Computational complexity and quantum Gibbs sampling for local Hamiltonians

Thesis by
Jiaqing Jiang

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

CALIFORNIA INSTITUTE OF TECHNOLOGY
Pasadena, California

2025
Defended May 13th, 2025

Jiaqing Jiang
ORCID: 0000-0003-4055-1950

ACKNOWLEDGEMENTS

The PhD journey has been more interesting and transformative than I ever anticipated, filled with moments of doubt, discovery, and personal growth. I am deeply grateful to everyone who supported me along the way.

I would like to thank my advisors Thomas Vidick, Urmila Mahadev, and John Preskill. It took me several years to discover the research topics I truly enjoy. I am especially grateful to Thomas for giving me unlimited freedom to explore different directions, even when I shifted my interest from interactive proofs to investigating quantum advantages in physics and chemistry—areas outside his usual expertise. He was fully supportive and told me, "The most important thing is doing what you are interested in!" Without his support, this thesis would not have been possible. Urmila has been like a friend throughout my PhD. We frequently met to chat and discuss research just for fun. She encouraged me to pursue the problems I'm truly passionate about, and her enthusiasm for research has profoundly influenced me. John is incredibly knowledgeable and brings unique insights to the problems I work on. He also has a great sense of humor, often kicking off group meetings with a joke, which made them enjoyable and fun.

I am deeply grateful to my collaborator Sandy Irani. I feel very fortunate to have met someone who shares such similar research interests. With Sandy I solved my first project on local Hamiltonians. Since then, we have worked together on topics ranging from Hamiltonian complexity to quantum Gibbs sampling and have enjoyed many spontaneous and stimulating discussions along the way.

I would also like to sincerely thank Garnet Chan for kindly welcoming me into their group meetings. The discussions in these meetings, as well as my conversations with Garnet, gave me valuable insights into the questions chemists are investigating and the challenges that matter to them. I am also grateful to Chris (Jielun) Chen, Yongtao Zhan, Gunhee Park, Tomislav Begusic, Runze Chi, Sijin Du and many other members of Garnet's group for fun discussions and their generosity in helping me better understand tensor networks, the Fermi-Hubbard model, and various concepts in chemistry.

I am deeply grateful to many collaborators and group members. I want to thank Sandy Irani, Chris Chen, Norbert Schuch, Dominik Hangleiter, Yeongwoo Hwang, Yu Tong, Yiyi Cai, Akshar Ramkumar, Adam Artymowicz, and Fernando Brandao

for the fun collaborations and insightful discussions. I would like to thank Saeed Mehraban and Atul Singh Arora for their encouragement and support during times when I felt deeply stuck and frustrated in the early years of my PhD. I still remember our long conversation at Urth Caffé that lasted until 2 a.m., where we talked about research, life, and Orion. I am also grateful for the many research discussions, Friday dinners, and beers with: Robbie King, Alexander Poremba, Ulysse Chabaud, Jiayu Zhang, Chris Chen, Abhinav Deshpande, Saeed Mehraban, Atul Singh Arora, Chris Partison, Yu Tong, Mehdi Soleimanifar, Shouzhen (Bailey) Gu, Eugene Tan, Isabel Franco, Yongtao Zhan, Alexander Jahn, Hsin-Yuan (Robert) Huang, Chi-Fang (Anthony) Chen, Leo Zhou, Eric Anschuetz, Samson Wang, Tommy Schuster, Andreas Elben, Laura Cui, Haimeng Zhao, and Nat Tantivasadakarn.

I am also grateful to the Simons Institute for the Theory of Computing at UC Berkeley. I participated in several summer programs there, where I had the opportunity to meet many outstanding researchers. Much of my work was carried out during my visits to the Simons Institute.

Finally, I would like to thank my parents, Xiaoping Jiang and Chunlin Dai, and my friend Yixuan Song. Their continued and unconditional support has carried me through the many ups and downs of this journey.

ABSTRACT

One of the primary motivations for building quantum computers is to simulate quantum many-body systems. While significant progress has been made in simulating quantum dynamics, much less is known about simulating ground states and Gibbs states, an essential task for understanding the static properties of quantum many-body systems. From a computer science perspective, problems on ground states and Gibbs states are quantum analogues of the Boolean satisfiability problem (SAT) and classical Gibbs sampling, which have wide applications in optimization, machine learning, and computational complexity.

This thesis leverages tools from computer science to explore the potential quantum advantage in simulating ground states and Gibbs states, through two complementary approaches: designing new quantum algorithms and evaluating the extent to which classical algorithms remain effective. In particular,

- **Quantum Gibbs sampling.** In the first part, we describe our progress in developing quantum algorithms for preparing quantum Gibbs states. For general Hamiltonians, we develop a quantum analogue of the Metropolis-Hastings algorithm that is both conceptually simple and provably correct, with the Gibbs state as its approximate unique fixed point. Note that generalizing the Metropolis-Hasting algorithm to the quantum setting is non-trivial due to the unclonability of quantum states. Additionally, for a broad class of commuting Hamiltonians, we propose a different approach which constructs efficient quantum Gibbs samplers by leveraging reductions to existing classical sampling algorithms.

- **Sharpening the understanding of classical algorithms.** In the second part, we present new complexity results to deepen our understanding of the capabilities of classical algorithms for ground energy estimation. The potential quantum advantage in solving many-body systems stems from the sign problem in general Hamiltonians, which classical algorithms struggle to handle. We give rigorous evidence to show that under certain conditions, widely used classical methods, such as fixed-node Monte Carlo and tensor network contraction, may overcome this barrier and effectively resolve the sign problem.

PUBLISHED CONTENT AND CONTRIBUTIONS

[HJ25] Yeongwoo Hwang and Jiaqing Jiang. "Gibbs state preparation for commuting Hamiltonian: Mapping to classical Gibbs sampling". In: *Contributed Talk at the 28th Annual Quantum Information Processing Conference (QIP2025) arXiv preprint arXiv:2410.04909* (2025).
The author list is ordered alphabetically (all authors contributed equally). JJQ conceived the project and participated in the development of the proofs and the writing of the manuscript.

[Jia25] Jiaqing Jiang. "Local Hamiltonian problem with succinct ground state is MA-complete". In: *Contributed Talk at the 27th Annual Conference on Quantum Information Processing (QIP2024). PRX Quantum* 6.2 (2025), p. 020312.
JJQ conceived the project, developed the key ideas for the proofs, and wrote the manuscript.

[JI25] Jiaqing Jiang and Sandy Irani. "Quantum Metropolis Sampling via Weak Measurement". In: *Contributed Talk at the 28th Annual Quantum Information Processing Conference (QIP2025) arXiv preprint arXiv:2406.16023* (2025).
JJQ conceived the project, developed the key ideas for the proofs, and wrote the majority of the manuscript.

[Jia+25] Jiaqing Jiang et al. "Positive bias makes tensor-network contraction tractable". In: *Proceedings of the 57th Annual ACM Symposium on Theory of Computing (STOC '25); Contribued Talk at the 27th Annual Conference on Quantum Information Processing (QIP2024).* (2025).
JJQ participated in the conception the project and the development of the key ideas of the proof, and wrote the manuscript.

[IJ24] Sandy Irani and Jiaqing Jiang. "Commuting Local Hamiltonian Problem on 2D beyond qubits". In: *Contributed Talk at the 27th Annual Conference on Quantum Information Processing (QIP2024).* (2024).
The author list is ordered alphabetically (all authors contributed equally). JJQ participated in the conception of the project, the development of the proofs, and the writing of the manuscript.

TABLE OF CONTENTS

LIST OF ILLUSTRATIONS

Number *Page*

NOMENCLATURE

MA. A complexity class that denotes decision problems which can be verified in polynomial time by probabilistic classical computer.

NP. A complexity class that denotes decision problems which can be verified in polynomial time by deterministic classical computer.

QMA. A complexity class that denotes decision problems which can be verified in polynomial time by quantum computer.

Gibbs State. A state that describes a many-body system in thermal equilibrium at a given finite temperature.

Ground Energy. The minimum energy that a many-body system can have, defined as the smallest eigenvalue of the corresponding local Hamiltonian.

Ground State. A state that describes a many-body system at zero finite temperature. It is the eigenvector of the local Hamiltonian corresponding to the ground energy.

Guided State. A state that has $1/poly(n)$ overlap with the ground state. Often used as a warm start in both quantum and classical algorithms.

Local Hamiltonian. An n-qubit local Hamiltonian is a $2^n \times 2^n$ size matrix which can be written as written as sum of local terms, where local terms only act non-trivially on constant qubits.

Quantum and Classical Hamiltonians. We say a local Hamiltonian H is classical if it is a diagonal matrix in computational basis. Otherwise, H is called a quantum Hamiltonian. An example of a classical Hamilontian is the Ising model, while an example of a quantum Hamilontian is the transverse-field Ising model.

Quantum Gibbs Sampling. A quantum algorithm to prepare the Gibbs states of quantum Hamiltonian.

Quantum Many-Body System. A physical system composed of many interacting quantum particles, such as electrons or spins, whose collective behavior is governed by quantum mechanics.

Quantum Monte Carlo Method. A classical method to simulate quantum many-body system by using Monte Carlo sampling.

Tensor Network. A computational tool for classically estimating properties of many-body systems, such as ground state properties.

Chapter 1

INTRODUCTION

One of the primary motivations for building quantum computers is to simulate quantum many-body systems [Fey18]—complex systems of interacting particles governed by quantum mechanics, such as multi-electron molecules and materials. The potential quantum advantage in these simulations arises from the inherently quantum nature of the systems: the state of an n-qubit (quantum bit) system is described by a vector in a 2^n-dimensional Hilbert space, meaning that direct classical simulation of such systems requires an exponentially large amount of computational resources.

Over the past few decades, significant progress has been made in simulating quantum dynamics [Ber+14; Ber+15; BCK15; CW16; LC19], including the development of optimal protocols for Hamiltonian evolution [LC19]. However, much less is known about simulating ground states and Gibbs states, an essential task in understanding the static properties of a many-body system.

Ground states and Gibbs states are fundamental concepts that play a central role in numerous areas. From a computer science perspective, problems on ground states and Gibbs states are the quantum analogues of the Boolean satisfiability problem (3SAT) and the classical Gibbs sampling. Estimating the ground state energy is a central problem in quantum complexity theory [KKR06; AAV13; AGM20; AAG22a; IJ23], playing a role analogous to that of SAT in classical complexity theory. Algorithms for sampling from Gibbs states can be used to solve semidefinite programs [Bra+19; Van+17], with wide applications in optimization. Additionally, Gibbs states can serve as generative models in machine learning, forming the foundation of classical and quantum Boltzmann machines [HS83; Ami+18]. From a natural science perspective, ground states and Gibbs states describe quantum many-body systems at zero and finite temperatures. Their properties are crucial to understanding the phase diagrams and the electronic binding energies, which are central topics in quantum chemistry and condensed matter physics. Developing quantum algorithms and deepening the complexity-theoretic understanding of simulating Gibbs states and ground states may lead to breakthroughs in materials science and quantum chemistry [Cao+19; Alh23; Qin+22].

In this thesis, I leverage tools from computer science to investigate the potential quantum advantage in simulating Gibbs states and ground states. The exploration follows a twofold approach: developing novel quantum algorithms and evaluating the extent to which classical algorithms can simulate or dequantize these quantum processes. In particular, in this introduction, I will start by reviewing the key concepts. I will then describe our progress in developing new quantum algorithms for quantum Gibbs state preparation, as well as new computational complexity characterizations for estimating ground state energy that sharpen our understanding of the regimes in which classical algorithms remain effective.

1.1 Local Hamiltonians, Gibbs states, and ground states

Before presenting our results, we briefly review the key concepts.

n-qubit local Hamiltonian. An n-qubit local Hamiltonian H, which describes a many-body system, is a $2^n \times 2^n$ size matrix that can be written as a sum of local terms, i.e.

$$H = \sum_{i=1}^{m} H_i, \tag{1.1}$$

where $m = poly(n)$, each H_i is Hermitian and only acts non-trivially on at most constant qubits. We say that a local Hamiltonian H is *classical* if each H_i is a diagonal matrix. Otherwise, we say H is *quantum*.

Gibbs states and quantum Gibbs sampling: A many-body system at finite temperature is described by its *Gibbs state*, which is known as the thermal equilibrium state and is written as

$$\rho_\beta = \frac{\exp(-\beta H)}{\mathrm{Tr}\exp(-\beta H)}, \tag{1.2}$$

where H is the local Hamiltonian and β is the inverse temperature.

Given (H, β), the *quantum Gibbs sampling* task is to design a quantum algorithm that prepares the Gibbs states ρ_β with *provable* correctness and efficiency. Although this problem has been extensively studied for classical Hamiltonians such as the Ising model [Met+53; MO94; CCS87; Hol85; SW87; FGW23] (classical Gibbs sampling), much less is known when H is a quantum Hamiltonian. In Chapter 1.2 we will describe our progress in designing quantum algorithms for preparing quantum Gibbs states.

Ground states and ground energy estimation: While the Gibbs state characterizes many-body systems at finite temperature, the ground state of a local Hamiltonian H represents the state of a many-body system at zero temperature. Mathematically, the *ground state* of H is the eigenvector corresponding to its minimum eigenvalue, where the minimum eigenvalue is denoted as the *ground energy*.

The problem of finding the ground energy of a local Hamiltonian is a quantum generalization of the classical Boolean satisfiability problem (SAT), where SAT corresponds to determining the ground energy of a classical Hamiltonian. Analogous to the central role that solving the SAT problem plays in science, operations research, and engineering, estimating the ground energy of a quantum Hamiltonian plays a fundamental role in chemistry, physics, and quantum information. Consequently, many quantum and classical algorithms have been proposed to tackle this challenge [Whi92; Whi93; Sch05; Cao+19; McA+20; Bau+20], though the majority are heuristic.

In Chapter 1.3, we present new rigorous results for ground energy estimation, which offer more insight into the capabilities of classical algorithms to solve the ground energy estimation problem.

Complexity class. Here we briefly explain the complexity classes discussed later in Chapter 1.3. All the representative problems mentioned are complete for their respective complexity class.

- **NP** denotes the set of decision problems that can be verified in polynomial time by a *deterministic* classical computer. A representative problem in this class is the classical Boolean satisfiability problem (SAT).
- **MA** denotes the set of decision problems that can be verified in polynomial time by a *probabilistic* classical computer.
- **QMA** denotes the set of decision problems that can be verified in polynomial time by a quantum computer. A representative problem in this class is the local Hamiltonian problem (LHP), which is the decision version of the ground energy estimation problem [KKR06].
- **QCMA** represents the set of decision problems that can be verified in polynomial time by a quantum computer, where the proof can be represented by $poly(n)$ classical bits. A representative problem in this class is the local Hamiltonian problem (LHP) where the ground state can be prepared by polynomial-size quantum circuits [WJB03].

- **#P** is a class of counting problems. A representative problem in this class is #SAT, which counts the number of satisfying assignments in the SAT formula.

1.2 Quantum Gibbs sampling

Quantum Gibbs sampling is a crucial computational technique with wide application in chemistry, physics, and computer science [Alh23; HS83; Ami+18; Bra+19; Van+17]. In designing a quantum algorithm to prepare quantum Gibbs states, there are two fundamental goals:

1. The first is to design a quantum algorithm which *correctly* prepares the Gibbs states.
2. The second is to make the proposed quantum algorithm *efficient* (fast mixing).

In this thesis, we present two of our works that address the above goals.

Chapter 2: Quantum Metropolis Sampling via Weak Measurement

The goal of correctly preparing Gibbs states of classical Hamiltonians is achieved by the celebrated Metropolis algorithm [Met+53], which has become one of the most widely used algorithms throughout science. For quantum Hamiltonians, designing an algorithm which provably converges to the Gibbs state has been more challenging since mimicking the rejection process in quantum Metropolis requires reverting a quantum measurement, which is hard. In addition, quantum computers with finite resources cannot distinguish eigenvalues to infinite precision (energy-time uncertainty principle), which brings additional technical difficulties. To ease the two issues, previous work [Tem+11] designed a quantum Metropolis algorithm assuming a special variant of quantum phase estimation algorithm, which is suggested to be impossible by recent work [Che+23].

