealers can result in orders-of-magnitude improvements in performance. *Quantum Science and Technology*, *5*(3), 035011.

Albash, T., & Lidar, D. A. (2018). Adiabatic quantum computation. *Reviews of Modern Physics*, *90*(1), 015002.

Albash, T., Vinci, W., Mishra, A., Warburton, P. A., & Lidar, D. A. (2015). Consistency tests of classical and quantum models for a quantum annealer. *Physical Review A*, *91*(4), 042314.

Anily, S., & Federgruen, A. (1987). Simulated annealing methods with general acceptance probabilities. *Journal of applied probability*, *24*(3), 657–667.

Battaglia, D. A., Santoro, G. E., & Tosatti, E. (2005). Optimization by quantum annealing: Lessons from hard satisfiability problems. *Physical Review E—Statistical, Nonlinear, and Soft Matter Physics*, *71*(6), 066707.

Ben-Ameur, W. (2004). Computing the initial temperature of simulated annealing. *Computational optimization and applications*, *29*, 369–385.

Blum, A., Dan, C., & Seddighin, S. (2021). Learning complexity of simulated annealing. In *International conference on artificial intelligence and statistics* (pp. 1540–1548).

Boixo, S., Rønnow, T. F., Isakov, S. V., Wang, Z., Wecker, D., Lidar, D. A., ... Troyer, M. (2014). Evidence for quantum annealing with more than one hundred qubits. *Nature physics*, *10*(3), 218–224.

Born, M., & Fock, V. (1928). Beweis des adiabatensatzes. *Zeitschrift für Physik*, *51*(3), 165–180.

Černỳ, V. (1985). Thermodynamical approach to the traveling salesman problem: An efficient simulation algorithm. *Journal of optimization theory and applications*, *45*, 41–51.

Chancellor, N. (2017). Modernizing quantum annealing using local searches. *New Journal of Physics*, *19*(2), 023024.

Cheung, D., Høyer, P., & Wiebe, N. (2011). Improved error bounds for the adiabatic approximation. *Journal of Physics A: Mathematical and Theoretical*, *44*(41), 415302.

Clarke, J., & Wilhelm, F. K. (2008). Superconducting quantum bits. *Nature*, *453*(7198), 1031–1042.

Connors, D. P., & Kumar, P. (1989). Simulated annealing type markov chains and their order balance equations. *SIAM Journal on Control and Optimization*, *27*(6), 1440–

1461.

Dattani, N., Szalay, S., & Chancellor, N. (2019). Pegasus: The second connectivity graph for large-scale quantum annealing hardware. *arXiv preprint arXiv:1901.07636*.

Dawson, C. M., & Nielsen, M. A. (2005). The solovay-kitaev algorithm. *arXiv preprint quant-ph/0505030*.

Delahaye, D., Chaimatanan, S., & Mongeau, M. (2019). Simulated annealing: From basics to applications. *Handbook of metaheuristics*, 1–35.

DiVincenzo, D. P. (1995). Two-bit gates are universal for quantum computation. *Physical Review A*, *51*(2), 1015.

Dueck, G., & Scheuer, T. (1990). Threshold accepting: A general purpose optimization algorithm appearing superior to simulated annealing. *Journal of computational physics*, *90*(1), 161–175.

Faigle, U., & Kern, W. (1991). Note on the convergence of simulated annealing algorithms. *SIAM journal on control and optimization*, *29*(1), 153–159.

Fleischer, M., & Jacobson, S. H. (1999). Information theory and the finite-time behavior of the simulated annealing algorithm: Experimental results. *INFORMS Journal on Computing*, *11*(1), 35-43. doi: 10.1287/ijoc.11.1.35

Fox, B. L. (1993). Integrating and accelerating tabu search, simulated annealing, and genetic algorithms. *Annals of Operations Research*, *41*(2), 47–67.

Franz, A., Hoffmann, K. H., & Salamon, P. (2001). Best possible strategy for finding ground states. *Physical Review Letters*, *86*(23), 5219.

Frausto-Solís, J., Sanvicente-Sánchez, H., & Imperial-Valenzuela, F. (2006). Andymark: an analytical method to establish dynamically the length of the markov chain in simulated annealing for the satisfiability problem. In *Simulated evolution and learning: 6th international conference, seal 2006, hefei, china, october 15-18, 2006. proceedings 6* (pp. 269–276).