In Chapter 2, we [JI24] design a provably-correct quantum Metropolis-based algorithm for quantum Gibbs states preparation. We addressed the difficulties in the previous algorithm [Tem+11] by incorporating two new ideas: applying weak measurement and using two different ways to implement an accepting move. The only provably correct quantum Gibbs sampler before our work [Che+23] is based on an approach significantly different from classical Metropolis. It uses the weighted operator Fourier transform technique to give an approximate quantum simulation of

the Davies generator, a Lindbladian[1] closely related to the thermalization process in nature. Compared to previous work, the main advantage of our Gibbs sampler is its simplicity and the conceptual connection to the classical Metropolis algorithm. These advantages might make it easier to adapt variations of Metropolis algorithms that speed up the convergence of classical Markov chains to the preparation of quantum Gibbs states. Examples of variations that have been successful in the classical setting include delayed rejection [ZK11], equi-energy sampling [KZW06], and adaptive methods [HST01].

Chapter 3: Gibbs state preparation for commuting Hamiltonian

While significant progress has been made in developing provably-correct quantum Gibbs samplers, much less is known about the mixing times of those methods. Based on the Davies generator, recent papers [CRF20; Bar+23; KB16] have designed fast mixing Gibbs samplers for various commuting local Hamiltonians (CLHs), in particular for 1D CLH at any temperature [Bar+23; KB16] and 2D CLH at high temperature [CRF20; KB16].

In Chapter 3 we design novel Gibbs samplers for various CLHs by giving a reduction from quantum Gibbs sampling to classical Gibbs sampling, rather than using Davies generator. Those CLHs include all 2-local CLHs (either 1D or 2D) and a large class of qubit 4-local 2D CLHs, including the defected Toric code. Combined with the existing fast mixing results for classical Hamiltonians, our Gibbs sampler is able to replicate the state-of-the-art performances mentioned above [Bar+23; KB16; CRF20], as well as prepare the Gibbs state in regimes which were previously unknown, such as the low temperature region, as long as there exists fast mixing Gibbs samplers for the corresponding classical Hamiltonians. For example, we are able to utilize low-temperature classical Gibbs sampling techniques such as the Swendsen-Wang algorithm [FGW23] and Barvinok's method [Bor+20], to prepare low-temperature Gibbs states for certain 2-local CLHs.

1.3 Sharpen the understanding for classical algorithms

Quantum many-body systems have been studied for decades by physicists and chemists even before the emergence of quantum computers. Numerous powerful classical algorithms have already been proposed for classically estimating properties

[1] Note that Davies generator assumes the ability to distinguish eigenvalue to infinite precision, thus cannot be directly simulated on quantum computers without using the weighted operator Fourier transform techniques developed in [Che+23].

of Gibbs states and ground states, such as the Monte Carlo based method [GKW16; MZ18], the tensor network based approach [Whi92; Orú19; Bañ23], and variational methods [Fis87; CT17]. In seeking quantum advantage in quantum many-body systems, it is not only important to design efficient quantum algorithms, but it is also important to sharpen our understanding of the regions where classical algorithms work. The intuitive belief in a potential quantum advantage for solving many-body systems stems from the fact that general Hamiltonians are affected by the sign problem, which classical algorithms struggle to handle. In this thesis we describe several works that have enhanced our understanding of classical algorithms' ability to address the sign problem, thereby refining our understanding of the boundary of quantum advantage.

*Chapter 4: Local Hamiltonian problem with succinct ground state is **MA-complete***

In ground energy estimation, a guided state for a local Hamiltonian refers to a state with $1/\mathrm{poly}(n)$ overlap with the ground state, which is commonly used as a warm start in both quantum and classical algorithms. It was believed that quantum algorithms assisted with guided states could achieve higher precision in ground energy estimation than classical algorithms, potentially offering exponential quantum advantage for quantum chemistry [Cao+19; McA+20; Bau+20]. This belief was supported by the theoretical evidence that achieving high-precision in ground energy estimation is inherently a quantum problem (**QCMA-complete** [GL22]) while achieving low-precision is a classical problem (in **NP** [WFC23]).

However, it is important to note that guided states are often derived from classical heuristics, and can be used as a warm start for not only quantum algorithms but also classical algorithms. Recent numerical investigation by chemists [Lee+23] suggests that these guided states may possess more structure than assumed in previous complexity results [GL22; WFC23], an observation which could lead to efficient classical algorithms. In Chapter 4 we describe a rigorous complexity result [Jia25] to support the observation [Lee+23] that classical access to more structured guided states may invalidate the assumed quantum advantage. More specifically, we use the fixed node Monte Carlo method [Ten+95; Bra+23a] to demonstrate that *classical query access* to an extremely good guided state (ground state) will make high-precision ground energy estimation a classical problem; this problem becomes **MA-complete**, in contrast to the **QCMA-complete** result mentioned above. This suggests that the sign problem can be resolved by classical algorithms with the help of certain good guided states.

Chapter 5: Positive bias makes tensor-network contraction tractable

We also provide rigorous results that advance our understanding of the complexity of tensor network contractions, another widely used computational tool for classically estimating ground state properties [Whi93; Whi92; MVC07; VC21]. In particular, it is well-known that contracting random tensor networks with zero mean *exactly* is **#P**-hard, and remains numerically hard for approximate contraction. While it is expected that tensor network contraction becomes easier when all entries are positive (eliminating the so-called sign problem), in Chapter 5, we [Jia+24] give rigorous evidence that random tensor network contraction becomes tractable even when the tensor network is only *slightly* positive. In particular, we show that a *small* bias on the mean value already dramatically decreases the computational complexity of 2D tensor network contractions, enabling a quasi-polynomial approximation algorithm. This work provides rigorous support for previous observations made by chemists and physicists [GC24; Che+25].

Chapter 6: Commuting Local Hamiltonian Problem on 2D beyond qubits

The ground energy estimation problem is typically formalized as the Local Hamiltonian Problem (LHP): given an n-qubit local Hamiltonian $H = \sum_{i=1}^{m} H_i$ and two real numbers $a > b$, the goal is to decide whether the ground energy of H is greater than a or less than b, under the promise that one of these is true. While it is widely believed that there are quantum advantages for ground energy estimation, it is well-known that LHP is **QMA-complete** [KKR06], which indicates that even quantum computers cannot efficiently solve general LHP.

A natural question is to identify additional properties and understand how they weaken the hardness of LHP. In Chapter 6 we study the commuting variant of LHP, where we additionally assume that all the local terms H_i commute with each other. Compared to the general LHP, commuting LHP is conjectured to be more classical — potentially in **NP** — based on the intuition [FS97] in quantum physics that it is the non-commutativity that makes the quantum world different from classical (as illustrated by the Heisenberg uncertainty principle). However, despite two decades of study, the complexity of CLHP still remains widely open, with a few special cases known to be in **NP** ([BV03; AE13; AE11; Sch11; AKV18; Has12]). We approach this question by focusing on the special case of the CLHP defined on a 2D lattice. In Chapter 6, we will show that on 2D lattice the qutrit commuting LHP and the factorized commuting LHP are both in **NP**.

Chapter 2

QUANTUM METROPOLIS SAMPLING VIA WEAK MEASUREMENT

2.1 Introduction

One of the primary motivations for building quantum computers is to simulate quantum many-body systems. While there has been significant progress in simulating quantum dynamics [Ber+14; BCK15; CW16], much less is known about preparing ground states and Gibbs states, an essential task in understanding the static properties of a system. In particular, the properties of Gibbs states, which describe the thermal equilibrium of a system at finite temperature, are closely related to central topics in condensed matter physics and quantum chemistry [Alh23], such as the electronic binding energy and phase diagrams. In addition to applications in physics, Gibbs states are widely used as generative machine learning models, such as classical and quantum Boltzmann machines [HS83; Ami+18]. Algorithms for preparing Gibbs states are also used as subroutines in other applications, such as solving semidefinite programs [Bra+19; Van+17].

Typically, a good Gibbs state preparation algorithm (Gibbs sampler) should satisfy two requirements: it should have the Gibbs state as its *(unique) fixed point* and it should be *rapidly mixing*. The fixed point property ensures the *correctness* of the Gibbs sampler. An algorithm that keeps the Gibbs states invariant and shrinks any other state will eventually converge to the Gibbs states after a sufficiently long time. Our algorithm satisfies an approximate version of the fixed point property. The mixing time determines the *efficiency* of the algorithm. In particular, an algorithm is said to be fast mixing if it convergences to Gibbs state in $poly(n)$ time.

The focus of this manuscript is the correctness part, that is, designing a Gibbs sampler which satisfies the fixed point property. For classical Hamiltonians, like the Ising model, the fixed point property is easily satisfied by the celebrated Metropolis algorithm [Met+53] which has become one of the most widely used algorithms throughout science. For quantum Hamiltonians, designing an algorithm which provably converges to the Gibbs state has been more challenging. As noticed in the pioneering work of [Tem+11] ten years ago, designing a Metropolis-type algorithm for quantum Hamiltonians is non-trivial, mainly due to two reasons:

(1) Quantum computers with finite resources cannot distinguish eigenvalues and eigenstates to infinite precision.

(2) Mimicking the rejection process in quantum Metropolis requires reverting a quantum measurement.

[Tem+11] eased the first challenge using a boosted version of QPE which sharpens the accuracy by taking the median of multiple runs. They addressed the second challenge using the Marriott-Watrous rewinding technique along with a shift-invariant version of QPE in the case of a rejected move. As a result, their analysis depends on a version of QPE that is both boosted and shift-invariant. Recent work [Che+23] suggests a version of QPE with both of those properties may be impossible. While [Tem+11] provides many innovative ideas, a provably correct quantum Gibbs sampler remained elusive for some time. Recently, Chen *et.al.* [Che+23] designed the first Gibbs sampler which provably satisfies the fixed point property for general Hamiltonians based on a significantly different approach. Their method is based on simulating quantum master equations (Lindbladians) which more closely mimics the way that systems thermalize in nature. Their algorithm approximately simulates the Davies generator [Dav76; Dav79], which describes the evolution of quantum systems coupled to a large heat bath in the weak coupling limit. It is worth mentioning that the Davies generator by itself also assumes the ability to distinguish eigenvalues to infinite precision, and thus cannot be efficiently simulated by quantum computers. To resolve this problem, [Che+23] devised a method to smooth the Davies' generator by using a weighted operator Fourier Transform for Lindbladians.

Although [Che+23] provides a provably correct quantum Gibbs sampler by simulating the thermalization process occurring in nature, it is natural to ask whether a Metropolis-style quantum Gibbs sampler can be designed. Are there intrinsic reasons why an algorithm based on the classical Metropolis process cannot work for quantum Hamiltonians? Or on the other hand:

Is it possible to design a provably correct quantum Gibbs sampler, which is analogous to the conceptually simple classical Metropolis algorithm?

In this manuscript, we give an affirmative answer to the above question, by designing a simple Metropolis-style Gibbs sampler. Our algorithm uses many of the components of [Tem+11], but there are some key differences: (1) we use *weak*

measurement in determining whether to accept or reject a given move; (2) we use a Boosted QPE and do not assume the shift-invariant property. (3) we do not use Marriott-Watrous Rewinding tehchnique [MW05], which simplifies the algorithm considerably.

For (3), more precisely, recall that the mechanism that [Tem+11] uses to back up in a reject case requires a $poly(n)$ sequence of forward and backward unitaries and complex measurements until the backing up process succeeds. In comparison, we use only *one single-qubit measurement* and *one unitary* for rewinding. This simplification is achieved by noticing that after one round of unsuccessful rewinding, the state is almost equivalent to the state in the accept case. This allows us to accept and conclude the iteration in one step instead of attempting to rewind again. We call this case an *Alternate Accept*. This observation is an essential feature of our analysis, since we would still need to perform Marriott-Watrous rewinding without the Alternate Accept case, even with the weak measurement. One remark is that while weak measurement helps in rewinding, it comes at the expense of increasing the number of iterations by a polynomial factor. Also, while our algorithm is simple, it does not have the most favorable scaling as a function of the system parameters and desired precision.

We note that weak measurement is also one of the reasons why the approaches based on the Davies generator succeed, since simulating a Lindbladian requires the use of weak measurement. Our algorithm also effectively approximates the evolution of a Lindbladian. In this sense our algorithm is conceptually similar to [CKG23]. The key difference is that our Gibbs sampler is directly designed from QPE instead of a Davies generator.

2.2 Overview

Algorithm Overview

Given a local Hamiltonian H and an inverse temperature β, the goal is to design a quantum algorithm which prepares the Gibbs states $\rho_\beta = \exp(-\beta H)/Z$, where $Z = tr(\exp(-\beta H))$.

Our quantum algorithm attempts to mimic the classical Metropolis algorithm similar to [Tem+11]. Recall that the classical Metropolis algorithm is a random walk whose states are eigenstates of a classical Hamiltonian. In each iteration, the algorithm starts in some state x with energy ν_x. A jump operator is applied to alter x in some way to obtain a new state y with energy ν_y. Then a randomized decision is made

whether to Accept the move and remain in state y, or Reject the move and revert back to x. The acceptance probability is defined by a function $f_{\nu_x\nu_y} \in [0,1]$. The Metropolis acceptance rule is designed so that the random walk converges to the Gibbs state. In particular, the rule must satisfy

$$\exp(-\beta\nu_x) f_{\nu_x\nu_y} = \exp(-\beta\nu_y) f_{\nu_y\nu_x}.$$

Metropolis sampling uses the following function f:

$$f_{\nu_x\nu_y} := \min\left\{1, \exp\left(\beta\nu_x - \beta\nu_y\right)\right\}.$$

The main obstacle in adapting the classical Metropolis algorithm to the quantum setting, is that a measurement must be performed in deciding whether to accept or reject. Then in the Reject case, one needs to rewind back to the state before the move, thus reverting a quantum measurement, which is hard. The algorithm presented here manages this difficulty effectively by using weak measurement in determining whether to Accept or Reject. It is worth noting that, in contrast to the classical case where we can compute the energy ν_x exactly, there are intrinsic limitations on our ability for estimating energies of quantum states (due to the energy-time uncertainty principle). Analyzing the errors incurred by imperfect quantum energy estimation is non-trivial, and is one of the most technical parts in all related works [Che+23; CKG23; WT23; Tem+11]. We will explain more in the overview of techniques section.

The quantum algorithm uses four registers. The first is an n-qubit register which stores the current state of the algorithm. The next two each have gr qubits for integers g, r and are used to store the output of an application of the Boosted Quantum Phase Estimation (BQPE) algorithm, which provides an estimate of the state's eigenvalue. The last register is a single qubit register which controls whether we accept or reject the new state.

The algorithm uses three different operations outlined below. Let H be an n-qubit local Hamiltonian. We use $\left\{\left|\psi_j\right\rangle, E_j\right\}_j$ to denote an ortho-normal eigenbasis of H and their corresponding eigenvalues.

Boosted Quantum Phase Estimation (BQPE): BQPE is a unitary on two registers of n and gr qubits respectively. If BQPE starts with an eigenstate of H in the first register and the second register is initialized to g copies of $|0^r\rangle$, then

BQPE corresponds to performing g independent iterations of the standard Quantum Phase Estimation with respect to the first register and storing the result in each copy of $|0^r\rangle$. This process leaves the first register unchanged and outputs g independent estimates of E_j in the second register. Each r-bit string $\boldsymbol{b}$ in the second register represents an energy $E(\boldsymbol{b})$ defined as

$$E(\boldsymbol{b}) := \kappa_H \sum_{j=1}^{r} b_j 2^{-j}, \tag{2.1}$$

where κ_H is a power of two that upper bounds $\|H\|$. The set $S(r) := \{E(\boldsymbol{b})\}_{\boldsymbol{b}\in\{0,1\}^r}$ is the set of energies that can be represented by r-bit strings, which are integer multiples of $\kappa_H \cdot 2^{-r}$. To ease notation, we use notation $|E\rangle$ for $E \in S(r)^{\otimes g}$ as the basis of the second register, instead of using strings in $\{0,1\}^{rg}$. Thus, BQPE operates as

$$\text{BQPE}\left|\psi_j\right\rangle |0^{gr}\rangle = \left|\psi_j\right\rangle \sum_{E\in S(r)^{\otimes g}} \beta_{jE} |E\rangle\,, \tag{2.2}$$

The cost of BQPE is $g \cdot poly(2^r, n)$.

We write $\overline{E}$ as the median of the g energy estimates in E. We denote $\lfloor E_j \rfloor, \lceil E_j \rceil$ as the best two approximations of E_j in $S(r)$, that is the closet value to E_j which is an integer multiples of $\kappa_H \cdot 2^{-r}$ and is smaller/greater than E_j respectively. More details on BQPE are given in Appendix 2.9.

We also use a variant of BQPE, which we call FBQPE (Flipped Boosted Quantum Phase Estimation), where the amplitudes of the output of FBQPE are the complex conjugates of BQPE:

$$\text{FBQPE}\left|\psi_j\right\rangle |0^{gr}\rangle = \left|\psi_j\right\rangle \sum_{E\in S(r)^{\otimes g}} \beta^*_{jE} |E\rangle\,.$$

The implementation of FBQPE is a slight modification of BQPE and is given in Appendix 2.9. It is worth noting that FBQPE $\neq$ BQPE†.

Jump operators: A set of unitaries $\{C_j\}_j$ called *jump operators* and a distribution μ over this set. We require that the set $\{C_j\}_j$ is closed under adjoint. In addition, for any $C \in \{C_j\}_j$, we require that μ chooses C and $C^\dagger$ with the same probability. We use $C \leftarrow \mu$ to denote a selection of C drawn according to distribution μ.

To ensure the uniqueness of the fixed point, we also require that the algebra generated by $\{C_j\}_j$ is equal to the full algebra, that is the set of all n-qubit operators. For example one can choose $\{C_j\}_j$ to be all single-qubit Paulis.

Acceptance Operator (W): Finally, we use a unitary which calculates the acceptance probability based on the two energies stored in registers 2 and 3, scaled by a factor of τ^2 and rotates the last qubit by the square root of the acceptance probability. More precisely, W operates on registers 2, 3, and 4 as

$$W := \sum_{E,E' \in S(r)^{\otimes g}} |EE'\rangle\langle EE'| \otimes \begin{bmatrix} \sqrt{1-\tau^2 f_{EE'}} & \tau\sqrt{f_{EE'}} \\ \tau\sqrt{f_{EE'}} & -\sqrt{1-\tau^2 f_{EE'}} \end{bmatrix}, \tag{2.3}$$

where $f_{EE'}$ is the Metropolis acceptance rate based on the median of energy estimates:

$$f_{EE'} := \min\left\{1, \exp\left(\beta\overline{E} - \beta\overline{E'}\right)\right\}. \tag{2.4}$$

We can think of W as computing the median of E and E' to get $\overline{E}$ and $\overline{E}'$ respectively and then rotating the last qubit w.r.t $f_{EE'}$. The median operation $\overline{E}$ is used to boost the energy estimation, suppressing the probability of an incorrect estimate with Chernoff bounds. The operator W is the same as the one used in [Tem+11] with the addition of the slow-down factor of τ^2. Note that

$$W|EE'\rangle|0\rangle = |EE'\rangle\left(\sqrt{1-\tau^2 f_{EE'}}|0\rangle + \tau\sqrt{f_{EE'}}|1\rangle\right). \tag{2.5}$$

With those components defined, we can now describe an iteration of our algorithm depicted in Figure 2.1. The Algorithm Sketch below is a high-level overview of the algorithm. The complete pseudo-code is given in Section 2.4.

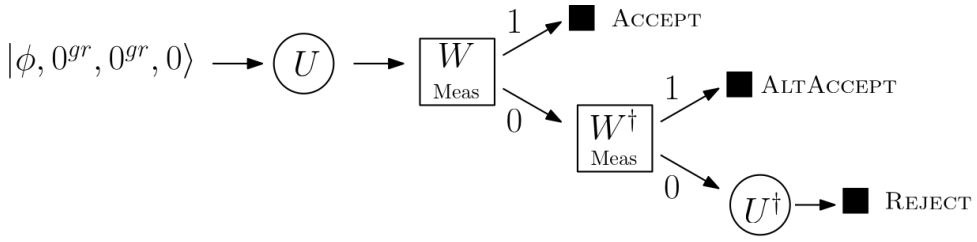


Figure 2.1: One iteration of the proposed Quantum Metropolis algorithm. The operation U is $\mathrm{QPE}_{1,3} \circ C \circ \mathrm{QPE}_{1,2}$. The two measurements are performed on the last qubit only. The ■ symbol indicates that the last three registers are traced out and replaced by fresh qubits in the $|0\rangle$ state.

Algorithm Sketch: In each iteration, the algorithm chooses one of {BQPE, FBQPE} with equal probability. The selected operation is called QPE. A jump operator $C \leftarrow \mu$ is also selected. We use C to denote the random choices for QPE and C made in a particular iteration. The algorithm starts an iteration with a state $|\phi\rangle$ in register 1 and all 0's in the other three registers. Then:

- QPE is first applied to the current state, and the estimate of the eigenvalue is stored in register 2. Then the jump operator C is applied to the state in register 1 to obtain a new state. QPE is then applied to the new state in register 1 and the estimate of its eigenvalue is stored in register 3. We call the sequence of these three operations $U_C := \mathrm{QPE}_{1,3} \circ C \circ \mathrm{QPE}_{1,2}$.
- Then W is applied to registers 2, 3, and 4, and the last qubit is measured to get Outcome1.
- If Outcome1 = 1, the algorithm accepts the move (Case: Accept) and continues.
- If Outcome1 = 0, then $W^\dagger$ is applied and the last qubit is measured again to get Outcome2.
  - The case in which Outcome2 = 1 represents an alternative way of accepting the move (Case: AltAccept).
  - If Outcome2 is 0, then this represents a rejection of the move (Case: Reject), in which case $U_C^\dagger$ is applied.
- Finally, registers 2, 3, and 4 are traced out and replaced by fresh qubits in all 0's states.

Informal Statement of Results

Let $\mathcal{E}(\rho)$ be the map corresponding to one iteration of the algorithm. Recall that τ is the parameter in the acceptance operator W, and g, r are the precision parameters in BQPE. Our main result is proving that our algorithm approximately fixes the Gibbs states:

Theorem 1 (Informal version of theorem 4) *$\mathcal{E}$ can be expressed as*

$$\mathcal{E} = \mathcal{I} + \tau^2 \mathcal{L} + \tau^4 \mathcal{J},$$

where $\mathcal{L}$ is independent of τ and approximately fixes the Gibbs state. More precisely for any δ, parameters g and r can be chosen so that $g = O(n + \log 1/\delta)$ and $r = O(\log \beta + \log \kappa_H + \log 1/\delta)$, and

$$\left|\mathcal{L}(\rho_\beta)\right|_1 \leq \delta.$$

Here $|\cdot|_1$ refers to the trace norm.

The proof that the fixed point of our algorithm is approximately the Gibbs State depends upon the assumption that $\mathcal{L}$ is fast mixing, meaning that $\mathcal{L}$ converges to its fixed point $\rho_{\mathcal{L}}$ in $poly(n)$ time. More precisely, we can combine the fast mixing property with the fact that $\left|\mathcal{L}(\rho_\beta)\right|_1 \approx 0$ from Theorem 1 above to establish that the fixed point of $\mathcal{L}$ is close to the true Gibbs state: $\rho_{\mathcal{L}} \approx \rho_\beta$. The next step then is to show that starting from an arbitrary state, repeated iterations of our algorithm will result in a good approximation of $\rho_{\mathcal{L}}$. In particular, the second part of Theorem 1, which says that $\mathcal{E} = \mathcal{I} + \tau^2\mathcal{L} + \tau^4\mathcal{J}$, implies that $\mathcal{E}^K \approx e^{K\tau^2\mathcal{L}}$, where $\mathcal{E}^K$ corresponds to K iterations of the algorithm. The error in the approximation scales as $K\tau^4$. K and τ can be chosen so that $K\tau^2$ is polynomial in n but the approximation error $K\tau^4$ is still small. Assuming that $\mathcal{L}$ is fast mixing, we can conclude that $e^{K\tau^2\mathcal{L}}$ drives any states to $\rho_{\mathcal{L}}$. This reasoning leads to Theorem 2 below.

Theorem 2 (Informal version of Theorem 6) *Suppose the mixing time of $\mathcal{L}$ is $poly(n)$. Assume $\beta, \kappa_H \leq poly(n)$. For parameters $\tau = 1/poly(n)$, $g = O(n)$, $r = O(\log n)$, $K = poly(n)$, and for any initial state ρ, we have*

$$\left|\mathcal{E}^K(\rho) - \rho_\beta\right|_1 \leq 1/poly(n). \tag{2.6}$$

Theorem 2 can also be rephrased in terms of the spectral gap similarly as in the classical Metropolis algorithm. The formal statement is in Corollary 9.

For the above choice of parameters the cost of implementing $\mathcal{E}^K$ is $Kg \cdot poly(2^r, n) = poly(n)$, where the $poly(2^r, n)$ is mainly the cost of implementing one run of standard Quantum Phase Estimation.

Overview of Techniques

Intuitively our algorithm should approximately fix the Gibbs state, since it mimics the classical Metropolis. The approximation errors come from two resources: one is controlled by the parameter τ in the acceptance operator W, and the other is controlled by the g and r in BQPE.

Let us begin with the error from τ. According to the algorithm, $\mathcal{E}$ can be expressed as the sum of three operators representing the three cases: $\mathcal{E}^{(a)}$ (for Accept), $\mathcal{E}^{(b)}$ for (AltAccept), and $\mathcal{E}^{(r)}$ (for Reject). So that $\mathcal{E} = \mathcal{E}^{(a)} + \mathcal{E}^{(b)} + \mathcal{E}^{(r)}$. Each of these can be further expanded to represent their dependence on τ. Mores specifically, in Section 2.5, we define additional operators, $\mathcal{M}^{(a)}$, $\mathcal{M}^{(r)}$, $\mathcal{J}^{(b)}$, $\mathcal{J}^{(r)}$, all with norm

bounded by a constant. We show that

$$\textsc{Accept}: \quad \mathcal{E}^{(a)} = \tau^2 \mathcal{M}^{(a)} \tag{2.7}$$

$$\textsc{AltAccept}: \quad \mathcal{E}^{(b)} = \tau^2 \mathcal{M}^{(a)} + \tau^4 \mathcal{J}^{(b)} \tag{2.8}$$

$$\textsc{Reject}: \quad \mathcal{E}^{(r)} = \mathcal{I} - \tau^2 \mathcal{M}^{(r)} + \tau^4 \mathcal{J}^{(r)} \tag{2.9}$$

Note that the operators for the Accept and AltAccept cases only differ by an operator on the order of τ^4, which means that the state in the AltAccept is very close to the resulting state in the Accept case. By defining $\mathcal{L} = 2\mathcal{M}^{(a)} - \mathcal{M}^{(r)}$ and $\mathcal{J} = \mathcal{J}^{(b)} + \mathcal{J}^{(r)}$, we have that

$$\mathcal{E} = \mathcal{I} + \tau^2 \mathcal{L} + \tau^4 \mathcal{J}.$$

Most of the technical effort in the proof of Theorem 1 is spent showing that $\left|\mathcal{L}(\rho_\beta)\right|_1 \approx 0$. There are two features of BQPE that make this fact challenging to prove. The first feature is that BQPE cannot be made deterministic. More precisely, recall that $\lfloor E_j \rfloor$ and $\lceil E_j \rceil$ are the best two approximations of E_j. Without additional assumptions on the Hamiltonian, the amplitudes $\beta_{j\lfloor E_j \rfloor}$ and $\beta_{j\lceil E_j \rceil}$ are both non-negligible. We use the boosted version of BQPE which guarantees that the probability of generating an estimate E that is different from $\lfloor E_j \rfloor$ or $\lceil E_j \rceil$ is negligibly small. However, the fact that the output of BQPE will still be a superposition of $\lfloor E_j \rfloor$ and $\lceil E_j \rceil$ is unavoidable and makes the process inherently different from a classical random walk. Mathematically, this means that if the algorithm begins in an eigenstate $\left|\psi_j\right\rangle$, after one iteration, the new state is no longer diagonal in the energy eigenbasis. In particular, $\mathcal{L}(\rho_\beta)$ can have exponentially many non-zero off-diagonal entries when expressed in the energy eigenbasis.

The second feature of BQPE that makes the analysis problematic is that there are intrinsic limitations on the precision of energy estimation of quantum states. In particular, the energy/time uncertainty principle suggests that a $poly(n)$-time quantum algorithm (like BQPE) can only estimate the energy of a state within $1/poly(n)$ precision. This means that the off-diagonal entries of $\mathcal{L}(\rho_\beta)$ can have magnitude on the order of $1/poly(n)$. The fact that $\mathcal{L}(\rho_\beta)$ can have exponentially many off-diagonal entries that have magnitude $1/poly(n)$ rules out a brute-force approach to bounding $\left|\mathcal{L}(\rho_\beta)\right|_1$.

To illustrate our approach in overcoming these technical difficulties, first imagine instead an ideal version of BQPE which deterministically maps every $\left|\psi_j\right\rangle$ to $\lfloor E_j \rfloor$.

The resulting process has a fixed point which is a Gibbs state where the probability of each state is proportional to $e^{-\beta\lfloor E_j\rfloor}$ instead of $e^{-\beta E_j}$. Call this truncated fixed point $\rho_{\beta 0}$. It's not too hard to show that $\rho_{\beta 0} \approx \rho_\beta$, which implies $\mathcal{L}(\rho_{\beta 0}) \approx \mathcal{L}(\rho_\beta)$, so we will focus instead on bounding $\left|\mathcal{L}(\rho_{\beta 0})\right|_1$. For the practically realizable, non-deterministic BQPE, our analysis effectively decomposes $\mathcal{L}(\rho_{\beta 0})$ into a sum of a constant number of terms and bounds the trace norm of each term separately by a $1/poly(n)$. The different terms are derived by inserting different projectors that separate out the cases when BQPE maps a state $\left|\psi_j\right\rangle$ to $\lfloor E_j\rfloor$, $\lceil E_j\rceil$, or some other E entirely. We can represent the cases by defining projectors:

$$P^{(0)} = \sum_j |\psi_j\rangle\langle\psi_j| \otimes \sum_{E:\overline{E}=\lfloor E_j\rfloor} |E\rangle\langle E|$$
$$P^{(1)} = \sum_j |\psi_j\rangle\langle\psi_j| \otimes \sum_{E:\overline{E}=\lceil E_j\rceil} |E\rangle\langle E|.$$

When the matrix

$$\mathcal{L}(\rho_{\beta 0}) = \sum_j \frac{e^{-\beta\lfloor E_j\rfloor}}{Z} \langle\psi_m| \mathcal{L}(\left|\psi_j\right\rangle \left\langle\psi_j\right|) |\psi_n\rangle$$

is written out, there are four applications of phase estimation: $\text{BQPE}_{1,2}$ and $\text{BQPE}_{1,3}$ are applied to $\left|\psi_j\right\rangle$, and $\text{BQPE}^\dagger_{1,2}$ and $\text{BQPE}^\dagger_{1,3}$ are applied to $\left\langle\psi_j\right|$. This results in a total of 16 terms depending on which of the two projectors ($P^{(0)}$ or $P^{(1)}$) is applied after each occurrence of BQPE. Recall that $\mathcal{L} = 2\mathcal{M}^{(a)} - \mathcal{M}^{(r)}$. In most cases, we don't get exact cancellation between the accept and reject operators because each case may have some multiplicative error of the form $e^{\pm\beta\delta}$, where $\delta = \lceil E_j\rceil - \lfloor E_j\rfloor$ is the precision of BQPE (which is independent of j). The essential observation is that each separate case results in exactly the same multiplicative error. This is because in each case, all of the BQPE estimates are erring in exactly the same direction and by exactly the same amount. We get that $\mathcal{L}(\rho_{\beta 0})$ can be expressed as a linear combination of terms $\{N_j\}$:

$$\mathcal{L}(\rho_{\beta 0}) \approx \sum_{j=1}^{16} \text{error}_j \cdot N_j, \tag{2.10}$$

where $\text{error}_j = e^{\pm\beta\delta} - 1$ and $\left|N_j\right|_1 = O(1)$. In the proof we manage to reduce the number of terms from 16 to 6 by carefully clustering terms. Note that the parameters are chosen so that $e^{\beta\delta} = (1+1/\text{poly}(n))$. Equation (2.10) is still approximate because we have not yet taken into account the case where BQPE maps $\left|\psi_j\right\rangle$ to some value

other than $\lfloor E_j \rfloor$ or $\lceil E_j \rceil$. For this case, we have a third projector:

$$P^{(else)} = \sum_j |\psi_j\rangle\langle\psi_j| \otimes \sum_{E:\overline{E}\neq\lfloor E_j\rfloor,\lceil E_j\rceil} |E\rangle\langle E|.$$

When $P^{(else)}$ is used, the norm of the resulting operator is exponentially small because of the use of boosted QPE. Specifically, we use the following lemma which is included in Appendix 2.9.