García-Pintos, L. P., Brady, L. T., Bringewatt, J., & Liu, Y.-K. (2023). Lower bounds on quantum annealing times. *Physical Review Letters*, *130*(14), 140601.

Gebhart, V., Smerzi, A., & Pezzè, L. (2021). Bayesian quantum multiphase estimation algorithm. *Physical Review Applied*, *16*(1), 014035.

Griffiths, D. J., & Schroeter, D. F. (2019). *Introduction to quantum mechanics*. Cambridge university press.

Grover, L. K. (1996). A fast quantum mechanical algorithm for database search. In *Proceedings of the twenty-eighth annual acm symposium on theory of computing* (pp. 212–219).

Hajek, B. (1988). Cooling schedules for optimal annealing. *Mathematics of operations research*, *13*(2), 311–329.

Harrow, A. W., Recht, B., & Chuang, I. L. (2002). Efficient discrete approximations of quantum gates. *Journal of Mathematical Physics*, *43*(9), 4445–4451.

Hauke, P., Katzgraber, H. G., Lechner, W., Nishimori, H., & Oliver, W. D. (2020). Perspectives of quantum annealing: Methods and implementations. *Reports on Progress in Physics*, *83*(5), 054401.

Henderson, D., Jacobson, S. H., & Johnson, A. W. (2003). The theory and practice of simulated annealing. In F. Glover & G. A. Kochenberger (Eds.), *Handbook of metaheuristics* (pp. 287–319). Springer US. doi: 10.1007/0-306-48056-5_10

Horodecki, R., Horodecki, P., Horodecki, M., & Horodecki, K. (2009). Quantum entanglement. *Reviews of modern physics*, *81*(2), 865–942.

Jansen, S., Ruskai, M.-B., & Seiler, R. (2007). Bounds for the adiabatic approximation with applications to quantum computation. *Journal of Mathematical Physics*, *48*(10).

Johnson, M. W., Amin, M. H., Gildert, S., Lanting, T., Hamze, F., Dickson, N., ... others (2011). Quantum annealing with manufactured spins. *Nature*, *473*(7346), 194–198.

Kadowaki, T., & Nishimori, H. (1998). Quantum annealing in the transverse ising model. *Physical Review E*, *58*(5), 5355.

Kaftal, T., & Demkowicz-Dobrzański, R. (2014). Usefulness of an enhanced kitaev phase-estimation algorithm in quantum metrology and computation. *Physical Review A*, *90*(6), 062313.

Karabin, M., & Stuart, S. J. (2020). Simulated annealing with adaptive cooling rates. *The Journal of Chemical Physics*, *153*(11).

Kim, Y., Eddins, A., Anand, S., Wei, K. X., Van Den Berg, E., Rosenblatt, S., ... others (2023). Evidence for the utility of quantum computing before fault tolerance. *Nature*, *618*(7965), 500–505.

Kimura, Y., & Nishimori, H. (2022). Rigorous convergence condition for quantum annealing. *Journal of Physics A: Mathematical and Theoretical*, *55*(43), 435302.

King, A. D., Suzuki, S., Raymond, J., Zucca, A., Lanting, T., Altomare, F., ... others (2022). Coherent quantum annealing in a programmable 2,000 qubit ising chain. *Nature Physics*, *18*(11), 1324–1328.

Kirkpatrick, S., Gelatt Jr, C. D., & Vecchi, M. P. (1983). Optimization by simulated annealing. *science*, *220*(4598), 671–680.

Kitaev, A. Y. (1995). Quantum measurements and the abelian stabilizer problem. *arXiv preprint quant-ph/9511026*.

Kitaev, A. Y. (1997). Quantum computations: algorithms and error correction. *Russian Mathematical Surveys*, *52*(6), 1191.

Kitaev, A. Y., Shen, A., & Vyalyi, M. N. (2002). *Classical and quantum computation* (No. 47). American Mathematical Soc.

Laarhoven, P. J. M., & Aarts, E. H. L. (1987). *Simulated annealing: Theory and applications*. Springer. doi: 10.1007/978-94-015-7744-1

Lanting, T., Przybysz, A. J., Smirnov, A. Y., Spedalieri, F. M., Amin, M. H., Berkley, A. J., ... others (2014). Entanglement in a quantum annealing processor. *Physical Review X*, *4*(2), 021041.