Lemma 3 *If $|\psi_j\rangle$ is an eigenstate of H with energy E_j, then*

$$BQPE\,|\psi_j\rangle\,|0^{gr}\rangle = |\psi_j\rangle \sum_{E\in S(r)^{\otimes g}} \beta_{jE}\,|E\rangle\,, \tag{2.11}$$

$$\textit{where} \sum_{E\in S(r)^{\otimes g}:\,\overline{E}\neq\lfloor E_j\rfloor,\overline{E}\neq\lceil E_j\rceil} |\beta_{jE}|^2 \leq 2^{-g/5}. \tag{2.12}$$

Our proof works for general Hamiltonians and we do not assume non-physical assumptions like rounding promise [WT23]. One more remark is that BQPE itself will create some biased phase in the resulting states. In our algorithm, we choose BQPE and FBQPE randomly to cancel this bias. Finally, the proof of uniqueness of the fixed point, which appears in Section 2.6, is standard, and is based on showing that $\mathcal{L}$ is of Lindbladian form, and that the generators of the Lindbladian generate the full algebra of operators on the n-qubit Hilbert space.

Related Work

We have recently become aware of the concurrent, independent work of [Gil+24a], which also provides a quantum generalization of Glauber/Metropolis dynamics. In contrast with our algorithm, their method does not use QPE. Instead, they construct a quantum extension of discrete and continuous-time Glauber/Metropolis dynamics, in the style of a smoothed Davies generator [Che+23; CKG23]. They show that their construction exactly fixes the Gibbs states and can be efficiently implemented on a quantum computer. For the continuous-time case, their implementation is achieved by Linbladian simulation [Che+23]. For the discrete-time case, they use oblivious amplitude amplification [Gil+19] in combination with techniques based on linear combinations of unitaries and the quantum singular value transform.

Before the appearance of [Che+23], there were many previous results on simulating Davies generators, with additional assumptions on the Hamiltonian. As mentioned before, Davis generators cannot be implemented exactly since they require the ability to estimate eigenvalues perfectly, which is impossible with quantum computers

with finite resources. [WT23] circumvents this problem by assuming a rounding promise on the Hamiltonian, which disallows eigenvalues from certain sub-intervals. [RWW23] later eliminates the rounding promise by using randomized rounding, which at the same time incurs an additional resource overhead. More recently, [CKG23] designed a weighted version of Davies generator which exactly fixes the Gibbs states (before truncating the infinite integral to a finite region). [DLL24] introduces a family of quantum Gibbs samplers satisfying the Kubo-Martin-Schwinger detailed balance condition, which includes the construction of [CKG23] as a special instance. In addition to approaches based on the Davies generator, there are Grover-based approaches [PW09; CS16], which prepare a purified version of Gibbs states. The performance of those algorithms depends on the overlap between the initial and the target state.

All of the above approaches use either quantum phase estimation or quantum simulation, which involve large quantum circuits. In contrast to those approaches, [ZBC23] designs a dissipative quantum Gibbs sampler with simple local update rules. [ZBC23] differs from the ordinary Gibbs samplers in that the Gibbs state is not generated as the fixed point of the Markov Chain, but is generated on a conditionally stopped process.

All the above work is focused on satisfying the fixed point requirement. A different but important task is to bound the mixing time, which is wide open with the exception of a few special cases. In particular, [KB16] shows that for a commuting Hamiltonian on a lattice, there is an equivalence between very rapid mixing (more precisely, constant spectral gap of the Linbladian) and a certain strong form of clustering of correlations in the Gibbs state, which generalizes the classical result [SZ92a; SZ92b] to the quantum setting. [Bar+23] proves fast mixing for 1D commuting Hamiltonian at any temperature. Recently [RFA24; Bak+24] prove fast mixing for non-commuting Hamiltonian at high temperature.

There are also many heuristic methods for preparing Gibbs states. Previous proposals include methods that emulate the physical thermalization process by repeatedly coupling the systems to a thermal bath [TD00; SN16; Met+20]. There are also approaches based on quantum imaginary time evolution [Mot+20] and the variational algorithms [WLW21; Lee+22; Con+23].

Conclusions and Future Work

In this manuscript, we use weak measurement to design a quantum Gibbs sampler based on the Metropolis algorithm which satisfies the approximate fixed point property. Compared with previous work, the main advantage of our algorithm is its conceptual simplicity. We hope that our Metropolis-style Gibbs sampler will make it easier to adapt variations of the Metropolis algorithm that speedup the convergence of classical Markov chains to the preparation of quantum Gibbs states. Examples of variations that have been successful in the classical setting include delayed rejection [ZK11], Equi-energy sampling [KZW06], and adaptive methods [HST01].

Our algorithm uses a Boosted QPE which takes the median value of multiple independent runs of the standard QPE. We do not require a version of QPE that satisfies shift-invariance. In addition, our algorithm is free of Marriott-Watrous rewinding techniques and only performs single-qubit measurements. Technically, we give a new way of bounding the errors incurred by imperfect energy estimation of QPE, by grouping the error terms into finite classes. Our proof works for general Hamiltonians and we do not assume the rounding promise. It might be interesting to investigate whether this technique can be applied to prove that the existing Gibbs Sampler for Hamiltonians with rounding promise [WT23] in fact works for general Hamiltonians.

While our algorithm is simple, it comes at a cost of not having the most favorable scaling in the parameters of the system and desired precision. In particular, our algorithm effectively corresponds to directly simulating a Lindblad evolution $e^{t\mathcal{L}}$ by discretizing t, where the cost is scaled as $O(t^2/\epsilon)$ for precision parameter ϵ. It is worth noting, however, that our algorithm itself is designed directly from QPE, and we use $e^{t\mathcal{L}}$ only in our analysis, as opposed to first designing the Lindbldian $\mathcal{L}$ and attempting to simulate it on a quantum computer. It would be interesting to explore how more sophisticated Linbladian simulation techniques (such as [LW22; CW16; CL16]) could be applied to our algorithmic structure to improve the dependence on parameters. Note that our $e^{t\mathcal{L}}$ circumvents the problem of precision in the Davies generator, because the operator $\mathcal{L}$ is already defined in terms of QPE of finite precision.

Another possible direction for future work is to compare the mixing time of different Gibbs samplers. In particular, it would be interesting to compare, either numerically or theoretically, the spectral gap of the Davies generator, the Davies-generator-inspired Lindbldian in [Che+23; CKG23], and the Lindbladian $\mathcal{L}$ in our algorithm.

Since our algorithm is similar to the classical Metropolis algorithm, it might also be interesting to explore whether classical techniques for analyzing mixing times can be generalized to the quantum setting.

Structure of the manuscript

The manuscript is structured as follows. In Section 2.3 we give necessary definitions and notations. In Section 2.4 we give explicit pseudo-code of our algorithm and express the result of each operation more formally. We state our main results in Theorem 4 and Theorem 6 at the beginning of Section 2.5.

The proofs of Theorem 4 and Theorem 6 are divided into several sections. Section 2.5 contains the derivation of the explicit formula for $\mathcal{L}$. In Section 2.6 we show that $\mathcal{L}$ can be written as a Lindbladian and has a unique fixed point. Section 2.6 is independent of Section 2.7 and can be skipped temporarily. In Section 2.7 we prove $\mathcal{L}$ approximately fixes the Gibbs state. Finally in Section 2.8 we prove Theorem 4 and Theorem 6.

2.3 Preliminary and Settings

Gibbs states and Assumptions

For any n-qubit Hamiltonian H, we always use $\left\{\left|\psi_j\right\rangle, E_j\right\}_j$ to denote an orthonormal set of (eigenstate, eigenvalue) for H. We use symbols different from $\left|\psi_j\right\rangle$ to denote other states. For any inverse temperature $\beta \geq 0$, we denote the Gibbs state as

$$\rho_\beta(H) := \exp(-\beta H)/tr(\exp(-\beta H)) \tag{2.13}$$

$$= \sum_j p_j \left|\psi_j\right\rangle \left\langle\psi_j\right|, \tag{2.14}$$

$$\text{where } p_j := \exp(-\beta E_j)/tr(\exp(-\beta H)). \tag{2.15}$$

We assume $H \geq 0$, and its spectrum norm is bounded by $\|H\| \leq poly(n)$. Note that one can always add multiples of identity matrices to H to ensure $H \geq 0$ and this operation does not change the Gibbs state. To ease notation, we will fix H and abbreviate $\rho_\beta(H)$ as ρ_β.

$\kappa_H = poly(n)$ is a power of two that upper bounds $\|H\|$. For example, for local Hamiltonian $H = \sum_{i=1}^m H_i$, $\|H_i\| \leq 1$, one can set κ_H to be the least integer which is a power of two and is greater than m. Then $\kappa_H \leq 2m$.

For simplicity, in this manuscript we assume that we can implement arbitrary 2-qubit gates exactly. Note that this assumption does not influence the generality of

our results, since the error analysis can be easily generalized to the practical case where we approximate arbitrary 2-qubit gate to $1/poly(n)$ precision, by noticing that the l_2 norm $\|(U-V)\,|\psi\rangle\,\|_2$ is bounded by the spectrum norm $\|U-V\|$ for any $|\psi\rangle$.

Notations and Norms.

We use log for $\log_2$. For a complex value $a \in \mathbb{C}$, we use a^* to represent its complex conjugate. For two numbers x, y, we use δ_{xy} to denote the function which equals to 1 if $x = y$ and 0 otherwise. For a matrix M, we use $M^\dagger$ to denote its complex conjugate transpose. For two matrices M, N, we use $\{M, N\}_+$ to denote their anti-commutator: $MN + NM$. We use $\|M\|$ to denote the spectrum norm of M. For a vector $|\phi\rangle$ we use $\|\,|\phi\rangle\,\|_2$ to denote the l_2 norm. $|\phi\rangle$ can be normalized or unnormalized. When it is necessary, we will use number subscripts to denote the name of the quantum registers. For example, $|\phi\rangle_1$ means the state is in register 1.

We use $\Xi(m)$ to denote the set of linear operators on an m-qubit Hilbert space. We use $\mathcal{H}(m)$ to denote the set of *Hermitian* linear operators on an m-qubit Hilbert space. We say $\rho \in \mathcal{H}(m)$ is an m-qubit quantum state if $\rho \geq 0, \rho = \rho^\dagger$ and $tr(\rho) = 1$. We use I_m to denote the identity matrix on m qubits. When m is clear we abbreviate I_m as I. Given a set of linear operators $S = \{M_1, M_2, ...\} \subseteq \Xi(m)$, the algebra generated by S is the set of linear operators which is a finite sum $\sum_k \alpha_k P_k$, where $\alpha_k \in \mathbb{C}$ and P_k is a product of finite operators in S.

We use symbols $\mathcal{R}, \mathcal{E}, \mathcal{F}...$ to represent linear maps from $\Xi(m)$ to $\Xi(m)$. We use $\mathcal{I}$ to denote the identity map. We say a linear map $\mathcal{E} : \mathcal{H}(m) \to \mathcal{H}(m)$ is Completely Positive and Trace Preserving (CPTP) if there exists a set of linear operators $\{A_u\}_u$ such that $\forall M \in \mathcal{H}(m), \mathcal{E}(M) = \sum_u A_u M A_u^\dagger$, where $\sum_u A_u^\dagger A_u = I$.

The trace norm of $M \in \mathcal{H}(m)$ is defined to be $|M|_1 := tr(\sqrt{M^\dagger M})$. The trace norm induces a norm on linear maps $\mathcal{R} : \mathcal{H}(m) \to \mathcal{H}(m)$, which quantifies how much $\mathcal{R}$ can scale the trace norm:

$$|\mathcal{R}|_\star := \max_{M \in \mathcal{H}(m); |M|_1 = 1} |\mathcal{R}(M)|_1 \,. \tag{2.16}$$

2.4 Quantum Metropolis Algorithm in More Detail

The algorithm takes in seven input parameters:

- An n-qubit local Hamiltonian H.
- Inverse temperature β.

- τ, which is a small $1/poly(n)$ real value that controls the weak measurement.
- $K \in \mathbb{N}$, which is the number of iterations in the main loop.
- ρ, which is an arbitrary initial state.
- $r, g \in \mathbb{N}$, which controls the precision of QPE.

The jump operators $\{C_j\}_j$, the acceptance operator W, and the Boosted Quantum Phase Estimation (BQPE, FBQPE) are described in Section 2.2. For QPE $\in$ {BQPE, FBQPE}, we use $\text{QPE}_{a,b}$ for applying QPE on register a, b of n and gr qubits respectively.

The pseudo-code for the main algorithm is given in Algorithm 1. All measurements are done in the computational basis. The outline of each iteration is given in Figure 2.1.

Algorithm 1 QMetropolis(H,β,τ,K,ρ,r,g)

1: Initialize Register 1 to ρ. {For example one can set $\rho = |0^n\rangle\langle 0^n|$} $iter = 1$ to K
2: QPE $\leftarrow$ {BQPE, FBQPE} (with equal probability.)
3: Sample $C \leftarrow \mu$,
4: Append (fresh) Registers 2, 3, 4 in state $|0^{gr}\rangle |0^{gr}\rangle |0\rangle$,
5: Define $U := \text{QPE}_{1,3} \circ C \circ \text{QPE}_{1,2}$.
6: Apply U, then apply W
7: Measure register 4 to get Outcome1 (Outcome1 = 1)
8: **Case** Accept: do nothing. (Outcome1 = 0)
9: Apply($W^\dagger$), and measure register 4 to get Outcome2. (Outcome2 = 1)
10: **Case** AltAccept: do nothing. (Outcome2 = 0)
11: **Case** Reject: Apply $U^\dagger$
12: Trace out (throw away) registers 2, 3, 4.

To illustrate the result of each step of the algorithm, we give explicit formulas for the contents of the registers throughout one iteration. Assume that the version of QPE chosen is BQPE. Let C be the chosen jump operator, which can be expressed in the energy eigenbasis of H as

$$C|\psi_j\rangle = \sum_k c_{jk} |\psi_k\rangle . \tag{2.17}$$

Assume we begin with the state $|\psi_j\rangle |0^{gr}\rangle |0^{gr}\rangle |0\rangle$, where $|\psi_j\rangle$ is an eigenstate of H. Recall that

$$\text{BQPE}|\psi_j\rangle |0^{gr}\rangle = |\psi_j\rangle \sum_{E \in S(r)^{\otimes g}} \beta_{jE} |E\rangle .$$

Then:

$$\mathrm{BQPE}_{1,2}\left|\psi_j\right\rangle|0^{gr}\rangle|0^{gr}\rangle|0\rangle=\sum_{E\in S(r)^{\otimes g}}\beta_{jE}\left|\psi_j\right\rangle|E\rangle|0^{gr}\rangle|0\rangle\,, \tag{2.18}$$

$$C\cdot\mathrm{BQPE}_{1,2}\left|\psi_j\right\rangle|0^{gr}\rangle|0^{gr}\rangle|0\rangle=\sum_{k;E\in S(r)^{\otimes g}}\beta_{jE}\cdot c_{jk}\left|\psi_k\right\rangle|E\rangle|0^{gr}\rangle|0\rangle\,, \tag{2.19}$$

$$\mathrm{BQPE}_{1,3}\cdot C\cdot\mathrm{BQPE}_{1,2}\left|\psi_j\right\rangle|0^{gr}\rangle|0^{gr}\rangle|0\rangle=\sum_{k;E,E'\in S(r)^{\otimes g}}\beta_{jE}\cdot c_{jk}\cdot\beta_{kE'}\left|\psi_k\right\rangle|E\rangle|E'\rangle|0\rangle\,. \tag{2.20}$$

Finally, when W is applied, the result is

$$\sum_{k;E,E'\in S(r)^{\otimes g}}\beta_{jE}\cdot c_{jk}\cdot\beta_{kE'}\left|\psi_k\right\rangle|E\rangle|E'\rangle\left(\sqrt{1-\tau^2 f_{EE'}}\,|0\rangle+\tau\sqrt{f_{EE'}}\,|1\rangle\right). \tag{2.21}$$

Note that if FBQPE is chosen instead of BQPE, the result is

$$\sum_{k;E,E'\in S(r)^{\otimes g}}\beta^*_{jE}\cdot c_{jk}\cdot\beta^*_{kE'}\left|\psi_k\right\rangle|E\rangle|E'\rangle\left(\sqrt{1-\tau^2 f_{EE'}}\,|0\rangle+\tau\sqrt{f_{EE'}}\,|1\rangle\right). \tag{2.22}$$

Then for the state in Eq. (2.21) we measure register 4 in computational basis. If we get measurement outcome 1, the (unnormalized) state becomes

$$\sum_{k;E,E'\in S(r)^{\otimes g}}\tau\sqrt{f_{EE'}}\beta_{jE}\cdot c_{jk}\cdot\beta_{kE'}\left|\psi_k\right\rangle|E\rangle|E'\rangle|1\rangle\,. \tag{2.23}$$

2.5 Main Theorems and the Effective Quantum Markov Chain

In this section, we state formal versions of the main theorems that we will prove about the performance of Algorithm 1. Note that each iteration in Algorithm 1 corresponds to a quantum channel which maps n-qubit states to n-qubit states. We denote this quantum channel as $\mathcal{E}[\tau]$. We will expand $\mathcal{E}[\tau]$ as power series of τ. The performance of Algorithm 1 can be analyzed by studying the term of order $\sim\tau^2$, which is $\mathcal{L}$ defined below. To ease notation, define

$$r_{\beta H}:=1+\log\kappa_H+\log\beta. \tag{2.24}$$

Theorem 4 *For any n-qubit state ρ,*

$$\mathcal{E}[\tau](\rho)=\left(\mathcal{I}+\tau^2\mathcal{L}+\tau^4\mathcal{J}[\tau]\right)(\rho), \tag{2.25}$$

where $\mathcal{L},\mathcal{J}[\tau]:\mathcal{H}(n)\to\mathcal{H}(n)$ are linear maps on operators. $\mathcal{L}$ is independent of τ. Assuming $r\geq r_{\beta H}$ we have

- ***(Fixed point)*** $\left|\mathcal{L}(\rho_\beta)\right|_1 \leq 2^{-g/10+2n+4} + 40\beta \cdot \kappa_H \cdot 2^{-r}$.
- ***(Error terms)*** $|\mathcal{J}[\tau]|_\star \leq 4$.
- ***(Uniqueness and Relaxation)*** *There is a unique* $\rho_{\mathcal{L}}$ *such that* $\mathcal{L}(\rho_{\mathcal{L}}) = 0$. *Besides,* $\rho_{\mathcal{L}}$ *is a full-rank quantum state and for any quantum state* ρ,

$$\lim_{t\to\infty} e^{t\mathcal{L}}(\rho) = \rho_{\mathcal{L}}. \tag{2.26}$$

To make $\left|\mathcal{L}(\rho_\beta)\right|_1 \leq \delta$ parameters g and r are chosen so that $g = O(n + \log 1/\delta)$ and $r = O(\log\beta + \log\kappa_H + \log 1/\delta)$. The constant in the bound of $\left|\mathcal{L}(\rho_\beta)\right|_1$ might be improved by a finer analysis. To ease notation, we will abbreviate $\mathcal{E}[\tau]$ as $\mathcal{E}$.

Theorem 4 suggests that Algorithm 1 effectively approximates a continuous-time chain $e^{t\mathcal{L}}$ with a step-size of τ^2. The divergence between the output of our algorithm and ρ_β depends on the mixing time of $\mathcal{L}$, which depends on the choice of the jump operators $\{C_j\}_j$.

Definition 5 (Mixing time) *Let* ϵ *be a precision parameter. The mixing time w.r.t* $(\mathcal{L}, \epsilon)$ *is defined to be the time needed for driving any initial state* ϵ*-close to its fixed point*

$$t_{mix}(\mathcal{L}, \epsilon) := \inf\{t \geq 0 : \left|e^{t\mathcal{L}}(\rho) - \rho_{\mathcal{L}}\right|_1 \leq \epsilon, \text{ for any quantum state } \rho\}. \tag{2.27}$$

Theorem 6 (Error bounds w.r.t Mixing time) *Let* τ, ϵ *be parameters. Assume* $r \geq r_{\beta H}$. *For integer*[1] $K := t_{mix}(\mathcal{L}, \epsilon)/\tau^2$, *we have for any quantum state* ρ,

$$\left|\mathcal{E}^K(\rho) - \rho_\beta\right|_1 \leq 2\epsilon + \left(2^{-g/10+2n+4} + 40\beta \cdot \kappa_H \cdot 2^{-r} + 2e^4\tau^2\right) t_{mix}(\mathcal{L}, \epsilon).$$

Abbreviate $t_{mix}(\mathcal{L}, \epsilon)$ as t_{mix}. For parameters

$$\begin{aligned}
&\tau^2 = O(\epsilon/t_{mix}) \\
&g = O(n + \log t_{mix} + \log 1/\epsilon) \\
&r = O(\log\beta + \log\kappa_H + \log t_{mix} + \log 1/\epsilon). \\
&K = t_{mix}/\tau^2 = O(t_{mix}^2/\epsilon)
\end{aligned}$$

we have $\left|\mathcal{E}^K(\rho) - \rho_\beta\right|_1 \leq 3\epsilon$. The total runtime of the algorithm is

$$K \cdot 4g \cdot poly(2^r, n) = O\left(poly(t_{mix}, \beta, \kappa_H, 1/\epsilon, n)\right),$$

[1]For simplicity, here we assume $K := t_{mix}(\mathcal{L}, \epsilon)/\tau^2$ is an integer. Otherwise we set K to be the least integer which is greater than $t_{mix}(\mathcal{L}, \epsilon)/\tau^2$ and the error bounds can be analyzed similarly.

which is $poly(n, 1/\epsilon)$ assuming $t_{mix} = poly(n)$, $\beta \leq poly(n)$ and $\kappa_H \leq poly(n)$. The $poly(2^r, n)$ in the above formula is mainly the cost of implementing one run of standard Quantum Phase estimation. Theorem 4 and Theorem 6 will be proved in Section 2.8.

Instead of mixing time, as in the classical Metropolis algorithm, one can also bound the error $\left|\mathcal{E}^K(\rho) - \rho_\beta\right|_1$ in terms of the spectral gap of $\mathcal{L}$. More precisely, since $\mathcal{L}$ may not be Hermitian, we need to define a symmetrized version of $\mathcal{L}$ in order for the spectral gap to be well-defined. Note that $\sigma := \rho_{\mathcal{L}}^{-1}$ is well-defined since $\rho_{\mathcal{L}}$ is full rank. Define $\mathcal{L}^*$ to be dual map w.r.t inner product

$$\langle M, N\rangle_\sigma := tr(\sigma^{\frac{1}{2}} M^\dagger \sigma^{\frac{1}{2}} N).$$

Define the symmetrized map $\mathcal{L}^{(s)} = \frac{1}{2}(\mathcal{L}+\mathcal{L}^*)$. Then by definition $\mathcal{L}^{(s)}$ is Hermitian w.r.t. $\langle, \rangle_\sigma$ and is diagonalizable, thus its spectral gap, denoted as Υ, is well-defined. Furthermore one can prove $\mathcal{L}^{(s)}$ has a unique fixed point thus the spectral gap Υ is strictly greater than 0. The following Theorem 7 implies t_{mix} can be bounded in terms of Υ. Thus one can translate Theorem 6 in terms of Υ to get Corollary 9. For completeness, we put a more detailed explanation of the dual map $\mathcal{L}^*$, the symmetrized map $\mathcal{L}^{(s)}$, their properties and a proof for Theorem 7 in Appendix 2.11.

Theorem 7 (Bounding mixing time w.r.t spectral gap) *Define $\sigma := \rho_{\mathcal{L}}^{-1}$. For any quantum state ρ, we have*

$$\left|e^{t\mathcal{L}}(\rho) - \rho_{\mathcal{L}}\right|_1 \leq 2^{n/2} \cdot \sqrt{tr(\sigma^{\frac{1}{2}}\rho\sigma^{\frac{1}{2}}\rho)} \cdot \exp(-\Upsilon t).$$

Corollary 8 *Define $\sigma := \rho_{\mathcal{L}}^{-1}$, we have*

$$t_{mix}(\mathcal{L}, \epsilon) \leq \frac{1}{\Upsilon}\left(\ln\frac{1}{\epsilon} + \frac{n\ln 2}{2} + \frac{1}{2}\ln tr(\sigma^{\frac{1}{2}}\rho\sigma^{\frac{1}{2}}\rho)\right).$$

Corollary 9 (Error bounds w.r.t Spectral gap) *Let τ, ϵ be two parameters. Let Υ be the spectral gap of $\mathcal{L}^{(s)}$. Assume $r \geq r_{\beta H}$. Define $\sigma := \rho_{\mathcal{L}}^{-1}$. Then for any quantum state ρ, we have*

$$\begin{aligned}&\left|\mathcal{E}^K(\rho) - \rho_\beta\right|_1 \\ &\leq 2\epsilon + \left(2^{-g/10+2n+4} + 40\beta\cdot\kappa_H\cdot 2^{-r} + 2e^4\tau^2\right)\frac{1}{\Upsilon}\left(\ln\frac{1}{\epsilon} + \frac{n\ln 2}{2} + \frac{1}{2}\ln tr(\sigma^{\frac{1}{2}}\rho\sigma^{\frac{1}{2}}\rho)\right),\end{aligned} \tag{2.28}$$

$$\text{for } K := \frac{1}{\tau^2}\frac{1}{\Upsilon}\left(\ln\frac{1}{\epsilon} + \frac{n\ln 2}{2} + \frac{1}{2}\ln tr(\sigma^{\frac{1}{2}}\rho\sigma^{\frac{1}{2}}\rho)\right). \tag{2.29}$$

Outline of this section. In Section 2.5 we will define the operators corresponding to the three cases of the algorithm. In Section 2.5 to Section 2.5 we will derive the evolution equation Eq. (2.25). In particular, we separate $\mathcal{E}$ into a sum of terms according to their dependence on τ. This defines the operator $\mathcal{L}$ which we will analyze in later sections. Subsection 2.6 then writes $\mathcal{L}_C$ in the Lindbladian form. In Section 2.6 we prove the fixed point of $\mathcal{L}$ is unique.

Definition of Operators for the Three Cases

We will use the subscript C to denote a particular choice for $\{C, \text{QPE}\}$. So, for example,

$$U_C = \text{QPE}_{1,3} \circ C \circ \text{QPE}_{1,2}. \tag{2.30}$$

We use (Δ_0, Δ_1) to denote measurement on register 4 in the computational basis, where

$$\Delta_0 := I_n \otimes I_{gr} \otimes I_{gr} \otimes |0\rangle\langle 0|, \tag{2.31}$$

$$\Delta_1 := I_n \otimes I_{gr} \otimes I_{gr} \otimes |1\rangle\langle 1|. \tag{2.32}$$

If the state at the beginning of an iteration is ρ, then the operations performed on $\rho \otimes \left|0^{2gr+1}\right\rangle\left\langle 0^{2gr+1}\right|$ in each of the three cases (before the last three registers are traced out) can be summarized as

$$\textsc{Accept}: \quad O_{a,C} := \Delta_1 W \Delta_0 \circ U_C \tag{2.33}$$

$$\textsc{AltAccept}: \quad O_{b,C} := \Delta_1 W^\dagger \Delta_0 \circ \Delta_0 W \Delta_0 \circ U_C \tag{2.34}$$

$$\textsc{Reject}: \quad O_{r,C} := U_C^\dagger \circ \Delta_0 W^\dagger \Delta_0 \circ \Delta_0 W \Delta_0 \circ U_C \tag{2.35}$$

The initial Δ_0 is added in for symmetry and has no effect since the last register is always initialized as $|0\rangle$ at the start of each iteration. For $s \in \{a, b, r\}$ which represents Accept, AltAccept, and Reject, the corresponding operator which includes tracing out the last three registers is:

$$\mathcal{E}_C^{(s)}(\rho) = tr_{2,3,4}\left(O_{s,C} \quad \left[\rho \otimes \left|0^{2gr+1}\right\rangle\left\langle 0^{2gr+1}\right|\right] \quad O_{s,C}^\dagger\right), \tag{2.36}$$

where $\mathcal{E} = \mathcal{E}^{(a)} + \mathcal{E}^{(b)} + \mathcal{E}^{(r)}$ is the operator representing one iteration. In each of the next three subsections, we will derive alternative expressions for the operators

in the three cases, as a sum of terms with different dependencies on the parameter τ. Recall the definition of W:

$$W := \sum_{E,E' \in S(r)^{\otimes g}} |EE'\rangle \langle EE'| \otimes \begin{bmatrix} \sqrt{1-\tau^2 f_{EE'}} & \tau\sqrt{f_{EE'}} \\ \tau\sqrt{f_{EE'}} & -\sqrt{1-\tau^2 f_{EE'}} \end{bmatrix}. \tag{2.37}$$

Note that W is Hermitian.

Operator for the Accept Case

Definition 10 ***[Operators for the Accept Case]***

$$W^{(10)} := \sum_{E,E' \in S(r)^{\otimes g}} \sqrt{f_{EE'}} |EE'\rangle \langle EE'| \otimes |1\rangle \langle 0| \tag{2.38}$$

$$\mathcal{M}_C^{(a)}(\rho) := tr_{2,3,4}\left(W^{(10)} U_C \quad \left[\rho \otimes \left|0^{2gr+1}\right\rangle \left\langle 0^{2gr+1}\right|\right] \quad U_C^\dagger (W^{(10)})^\dagger\right). \tag{2.39}$$

Lemma 11

$$\mathcal{E}_C^{(a)}(\rho) = \tau^2 \mathcal{M}_C^{(a)} \text{ where } \left|\mathcal{M}_C^{(a)}\right|_\star \leq 1.$$

Proof: The main observation in proving the lemma is that $\Delta_1 W \Delta_0 = \tau W^{(10)}$. Therefore

$$\mathcal{E}_C^{(a)}(\rho) = tr_{2,3,4}\left(O_{a,C} \quad \left[\rho \otimes \left|0^{2gr+1}\right\rangle \left\langle 0^{2gr+1}\right|\right] \quad O_{a,C}^\dagger\right) \tag{2.40}$$

$$= tr_{2,3,4}\left(\Delta_1 W \Delta_0 \circ U_C \quad \left[\rho \otimes \left|0^{2gr+1}\right\rangle \left\langle 0^{2gr+1}\right|\right] \quad U_C^\dagger \circ \Delta_0 W^\dagger \Delta_1\right) \tag{2.41}$$

$$= \tau^2 \cdot tr_{2,3,4}\left(W^{(10)} \circ U_C \quad \left[\rho \otimes \left|0^{2gr+1}\right\rangle \left\langle 0^{2gr+1}\right|\right] \quad U_C^\dagger \circ (W^{(10)})^\dagger\right) \tag{2.42}$$

$$= \tau^2 \mathcal{M}_C^{(a)}(\rho). \tag{2.43}$$

Using Lemma 37 in Appendix 2.10, we can observe that $\left|\mathcal{M}_C^{(a)}\right|_\star \leq 1$. ∎

Operator for the AltAccept Case

Lemma 12

$$\mathcal{E}_C^{(b)}(\rho) = \tau^2 \mathcal{M}_C^{(a)} + \tau^4 \mathcal{J}_C^{(b)}[\tau],$$

where $\left|\mathcal{J}_C^{(b)}[\tau]\right|_\star \leq 3$ *for* $\tau \in [0,1]$.