Levin, D. A., & Peres, Y. (2017). *Markov chains and mixing times* (Vol. 107). American Mathematical Soc.

Li, Y., Pezzè, L., Gessner, M., Ren, Z., Li, W., & Smerzi, A. (2018). Frequentist and bayesian quantum phase estimation. *Entropy*, *20*(9), 628.

Lu, F. (2021). Several ways to implement qubits in physics. In *Journal of physics: Conference series* (Vol. 1865, p. 022007).

Marinescu, D. C. (2011). *Classical and quantum information*. Academic Press.

Martínez-García, F., Vodola, D., & Müller, M. (2019). Adaptive bayesian phase estimation for quantum error correcting codes. *New Journal of Physics*, *21*(12), 123027.

Martoňák, R., Santoro, G. E., & Tosatti, E. (2004). Quantum annealing of the traveling-salesman problem. *Physical Review E—Statistical, Nonlinear, and Soft Matter*

Physics, *70*(5), 057701.

McGeoch, C. C. (2014). *Adiabatic quantum computation and quantum annealing: Theory and practice*. Morgan & Claypool Publishers.

Metropolis, N., Rosenbluth, A. W., Rosenbluth, M. N., Teller, A. H., & Teller, E. (1953). Equation of state calculations by fast computing machines. *The journal of chemical physics*, *21*(6), 1087–1092.

Mohammadbagherpoor, H., Oh, Y.-H., Dreher, P., Singh, A., Yu, X., & Rindos, A. J. (2019). An improved implementation approach for quantum phase estimation on quantum computers. In *2019 ieee international conference on rebooting computing (icrc)* (pp. 1–9).

Montanaro, A. (2016). Quantum algorithms: an overview. *npj Quantum Information*, *2*(1), 1–8.

Morita, S., & Nishimori, H. (2008). Mathematical foundation of quantum annealing. *Journal of Mathematical Physics*, *49*(12).

Moscato, P., & Fontanari, J. F. (1990). Stochastic versus deterministic update in simulated annealing. *Physics Letters A*, *146*(4), 204–208.

Möttönen, M., & Vartiainen, J. J. (2006). Decompositions of general quantum gates. *Trends in quantum computing research*, 149.

Musk, D. R. (2020). A comparison of quantum and traditional fourier transform computations. *Computing in Science & Engineering*, *22*(6), 103–110.

Nielsen, M. A., & Chuang, I. L. (2010). *Quantum computation and quantum information*. Cambridge university press.

Nishimori, H., & Takada, K. (2017). Exponential enhancement of the efficiency of quantum annealing by non-stoquastic hamiltonians. *Frontiers in ICT*, *4*, 2.

Nourani, Y., & Andresen, B. (1998). A comparison of simulated annealing cooling strategies. *Journal of Physics A: Mathematical and General*, *31*(41), 8373.

Ogbu, F., & Smith, D. K. (1990). The application of the simulated annealing algorithm to the solution of the n/m/cmax flowshop problem. *Computers & Operations Research*, *17*(3), 243–253.

Ozols, M., Roetteler, M., & Roland, J. (2013). Quantum rejection sampling. *ACM Transactions on Computation Theory (TOCT)*, *5*(3), 1–33.

Paesani, S., Gentile, A. A., Santagati, R., Wang, J., Wiebe, N., Tew, D. P., ... Thompson, M. G. (2017). Experimental bayesian quantum phase estimation on a silicon photonic chip. *Physical review letters*, *118*(10), 100503.

Penna, T. J. (1995). Traveling salesman problem and tsallis statistics. *Physical Review E*, *51*(1), R1.

Perdomo-Ortiz, A., Venegas-Andraca, S. E., & Aspuru-Guzik, A. (2011). A study of heuristic guesses for adiabatic quantum computation. *Quantum Information Processing*, *10*, 33–52.

Qiang, X., Zhou, X., Wang, J., Wilkes, C. M., Loke, T., O'Gara, S., ... others (2018). Large-scale silicon quantum photonics implementing arbitrary two-qubit processing. *Nature photonics*, *12*(9), 534–539.

Rajak, A., Suzuki, S., Dutta, A., & Chakrabarti, B. K. (2023). Quantum annealing: An overview. *Philosophical Transactions of the Royal Society A*, *381*(2241), 20210417.