Proof: Define the function $W^{(00)}$ to be

$$W^{(00)} = \sum_{E,E' \in S(r)^{\otimes g}} \frac{f_{EE'}}{\sqrt{1-\tau^2 f_{EE'}} + 1} |EE'\rangle \langle EE'| \otimes |0\rangle \langle 0|.$$

Note that $f_{EE'}/(\sqrt{1-\tau^2 f_{EE'}}+1) \in [0,1]$ since both τ and $f_{EE'}$ are in $[0,1]$. With this definition in place, observe that

$$\Delta_0 W \Delta_0 = \Delta_0 - \tau^2 W^{(00)},$$

and

$$\Delta_1 W^\dagger \Delta_0 \circ \Delta_0 W \Delta_0 = \tau W^{(10)}(\Delta_0 - \tau^2 W^{(00)}) = \tau W^{(10)} - \tau^3 W^{(10)} W^{(00)}.$$

Now we can express $\mathcal{E}_C^{(b)}(\rho)$ as

$$tr_{2,3,4}\left(\left(\tau W^{(10)} - \tau^3 W^{(10)} W^{(00)}\right) U_C \quad \left[\rho \otimes \left|0^{2gr+1}\right\rangle \left\langle 0^{2gr+1}\right|\right] \quad U_C^\dagger \left(\tau W^{(10)} - \tau^3 W^{(10)} W^{(00)}\right)^\dagger\right).$$

In multiplying out the terms, there is one τ^2 term:

$$\tau^2 tr_{2,3,4}\left(W^{(10)} U_C \quad \left[\rho \otimes \left|0^{2gr+1}\right\rangle \left\langle 0^{2gr+1}\right|\right] \quad U_C^\dagger (W^{(10)})^\dagger\right),$$

which is equal to $\tau^2 \mathcal{M}_C^{(a)}$. There are two τ^4 terms:

$$-\tau^4 tr_{2,3,4}\left(W^{(10)} W^{(00)} U_C \quad \left[\rho \otimes \left|0^{2gr+1}\right\rangle \left\langle 0^{2gr+1}\right|\right] \quad U_C^\dagger (W^{(10)})^\dagger\right),$$

$$-\tau^4 tr_{2,3,4}\left(W^{(10)} U_C \quad \left[\rho \otimes \left|0^{2gr+1}\right\rangle \left\langle 0^{2gr+1}\right|\right] \quad U_C^\dagger (W^{(10)} W^{(00)})^\dagger\right).$$

Finally, there is one τ^6 term:

$$\tau^6 tr_{2,3,4}\left(W^{(10)} W^{(00)} U_C \quad \left[\rho \otimes \left|0^{2gr+1}\right\rangle \left\langle 0^{2gr+1}\right|\right] \quad U_C^\dagger (W^{(10)} W^{(00)})^\dagger\right).$$

The sum of the two τ^4 term and the τ^6 term is denoted as $\tau^4 \cdot \mathcal{J}_C^{(b)}[\tau]$. Using Lemma 37 in Appendix 2.10, we can observe that $\left|\mathcal{J}_C^{(b)}[\tau]\right|_\star \leq 3$ for $\tau \in [0,1]$. ■

Operator for the Reject Case

Definition 13 ***[Operators for the Reject Case]***

$$W^{(000)} := \sum_{E,E' \in S(r)^{\otimes g}} f_{EE'} \left|EE'\right\rangle \left\langle EE'\right| \otimes \left|0\right\rangle \left\langle 0\right| = (W^{(10)})^\dagger W^{(10)} \tag{2.44}$$

$$\mathcal{M}_C^{(r)}(\rho) := \left\langle 0^{2gr+1}\right| U_C^\dagger W^{(000)} U_C \left|0^{2gr+1}\right\rangle \cdot \rho + \rho \cdot \left\langle 0^{2gr+1}\right| U_C^\dagger W^{(000)} U_C \left|0^{2gr+1}\right\rangle \tag{2.45}$$

Lemma 14

$$\mathcal{E}_C^{(r)}(\rho) = \mathcal{I} - \tau^2 \mathcal{M}_C^{(r)} + \tau^4 \mathcal{J}_C^{(r)},$$

where $\left|\mathcal{M}_C^{(r)}\right|_\star \leq 2$ *and* $\left|\mathcal{J}_C^{(r)}\right|_\star \leq 1$.

Proof: First observe that

$$\Delta_0 W \Delta_0 \circ \Delta_0 W \Delta_0 = \Delta_0 - \tau^2 W^{(000)}.$$

Also $W^{(000)}$ is Hermetian. Now we can express $\mathcal{E}_C^{(r)}(\rho)$ as

$$tr_{2,3,4}\left(U_C^\dagger\left(\Delta_0 - \tau^2 W^{(000)}\right)U_C \quad \left[\rho \otimes \left|0^{2gr+1}\right\rangle\left\langle 0^{2gr+1}\right|\right] \quad U_C^\dagger\left(\Delta_0 - \tau^2 W^{(000)}\right)^\dagger U_C\right).$$

There is one term independent of τ which is the identity since Δ_0 and U_C commute:

$$tr_{2,3,4}\left(U_C^\dagger \Delta_0 U_C \quad \left[\rho \otimes \left|0^{2gr+1}\right\rangle\left\langle 0^{2gr+1}\right|\right] \quad U_C^\dagger \Delta_0 U_C\right) = \mathcal{I}(\rho) = \rho.$$

There are two τ^2 terms:

$$-\tau^2 tr_{2,3,4}\left(U_C^\dagger W^{(000)} U_C \quad \left[\rho \otimes \left|0^{2gr+1}\right\rangle\left\langle 0^{2gr+1}\right|\right]\right) = -\tau^2\left\langle 0^{2gr+1}\right| U_C^\dagger W^{(000)} U_C \left|0^{2gr+1}\right\rangle \rho,$$

$$-\tau^2 tr_{2,3,4}\left(\quad \left[\rho \otimes \left|0^{2gr+1}\right\rangle\left\langle 0^{2gr+1}\right|\right] U_C^\dagger W^{(000)} U_C\right) = -\tau^2 \rho \cdot \left\langle 0^{2gr+1}\right| U_C^\dagger W^{(000)} U_C \left|0^{2gr+1}\right\rangle.$$

The sum of the τ^2 terms is equal to $-\tau^2 \mathcal{M}_C^{(r)}(\rho)$. Finally there is a τ^4:

$$\tau^4 tr_{2,3,4}\left(U_C^\dagger W^{(000)} U_C \quad \left[\rho \otimes \left|0^{2gr+1}\right\rangle\left\langle 0^{2gr+1}\right|\right] \quad U_C^\dagger W^{(000)} U_C\right).$$

This last term is defined to be $\mathcal{J}_C^{(r)}$. Using Lemma 37 in Appendix 2.10, we can observe that $\left|\mathcal{M}_C^{(r)}\right|_\star \leq 2$ and $\left|\mathcal{J}_C^{(r)}\right|_\star \leq 1$ for $\tau \in [0,1]$. ■

The Definition of Operator $\mathcal{L}_C$ and $\mathcal{L}$

Definition 15 ***[The Operators $\mathcal{L}_C, \mathcal{J}_C$ and $\mathcal{L}, \mathcal{J}$]***

$$\mathcal{L}_C := 2\mathcal{M}_C^{(a)} - \mathcal{M}_C^{(r)} \tag{2.46}$$

$$\mathcal{J}_C[\tau] := \mathcal{J}_C^{(b)}[\tau] + \mathcal{J}_C^{(r)} \tag{2.47}$$

Averaging the random choices for QPE and the jump operator C,

$$\mathcal{L} := \sum_{C=\{C,QPE\}} \frac{1}{2}\mu(C)\mathcal{L}_C, \tag{2.48}$$

where the $\frac{1}{2}$ comes from the fact that QPE is chosen uniformly from $\{BQPE, FBQPE\}$. Define $\mathcal{M}^{(a)}, \mathcal{M}^{(r)}$ and $\mathcal{J}[\tau]$ from $\mathcal{M}_C^{(a)}, \mathcal{M}_C^{(r)}$ and $\mathcal{J}_C[\tau]$ similarly.

Lemma 16

$$\mathcal{E} = \mathcal{I} + \tau^2\mathcal{L} + \tau^4\mathcal{J}[\tau], \tag{2.49}$$

where $|\mathcal{L}|_\star \leq 4$, and $|\mathcal{J}[\tau]|_\star \leq 4$ for $\tau \in [0,1]$.

Proof: By Lemmas 11, 12, and 14,

$$\begin{aligned}
\mathcal{E}_C &= \mathcal{E}_C^{(a)} + \mathcal{E}_C^{(b)} + \mathcal{E}_C^{(r)} && (2.50)\\
&= \tau^2 \mathcal{M}_C^{(a)} + (\tau^2 \mathcal{M}_C^{(a)} + \tau^4 \mathcal{J}_C^{(b)}[\tau]) + (\mathcal{I} - \tau^2 \mathcal{M}_C^{(r)} + \tau^4 \mathcal{J}_C^{(r)}) && (2.51)\\
&= \mathcal{I} + \tau^2 (2\mathcal{M}_C^{(a)} - \mathcal{M}_C^{(r)}) + \tau^4 (\mathcal{J}_C^{(b)}[\tau] + \mathcal{J}_C^{(r)}) && (2.52)\\
&= \mathcal{I} + \tau^2 \mathcal{L}_C + \tau^4 \mathcal{J}_C[\tau]. && (2.53)
\end{aligned}$$

Since $\mathcal{J}_C = \mathcal{J}_C^{(b)}[\tau] + \mathcal{J}_C^{(r)}$, $\left|\mathcal{J}_C^{(b)}[\tau]\right|_\star \leq 3$, and $\left|\mathcal{J}_C^{(r)}\right|_\star \leq 1$, the bound on $|\mathcal{J}_C[\tau]|_\star$ follows by triangle inequality. Similarly $|\mathcal{L}_C|_\star \leq 2\left|\mathcal{M}_C^{(a)}\right|_\star + \left|\mathcal{M}_C^{(r)}\right|_\star \leq 4$. The equation for $\mathcal{E}$ comes from linearity. The bounds for $|\mathcal{J}[\tau]|_\star$ and $|\mathcal{L}|_\star$ come from triangle inequality. ∎

2.6 Uniqueness of the Fixed Point

In this section, we rewrite $\mathcal{L}$ in the Lindbladian form, and prove that $\mathcal{L}$ has a unique fixed point if the jump operators $\{C_j\}_j$ generate the full algebra. This section is independent of Section 2.7 which proves $\mathcal{L}$ approximately fixes the Gibbs state, and can be skipped temporarily.

Lindbladian form of $\mathcal{L}$

Lemma 17 *$\mathcal{L}_C$ can be written in Lindbladian form, that is defining*

$$S_C(EE'z) := \langle EE'z | \, W^{(10)} \circ U_C \left| 0^{2gr+1} \right\rangle$$

we have

$$\mathcal{L}_C(\rho) = \sum_{EE'z} 2 \cdot S_C(EE'z) \cdot \rho \cdot S_C(EE'z)^\dagger - \left\{ S_C(EE'z)^\dagger S_C(EE'z), \rho \right\}_+ . \tag{2.54}$$

Proof: One can check that

$$\begin{aligned}
\mathcal{L}_C(\rho) &= 2\mathcal{M}_C^{(a)}(\rho) - \mathcal{M}_C^{(r)}(\rho) && (2.55)\\
&= \sum_{EE'z} 2 \cdot S_C(EE'z) \cdot \rho \cdot S_C(EE'z)^\dagger && (2.56)\\
&\quad - \sum_{EE'z} \left(S_C(EE'z)^\dagger \cdot S_C(EE'z) \cdot \rho + \rho \cdot S_C(EE'z)^\dagger S_C(EE'z) \right) && (2.57)\\
&= \sum_{EE'z} 2 \cdot S_C(EE'z) \cdot \rho \cdot S_C(EE'z)^\dagger - \left\{ S_C(EE'z)^\dagger S_C(EE'z), \rho \right\}_+ && (2.58)
\end{aligned}$$

∎

In the following we give a sketch that $\mathcal{L}_C$ and the Davies generator $\mathcal{D}_\alpha(w)$ have similar forms. This observation is just for intuition and will not be used in any proof. Due to the indeterministic and imperfect energy estimation of BQPE, it is unclear whether the proof techniques used for Davies-generator-based Gibbs sampler [Che+23; DLL24] can be adapted to show that our $\mathcal{L}$ satisfying $\mathcal{L}(\rho_\beta) \approx 0$. These proof techniques [Che+23; DLL24] are based on bounding the approximation error by truncating the infinite integral in the (weighted) Davies generator to a finite region.

Recall that the canonical form of Davies generator $\mathcal{D}$ w.r.t. jump operators $\{A_\alpha\}_\alpha$, in the Schrodinger picture, is given by

$$\mathcal{D}(\rho) = -i[H, \rho] + \sum_{w,\alpha} \mathcal{D}_\alpha(w)(\rho) \tag{2.59}$$

$$\mathcal{D}_\alpha(w)(\rho) = G^\alpha(w) \left(2 \cdot A_\alpha(w) \cdot \rho \cdot A_\alpha(w)^\dagger - \left\{A_\alpha(w)^\dagger A_\alpha(w), \rho\right\}_+\right) \tag{2.60}$$

$$A_\alpha(w) := \int_{-\infty}^{+\infty} e^{iwt} e^{-iHt} A_\alpha e^{iHt} dt. \tag{2.61}$$

Here $G^\alpha(w)$ is the acceptance rate, A_α is the jump operator, and $A_\alpha(w)$ is an operator which maps a state of energy ν to energy $\nu + w$. The summation $\sum_w$ sums over all possible energy difference $\{E_j - E_k\}_{j,k}$. The $S_C(EE'z)$ in our $\mathcal{L}$ is a conceptual analog of $\sqrt{G^\alpha} A_\alpha(w)$, which map states with energy approximately $\overline{E}$ to states with energy approximately $\overline{E'}$. The jump operator C is an analog of A_α.

Uniqueness of the Fixed Point

In this subsection, we prove the following theorem.

Theorem 18 *[Uniqueness of full-rank fixed point] Suppose that in Algorithm 1, the algebra generated by jump operators $\{C_j\}_j$ is equal to the full algebra, that is, the set of all n-qubit operators. Then there is a unique $\rho_{\mathcal{L}} \in \Xi(n)$ such that $\mathcal{L}(\rho_{\mathcal{L}}) = 0$. In addition, $\rho_{\mathcal{L}}$ is a full-rank quantum state, and for any quantum state ρ,*

$$\lim_{t\to\infty} e^{t\mathcal{L}}(\rho) = \rho_{\mathcal{L}}. \tag{2.62}$$

We first prove a Lemma.

Lemma 19 *$e^{t\mathcal{L}}$ is CPTP for any $t \geq 0$.*

Proof: Note that

$$e^{t\mathcal{L}} = \lim_{\delta\to 0} \mathcal{E}[\delta]^{t/\delta^2}, \tag{2.63}$$

where the limit is taking by decreasing $\delta \geq 0$ to 0. Since $\mathcal{E}[\delta]$ is CPTP for any $\delta \in [0,1]$, $e^{t\mathcal{L}}$ is also CPTP. An alternative proof can be obtained by noticing that Lemma 17 implies that $\mathcal{L}$ satisfies Theorem 40 in Appendix 2.11. ∎

We invoke the following Theorems to prove the uniqueness of the fixed point, which can be adapted from Corollary 7.2 of [Wol12] or Lemma 2 in [DLL24].

Theorem 20 ([Wol12; DLL24]) *Suppose $e^{\mathcal{P}t} : \Xi(n) \to \Xi(n)$ is a CPTP map for any $t \geq 0$, with generator*

$$\mathcal{P}(\rho) = -i[H_{system}, \rho] + \sum_{k \in S} \left(V_k \rho V_k^\dagger - \frac{1}{2} \left\{ V_k^\dagger V_k, \rho \right\}_+ \right). \tag{2.64}$$

If the algebra generated by operators $\{V_k\}_k$ is the full algebra $\Xi(n)$. Then there exists a unique $\rho_{\mathcal{P}}$ such that $\mathcal{P}(\rho_{\mathcal{P}}) = 0$. In addition, $\rho_{\mathcal{P}}$ is a full-rank quantum state, and for any quantum state ρ,

$$\lim_{t \to \infty} e^{t\mathcal{P}}(\rho) = \rho_{\mathcal{P}}. \tag{2.65}$$

We use the above Theorem to prove that $\mathcal{L}$ has a unique fixed point.

Proof:[Proof of Theorem 18] By Lemma 19 we know that $e^{t\mathcal{L}}$ is CPTP. Then we verify the conditions in Theorem 20. Recall that in Lemma 17 we have written $\mathcal{L}_C$ in Lindbladian form in terms of $S_C(EE'z)$. By definition

$$\mathcal{L} = \sum_{C=\{C,\mathrm{QPE}\}} \frac{1}{2} \mu(C) \cdot \mathcal{L}_C. \tag{2.66}$$

Define

$$H_{system} = 0 \tag{2.67}$$

$$V_C(EE'z) := \sqrt{\mu(C)} \cdot S_C(EE'z). \tag{2.68}$$

One can check that

$$\mathcal{L} = -i[H_{system}, \rho] + \sum_{C,EE'z} \left(V_C(EE'z) \rho V_C(EE'z)^\dagger - \frac{1}{2} \left\{ V_C(EE'z)^\dagger V_C(EE'z), \rho \right\}_+ \right). \tag{2.69}$$

Recall that in Section 2.4 Eq. (2.23) we have computed

$$\left| \eta_{j\{C,\mathrm{BQPE}\}} \right\rangle := \tau \cdot W^{(10)} \circ U_{C,\mathrm{BQPE}} \left| \psi_j, 0^{2gr+1} \right\rangle \tag{2.70}$$

$$= \sum_{k;E,E' \in S(r)^{\otimes g}} \tau \sqrt{f_{EE'}} \beta_{jE} \cdot c_{jk} \cdot \beta_{kE'} \left| \psi_k \right\rangle \left| E \right\rangle \left| E' \right\rangle \left| 1 \right\rangle. \tag{2.71}$$

One can check that

$$V_{\{C,\mathrm{BQPE}\}}(EE'z) = \sqrt{\mu(C)} \cdot S_C(EE'z) \tag{2.72}$$

$$= \sqrt{\mu(C)} \cdot \langle EE'z| \, W^{(10)} \circ U_{\{C,\mathrm{BQPE}\}} \sum_j |\psi_j\rangle \langle\psi_j| \otimes |0^{2gr+1}\rangle \tag{2.73}$$

$$= \sqrt{\mu(C)} \cdot \sum_j \frac{1}{\tau} \langle EE'z | \eta_{j\{C,\mathrm{BQPE}\}}\rangle \langle\psi_j| \tag{2.74}$$

$$= \sqrt{\mu(C) f_{EE'}} \cdot \delta_{z1} \sum_{j,k} \beta_{jE} \cdot c_{jk} \cdot \beta_{kE'} \, |\psi_k\rangle \langle\psi_j| \tag{2.75}$$

Define a matrix $B_E \in \Xi(n)$ such that

$$B_E |\psi_j\rangle = \beta_{jE} |\psi_j\rangle , \tag{2.76}$$

From Eq. (2.75) one can check that

$$V_{\{C,\mathrm{BQPE}\}}(EE'1) = \sqrt{\mu(C) f_{EE'}} \cdot B_{E'} \cdot C \cdot B_E \tag{2.77}$$

From Eq. (2.170) in Appendix 2.9 we know that

$$\sum_E B_E = I_n, \tag{2.78}$$

$$\sum_{EE'} \frac{1}{\sqrt{\mu(C) f_{EE'}}} V_{\{C,\mathrm{BQPE}\}}(EE'1) = \left(\sum_{E'} B_{E'}\right) \cdot C \cdot \left(\sum_E B_E\right) = C. \tag{2.79}$$

Since the algebra generated by $\{C_j\}_j$ is the full algebra $\Xi(n)$, and thus the algebra generated by $\{V_C(EE'z)\}_{C,E,E',z}$ is the full algebra $\Xi(n)$. We can then use Theorem 20 to complete the proof.

■

2.7 Gibbs States as Approximate Fixed Point

In this section, we will prove $\mathcal{L}$ approximately fixes the Gibbs state, that is

Lemma 21 *If $r \geq r_{\beta H}$, we have*

$$\left|\mathcal{L}(\rho_\beta)\right|_1 \leq 2^{-g/10+2n+4} + 40\beta \cdot \kappa_H \cdot 2^{-r}.$$

The outline of the proof is as follows. In section 2.7 we define a truncated Gibbs states $\rho_{\beta 0}$ where $\left|\rho_\beta - \rho_{\beta 0}\right|_1 \approx 0$, thus $\left|\mathcal{L}(\rho_\beta)\right|_1 \approx \left|\mathcal{L}(\rho_{\beta 0})\right|_1$ since $|\mathcal{L}|_\star$ is bounded.

The remaining subsections then focus on bounding $\left|\mathcal{L}(\rho_{\beta 0})\right|_1$. Then in Section 2.7 and Section 2.7 we define projected operators which will be used as auxiliary notations in the following proofs. In 2.7 we show that $\mathcal{L}(\rho_{\beta 0})$ can be written as the sum of constant number of matrices, where each matrix has small trace norm. Finally in Section 2.7 we complete the proof of Lemma 21.

Truncated Energy and Truncated Gibbs States

Recall that from Section 2.2, that Boosted Quantum Phase Estimation acts as

$$\mathrm{BQPE}\left|\psi_j\right\rangle\left|0^{gr}\right\rangle = \left|\psi_j\right\rangle \sum_{E\in S(r)^{\otimes g}} \beta_{jE}\left|E\right\rangle,$$

where $S(r)$ is the set of energy estimations that are integer multiples of

$$w := \kappa_H \cdot 2^{-r}.$$

We have defined $r_{\beta H} = 1 + \log\kappa_H + \log\beta$, so for $r \geq r_{\beta H}$, we have $2\beta w \leq 1$. For any real value $\nu \geq 0$, $\lfloor\nu\rfloor$ denotes the closet value to ν which is an integer multiple of w and is smaller or equal to ν. For any $k \in \mathbb{N}$, define

$$\nu^{(k)} := \lfloor\nu\rfloor + kw, \tag{2.80}$$

$$\text{in particular } E_j^{(k)} := \lfloor E_j\rfloor + kw. \tag{2.81}$$

Recall that $Z = tr(\exp(-\beta H))$ is the partition function of ρ_β, and $p_j = \exp(-\beta E_j)/Z$ is the corresponding probability. Define the truncated probability and truncated Gibbs states as

$$p_{jk} := \exp\left(-\beta E_j^{(k)}\right)/Z, \qquad \text{for } k \in \mathbb{N}, \tag{2.82}$$

$$\rho_{\beta 0} := \sum_j p_{j0}\left|\psi_j\right\rangle\left\langle\psi_j\right|. \tag{2.83}$$

One can check the following.

Lemma 22 *If $2\beta w \leq 1$, then $|p_j - p_{j0}| \leq p_j \cdot 2\beta w$ and $\left|\rho_{\beta 0} - \rho_\beta\right|_1 \leq 2\beta w$.*

Lemma 16 proves that $|\mathcal{L}|_\star \leq 4$. Using this fact, along with the triangle inequality and Lemma 22, we have the following.

Lemma 23

$$\left|\mathcal{L}(\rho_\beta)\right|_1 \leq \left|\mathcal{L}(\rho_{\beta 0})\right|_1 + \left|\mathcal{L}(\rho_\beta - \rho_{\beta 0})\right|_1 \leq \left|\mathcal{L}(\rho_{\beta 0})\right|_1 + 4\cdot 2\beta w.$$

The remainder of this section focuses on bounding $\left|\mathcal{L}(\rho_{\beta 0})\right|_1$.

Projected BQPE.

From Lemma 3, we know that the median estimation from BQPE almost always maps to one of two possible values: $E_j^{(0)} = \lfloor E_j \rfloor$ or $E_j^{(1)} = \lceil E_j \rceil$. A remark is that the number "two" is not important in the analysis as long as it is a constant. We define the following projections that separate out the cases depending on the output of BQPE:

$$P^{(k)} := \sum_j \left|\psi_j\right\rangle \left\langle\psi_j\right| \otimes \sum_{E \in S(r)^{\otimes g} : \overline{E} = E_j^{(k)}} |E\rangle \langle E|, \text{ for } k \in \{0,1\}, \tag{2.84}$$

$$P^{(else)} := I - P^{(0)} - P^{(1)}. \tag{2.85}$$

To analyze the performance of our algorithm, we decompose BQPE into three operators, according to the above projections. That is,

$$T := \{0, 1, \text{“}else\text{”}\}, \tag{2.86}$$

$$\text{BQPE} = \sum_{k \in T} \text{BQPE}^{(k)}, \tag{2.87}$$

$$\text{where BQPE}^{(k)} := P^{(k)} \cdot \text{BQPE}, \tag{2.88}$$

$$\text{BQPE}^{(k)} \left|\psi_j\right\rangle |0^{gr}\rangle = \left|\psi_j\right\rangle \sum_E \beta_{jE}^{(k)} |E\rangle, \tag{2.89}$$

$$\beta_{jE}^{(k)} := \begin{cases} \beta_{jE}, & \text{if } \overline{E} = E_j^{(k)}, \\ 0, & \text{else,} \end{cases} \quad \text{for } k \in \{0,1\} \tag{2.90}$$

$$\beta_{jE}^{else} := \begin{cases} \beta_{jE}, & \text{if } \overline{E} \notin \{E_j^{(0)}, E_j^{(1)}\}. \\ 0, & \text{else.} \end{cases} \tag{2.91}$$

For convenience, for any subset $A \subseteq T$, we also define

$$P^{(A)} := \sum_{k \in A} P^{(k)}, \tag{2.92}$$

and define $\text{BQPE}^{(A)}$, $\beta_{jE}^{(A)}$ accordingly. For FBQPE we similarly define $\text{FBQPE}^{(k)}$ and $\text{FBQPE}^{(A)}$.

Projected Operators

Recall that the $\mathcal{L}$ for our algorithm is defined as

$$\mathcal{L}_C = 2\mathcal{M}_C^{(a)} - \mathcal{M}_C^{(r)},$$

$$\mathcal{M}_C^{(a)}(\rho) = tr_{2,3,4}\left(W^{(10)} U_C \left[\rho \otimes \left|0^{2gr+1}\right\rangle \left\langle 0^{2gr+1}\right|\right] U_C^\dagger (W^{(10)})^\dagger\right),$$

$$\mathcal{M}_C^{(r)}(\rho) = \left\langle 0^{2gr+1}\right| U_C^\dagger (W^{(10)})^\dagger W^{(10)} U_C \left|0^{2gr+1}\right\rangle \rho + \rho \left\langle 0^{2gr+1}\right| U_C^\dagger (W^{(10)})^\dagger W^{(10)} U_C \left|0^{2gr+1}\right\rangle,$$

where

$$W^{(10)} = \sum_{E,E' \in S(r)^{\otimes g}} \sqrt{f_{EE'}} \left|EE'\right\rangle \left\langle EE'\right| \otimes \left|1\right\rangle \left\langle 0\right|, \tag{2.93}$$

$$U_C = \mathrm{QPE}_{1,3} \circ C \circ \mathrm{QPE}_{1,2}. \tag{2.94}$$

For convenience, we divide $\mathcal{M}_C^{(r)}$ into "right" and "left" terms

$$\mathcal{M}_C^{(rr)}(\rho) = \left\langle 0^{2gr+1}\right| U_C^\dagger (W^{(10)})^\dagger W^{(10)} U_C \left|0^{2gr+1}\right\rangle \rho, \tag{2.95}$$

$$\mathcal{M}_C^{(rl)}(\rho) = \rho \left\langle 0^{2gr+1}\right| U_C^\dagger (W^{(10)})^\dagger W^{(10)} U_C \left|0^{2gr+1}\right\rangle. \tag{2.96}$$

By the following Lemma it suffices to only analyze the right term $\mathcal{M}^{(rr)}$.

Lemma 24 *For any Hermitian ρ,*

$$|\mathcal{L}(\rho)|_1 \leq 2 \left|\mathcal{M}^{(a)}(\rho) - \mathcal{M}^{(rr)}(\rho)\right|_1. \tag{2.97}$$

Proof: Note that $\mathcal{M}^{(a)}(\rho) = \mathcal{M}^{(a)}(\rho)^\dagger$, $\mathcal{M}^{(rl)}(\rho) = \mathcal{M}^{(rr)}(\rho)^\dagger$. Since $\left|N^\dagger\right|_1 = |N|_1$ for any matrix N, we have

$$\left|\mathcal{M}^{(a)}(\rho) - \mathcal{M}^{(rl)}(\rho)\right|_1 = \left|\mathcal{M}^{(a)}(\rho) - \mathcal{M}^{(rr)}(\rho)\right|_1.$$

Thus

$$|\mathcal{L}(\rho)|_1 \leq \left|\mathcal{M}^{(a)}(\rho) - \mathcal{M}^{(rl)}(\rho)\right|_1 + \left|\mathcal{M}^{(a)}(\rho) - \mathcal{M}^{(rr)}(\rho)\right|_1 \leq 2 \left|\mathcal{M}^{(a)}(\rho) - \mathcal{M}^{(rr)}(\rho)\right|_1.$$

■

We can now use the decomposition of BQPE to decompose the accept and reject operators.

Definition 25 (Projected Operators) *For any subsets $A, B \subseteq T$, we define $U_C^{(AB)}$ by substituting QPE with corresponding operators:*

$$U_C^{(AB)} := QPE_{1,3}^{(A)} \circ C \circ QPE_{1,2}^{(B)}. \tag{2.98}$$

Accordingly for subsets $A, B, X, Y \subseteq T$, we define

$$\mathcal{M}_C^{(a,ABXY)}(\rho) := tr_{2,3,4} \left(W^{(10)} U_C^{(AB)} \left[\rho \otimes \left|0^{2gr+1}\right\rangle \left\langle 0^{2gr+1}\right|\right] (U_C^{(XY)})^\dagger (W^{(10)})^\dagger \right) \tag{2.99}$$

$$\mathcal{M}_C^{(rr,ABXY)}(\rho) := \left\langle 0^{2gr+1}\right| (U_C^{(BA)})^\dagger (W^{(10)})^\dagger W^{(10)} U_C^{(YX)} \left|0^{2gr+1}\right\rangle \rho. \tag{2.100}$$

Note that in Eq. (2.100) we use $U_C^{(BA)}, U_C^{(YX)}$ instead of $U_C^{(AB)}, U_C^{(XY)}$.

One can directly check that the following Lemma are true, which we omit the proofs.

Lemma 26

$$\mathcal{M}_C^{(a)} = \sum_{u,v,s,t\in T} \mathcal{M}_C^{(a,uvst)}, \quad \mathcal{M}_C^{(rr)} = \sum_{u,v,s,t\in T} \mathcal{M}_C^{(rr,uvst)}. \tag{2.101}$$

Similarly after averaging over the random selection of C, we get

$$\mathcal{M}^{(a)} = \sum_{u,v,s,t\in T} \mathcal{M}^{(a,uvst)}, \quad \mathcal{M}^{(rr)} = \sum_{u,v,s,t\in T} \mathcal{M}^{(rr,uvst)}. \tag{2.102}$$

Explicit Expressions for Operators in the Accept and Reject Cases

The following two lemmas give explicit representations for the operators $\mathcal{M}^{(a,ABXY)}$ and $\mathcal{M}^{(rr,ABXY)}$ in terms of the energy eigenstates $\{|\psi_j\rangle\}_j$. To ease notations, we write $(\beta_{jE}^{(\cdot)})^*$ as $\beta_{jE}^{(\cdot)*}$.