Rieffel, E., & Polak, W. (2000). An introduction to quantum computing for non-physicists. *ACM Computing Surveys (CSUR)*, *32*(3), 300–335.

Sanvicente-Sánchez, H., & Frausto-Solís, J. (2004). A method to establish the cooling scheme in simulated annealing like algorithms. In *Computational science and its applications–iccsa 2004: International conference, assisi, italy, may 14-17, 2004, proceedings, part iii 4* (pp. 755–763).

Schrödinger, E. (1926). An undulatory theory of the mechanics of atoms and molecules. *Physical review*, *28*(6), 1049.

Schuur, P. C. (1997). Classification of acceptance criteria for the simulated annealing algorithm. *Mathematics of Operations Research*, *22*(2), 266–275.

Shor, P. W. (1994). Algorithms for quantum computation: discrete logarithms and factoring. In *Proceedings 35th annual symposium on foundations of computer science* (pp. 124–134).

Strenski, P. N., & Kirkpatrick, S. (1991). Analysis of finite length annealing schedules. *Algorithmica*, *6*, 346–366.

Su, J., Tu, T., & He, L. (2016). A quantum annealing approach for boolean satisfiability problem. In *Proceedings of the 53rd annual design automation conference* (pp. 1–6).

Susa, Y., Yamashiro, Y., Yamamoto, M., Hen, I., Lidar, D. A., & Nishimori, H. (2018). Quantum annealing of the p-spin model under inhomogeneous transverse field driving. *Physical Review A*, *98*(4), 042326.

Susa, Y., Yamashiro, Y., Yamamoto, M., & Nishimori, H. (2018). Exponential speedup of quantum annealing by inhomogeneous driving of the transverse field. *Journal of the Physical Society of Japan*, *87*(2), 023002.

Svore, K. M., Hastings, M. B., & Freedman, M. (2013). Faster phase estimation. *arXiv preprint arXiv:1304.0741*.

Tasseff, B., Albash, T., Morrell, Z., Vuffray, M., Lokhov, A. Y., Misra, S., & Coffrin, C. (2022). On the emerging potential of quantum annealing hardware for combinatorial optimization. *arXiv e-prints*, arXiv–2210.

Temme, K., Osborne, T. J., Vollbrecht, K. G., Poulin, D., & Verstraete, F. (2011). Quantum metropolis sampling. *Nature*, *471*(7336), 87–90.

Ventresca, M., & Tizhoosh, H. R. (2007). Simulated annealing with opposite neighbors. In *2007 ieee symposium on foundations of computational intelligence* (p. 186-192). doi: 10.1109/FOCI.2007.372167

Wang, Y. (2012). Quantum computation and quantum information. *Statistical Science*, *27*(3), 373–394.

Wang, Y., & Liu, H. (2022). Quantum computing in a statistical context. *Annual Review of Statistics and Its Application*, *9*(1), 479–504.

Wang, Y., Wu, S., & Liu, H. (2023). Statistical analysis of quantum annealing. *Statistica Sinica*.

Wiebe, N., & Granade, C. (2016). Efficient bayesian phase estimation. *Physical review letters*, *117*(1), 010503.

Willsch, D., Willsch, M., Gonzalez Calaza, C. D., Jin, F., De Raedt, H., Svensson, M., & Michielsen, K. (2022). Benchmarking advantage and d-wave 2000q quantum annealers with exact cover problems. *Quantum Information Processing*, *21*(4), 141.

Wootters, W. K., & Zurek, W. H. (1982). A single quantum cannot be cloned. *Nature*, *299*(5886), 802–803.

Xinchao, Z. (2011). Simulated annealing algorithm with adaptive neighborhood. *Applied Soft Computing*, *11*(2), 1827-1836. doi: 10.1016/j.asoc.2010.05.029

Yamamoto, K., Duffield, S., Kikuchi, Y., & Muñoz Ramo, D. (2024). Demonstrating bayesian quantum phase estimation with quantum error detection. *Physical Review Research*, *6*(1), 013221.

Yarkoni, S., Raponi, E., Bäck, T., & Schmitt, S. (2022). Quantum annealing for industry applications: Introduction and review. *Reports on Progress in Physics*, *85*(10), 104001.