Lemma 27 *For any subsets $A, B, X, Y \subseteq T$,*

$$\langle\psi_m|\,\mathcal{M}^{(a,ABXY)}(\rho_{\beta 0})\,|\psi_k\rangle = \sum_j p_{j0} \sum_{E,E'} f_{EE'} \sum_C \mu(C) \left(c_{jm}c_{jk}^*\right) \cdot \Re(\beta_{jE}^{(B)}\beta_{jE}^{(Y)*}\beta_{mE'}^{(A)}\beta_{kE'}^{(X)*}),$$

$$\langle\psi_m|\mathcal{M}^{(rr,ABXY)}(\rho_{\beta 0})|\psi_k\rangle = p_{k0} \sum_{j;E,E'} f_{E'E} \sum_C \mu(C) \left(c_{jm}c_{jk}^*\right) \cdot \Re\left(\beta_{jE}^{(B)}\beta_{jE}^{(Y)*}\beta_{mE'}^{(A)}\beta_{kE'}^{(X)*}\right),$$

where $\Re(\alpha) = \frac{1}{2}(\alpha + \alpha^*)$ is the real part of a complex number.

Proof: Recall that

$$\rho_{\beta 0} = \sum_j p_{j0} \left|\psi_j\right\rangle \left\langle\psi_j\right|. \tag{2.103}$$

Suppose that the gates chosen in the algorithm are $C = \{\text{BQPE}, C\}$. Similarly to how we derive Eq. (2.23) in Section 2.4, one can check that

$$W^{(10)}U_C^{(AB)}\left|\psi_j\right\rangle\left|0^{2gr+1}\right\rangle = \sum_{l;E,E'} \sqrt{f_{EE'}}\, c_{jl} \cdot \beta_{jE}^{(B)} \cdot \beta_{lE'}^{(A)} \,|\psi_l\rangle\,|E\rangle\,|E'\rangle\,|1\rangle\,. \tag{2.104}$$

(a) Accept Case:

The whole operator $\mathcal{M}_C^{(a,ABXY)}$ then looks like

$$\begin{aligned}
&\mathcal{M}_C^{(a,ABXY)}(|\psi_j\rangle\langle\psi_j|) \\
=&tr_{2,3,4}\left(W^{(10)}U_C^{(AB)}\left[|\psi_j\rangle\langle\psi_j|\otimes|0^{2gr+1}\rangle\langle 0^{2gr+1}|\right](U_C^{(XY)})^\dagger(W^{(10)})^\dagger\right) \quad (2.105)\\
=&\sum_{l,h;E,E'} f_{EE'}(c_{jl}c_{jh}^*)(\beta_{jE}^{(B)}\beta_{jE}^{(Y)*})(\beta_{lE'}^{(A)}\beta_{hE'}^{(X)*})\,|\psi_l\rangle\langle\psi_h|\,. \quad (2.106)
\end{aligned}$$

Then

$$\langle\psi_m|\mathcal{M}_C^{(a,ABXY)}\left[\rho_{\beta 0}\right]|\psi_k\rangle = \sum_j p_{j0}\sum_{E,E'} f_{EE'}(c_{jm}c_{jk}^*)(\beta_{jE}^{(B)}\beta_{jE}^{(Y)*})(\beta_{mE'}^{(A)}\beta_{kE'}^{(X)*}). \quad (2.107)$$

When we average over the choice of BQPE and FBQPE, the $(\beta_{jE}^{(B)}\beta_{jE}^{(Y)*})(\beta_{mE'}^{(A)}\beta_{kE'}^{(X)*})$ term will become

$$\frac{1}{2}\left(\beta_{jE}^{(B)}\beta_{jE}^{(Y)*}\beta_{mE'}^{(A)}\beta_{kE'}^{(X)*}+\beta_{jE}^{(B)*}\beta_{jE}^{(Y)}\beta_{mE'}^{(A)*}\beta_{kE'}^{(X)}\right)=\Re(\beta_{jE}^{(B)}\beta_{jE}^{(Y)*}\beta_{mE'}^{(A)}\beta_{kE'}^{(X)*}). \quad (2.108)$$

We also average over the choice of C and finally get

$$\langle\psi_m|\,\mathcal{M}^{(a,ABXY)}(\rho_{\beta 0})\,|\psi_k\rangle = \sum_j p_{j0}\sum_{E,E'} f_{EE'}\sum_C \mu(C)\left(c_{jm}c_{jk}^*\right)\cdot\Re(\beta_{jE}^{(B)}\beta_{jE}^{(Y)*}\beta_{mE'}^{(A)}\beta_{kE'}^{(X)*}).$$

(b) Reject Case: Note that

$$\begin{aligned}
&\left\langle 0^{2gr+1}\right|\langle\psi_m|\,(U_C^{(BA)})^\dagger(W^{(10)})^\dagger W^{(10)}U_C^{(YX)}\,|\psi_k\rangle\left|0^{2gr+1}\right\rangle \quad (2.109)\\
=&\sum_{l;E,E'} f_{EE'}\left(c_{ml}^*c_{kl}\right)\left(\beta_{lE'}^{(B)*}\beta_{lE'}^{(Y)}\right)\left(\beta_{mE}^{(A)*}\beta_{kE}^{(X)}\right)\\
=&\sum_{j;E,E'} f_{E'E}\left(c_{mj}^*c_{kj}\right)\left(\beta_{jE}^{(B)*}\beta_{jE}^{(Y)}\right)\left(\beta_{mE'}^{(A)*}\beta_{kE'}^{(X)}\right),
\end{aligned}$$

where the last equality comes from changing the name l, E, E' to j, E', E. Use the definition of $\mathcal{M}^{(rr,ABXY)}$, we further have

$$\begin{aligned}
&\langle\psi_m|\mathcal{M}_C^{(rr,ABXY)}(\rho_{\beta 0})|\psi_k\rangle\\
=&\,p_{k0}\left\langle 0^{2gr+1}\right|\langle\psi_m|\,(U_C^{(BA)})^\dagger(W^{(10)})^\dagger W^{(10)}U_C^{(YX)}\,|\psi_k\rangle\left|0^{2gr+1}\right\rangle \quad (2.110)\\
=&\,p_{k0}\sum_{j;E,E'} f_{E'E}\left(c_{mj}^*c_{kj}\right)\left(\beta_{jE}^{(B)*}\beta_{jE}^{(Y)}\right)\left(\beta_{mE'}^{(A)*}\beta_{kE'}^{(X)}\right). \quad (2.111)
\end{aligned}$$

Note that in Eq. (2.111), the term $\left(\beta_{jE}^{(B)*}\beta_{jE}^{(Y)}\right)\left(\beta_{mE'}^{(A)*}\beta_{kE'}^{(X)}\right)$ is the complex conjugate of the corresponding term in Eq. (2.107). This is why in our algorithm we use

BQPE and FBQPE randomly to cancel this phase: when we average over the choice of BQPE and FBQPE, the $\left(\beta_{jE}^{(B)*}\beta_{jE}^{(Y)}\right)\left(\beta_{mE'}^{(A)*}\beta_{kE'}^{(X)}\right)$ term will become

$$\frac{1}{2}\left(\beta_{jE}^{(B)*}\beta_{jE}^{(Y)}\beta_{mE'}^{(A)*}\beta_{kE'}^{(X)}+\beta_{jE}^{(B)}\beta_{jE}^{(Y)*}\beta_{mE'}^{(A)}\beta_{kE'}^{(X)*}\right)=\Re(\beta_{jE}^{(B)}\beta_{jE}^{(Y)*}\beta_{mE'}^{(A)}\beta_{kE'}^{(X)*}). \quad (2.112)$$

We also average over the choice of C and get

$$\langle\psi_m|\mathcal{M}^{(rr,ABXY)}(\rho_{\beta 0})|\psi_k\rangle=p_{k0}\sum_{j;E,E'}f_{E'E}\sum_C\mu(C)\left(c_{mj}^*c_{kj}\right)\cdot\Re\left(\beta_{jE}^{(B)}\beta_{jE}^{(Y)*}\beta_{mE'}^{(A)}\beta_{kE'}^{(X)*}\right).$$

Further note that μ chooses C and $C^\dagger$ with the same probability, thus

$$\sum_C\mu(C)c_{jm}c_{jk}^*=\sum_C\mu(C^\dagger)c_{mj}^*c_{kj}=\sum_C\mu(C)c_{mj}^*c_{kj},$$

where the amplitudes for $C^\dagger$ are obtained from C by swapping the indices and taking the complex conjugate. Thus we conclude the proof.

■

Uniform Error

Before we analyze the errors, we first prove a Lemma which bound the trace norm of certain matrices.

Lemma 28 *If $2\beta w\le 1$, for any four subsets $A,B,X,Y\subseteq T$, we have*

$$\left|\mathcal{M}_C^{(a,ABXY)}(\rho_{\beta 0})\right|_1\le 2,\qquad \left|\mathcal{M}^{(a,ABXY)}(\rho_{\beta 0})\right|_1\le 2. \quad (2.113)$$

Proof: To ease notation, we abbreviate $\left|\psi_j\right\rangle\left|0^{2gr+1}\right\rangle$ as $\left|\psi_j 0^{2gr+1}\right\rangle$. Recall that

$$\mathcal{M}_C^{(a,ABXY)}(\left|\psi_j\right\rangle\left\langle\psi_j\right|)=tr_{2,3,4}\left(W^{(10)}U_C^{(AB)}\left|\psi_j 0^{2gr+1}\right\rangle\left\langle\psi_j 0^{2gr+1}\right|(U_C^{(XY)})^\dagger(W^{(10)})^\dagger\right) \quad (2.114)$$

$$\|U_C^{(AB)}\|\le\|P^{(A)}\|\cdot\|\mathrm{QPE}_{1,3}\|\cdot\|C\|\cdot\|P^{(B)}\|\cdot\|\mathrm{QPE}_{1,2}\|\le 1, \quad (2.115)$$

where in the last inequality we use that $P^{(A)},P^{(B)}$ are projections, and thus their spectrum norm is bounded by 1. Besides, $\|W^{(10)}\|\le 1$ by definition. Thus by Corollary 35 in Appendix 2.10 we get

$$\left|\mathcal{M}_C^{(a,ABXY)}(\left|\psi_j\right\rangle\left\langle\psi_j\right|)\right|_1\le 1. \quad (2.116)$$

Then when $2\beta w \leq 1$, by Lemma 22 we have $p_{j0} \leq p_j \cdot (1 + 2\beta w) \leq 2p_j$. By triangle inequality we have

$$\left|\mathcal{M}_C^{(a,ABXY)}(\rho_{\beta 0})\right|_1 \leq \sum_j 2p_j \left|\mathcal{M}_C^{(a,ABXY)}(|\psi_j\rangle\langle\psi_j|)\right|_1 \leq 2. \tag{2.117}$$

When averaging over the random selection of C, we can get the bound for $\mathcal{M}^{(a,uvst)}$ by triangle inequality.

■

The following is a key lemma in the analysis which clusters the projected operators and bounds the norm of each cluster separately. While $\langle\psi_m| (\mathcal{M}^{(a,AvsY)} - \mathcal{M}^{(rr,AvsY)})(\rho_{\beta 0}) |\psi_k\rangle$ may not cancel exactly to 0, the error for each cluster is independent of m and k, which allows us to factor out the error term over the entire matrix.

Lemma 29 *Recall that $T = \{0, 1, \text{“}else\}$. Consider four subsets $A, B, X, Y \subseteq T$. If $2\beta w \leq 1$*

(1) For any $v, s \in \{0, 1\}$, we have

$$\langle\psi_m| (\mathcal{M}^{(a,AvsY)} - \mathcal{M}^{(rr,AvsY)})(\rho_{\beta 0}) |\psi_k\rangle = (1 - e^{\beta(s-v)w}) \langle\psi_m| \mathcal{M}^{(a,AvsY)}(\rho_{\beta 0}) |\psi_k\rangle .$$
$$\left|\left(\mathcal{M}^{(a,AvsY)} - \mathcal{M}^{(rr,AvsY)}\right)(\rho_{\beta 0})\right|_1 \leq 4\beta w.$$

Note that the error $e^{\beta(s-v)w}$ in the first equality is uniform and independent of m, k.

(2) If one of A, B, X, Y equal to $\{\text{“}else''\}$, then we have

$$\left|\mathcal{M}^{(a,ABXY)}(\rho_{\beta 0})\right|_1 \leq 2 \cdot 2^{-g/10+2n}.$$
$$\left|\mathcal{M}^{(rr,ABXY)}(\rho_{\beta 0})\right|_1 \leq 2 \cdot 2^{-g/10+2n}.$$

Proof: Recall that from Lemma 27 we have for any m, k,

$$\langle\psi_m| \mathcal{M}^{(a,ABXY)}(\rho_{\beta 0}) |\psi_k\rangle = \sum_j p_{j0} \sum_{E,E'} f_{EE'} \sum_C \mu(C) \left(c_{jm} c_{jk}^*\right) \cdot \Re(\beta_{jE}^{(B)} \beta_{jE}^{(Y)*} \beta_{mE'}^{(A)} \beta_{kE'}^{(X)*}). \tag{2.118}$$

$$\langle\psi_m|\mathcal{M}^{(rr,ABXY)}(\rho_{\beta 0})|\psi_k\rangle = p_{k0} \sum_{j;E,E'} f_{E'E} \sum_C \mu(C) \left(c_{jm} c_{jk}^*\right) \cdot \Re\left(\beta_{jE}^{(B)} \beta_{jE}^{(Y)*} \beta_{mE'}^{(A)} \beta_{kE'}^{(X)*}\right)$$

Recall that

$$p_{js} = \exp\left(-\beta E_j^{(s)}\right)/Z.$$

With some abuse of notation, we use f for both

$$f_{EE'} = \min\left\{1, \exp\left(\beta\overline{E} - \beta\overline{E'}\right)\right\}, \tag{2.119}$$

$$f_{E_k^{(s)}E_j^{(v)}} = \min\left\{1, \exp\left(\beta E_k^{(s)} - \beta E_j^{(v)}\right)\right\}. \tag{2.120}$$

In other words, if E is a vector of g energies from $S(r)$, the function f implicitly takes the median value in determining $\min\left\{1, \exp\left(\beta\overline{E} - \beta\overline{E'}\right)\right\}$.

For (1): Suppose $v, s \in \{0, 1\}$. The key thing to notice is that, by definition $\beta_{jE}^{(v)}$ is non-zero only if $\overline{E} = E_j^{(v)}$. Similarly for $\beta_{kE'}^{(s)*}$. Thus $\langle\psi_m| \mathcal{M}^{(a,AvsY)}(\rho_{\beta 0}) |\psi_k\rangle$ is a sum of terms, where all the non-zero terms $f_{EE'}$ take a uniform value as $f_{E_j^{(v)}E_k^{(s)}}$:

$$\begin{aligned}
&\langle\psi_m| \mathcal{M}^{(a,AvsY)}(\rho_{\beta 0}) |\psi_k\rangle \\
&= \sum_j p_{j0} \sum_{\substack{E,E':\overline{E}=E_j^{(v)} \\ \overline{E'}=E_k^{(s)}}} f_{EE'} \sum_C \mu(C)\left(c_{jm}c_{jk}^*\right) \cdot \Re(\beta_{jE}^{(v)}\beta_{jE}^{(Y)*}\beta_{mE'}^{(A)}\beta_{kE'}^{(s)*}) \\
&= \sum_j p_{j0} \cdot f_{E_j^{(v)}E_k^{(s)}} \sum_{\substack{E,E':\overline{E}=E_j^{(v)} \\ \overline{E'}=E_k^{(s)}}} \sum_C \mu(C)\left(c_{jm}c_{jk}^*\right) \cdot \Re(\beta_{jE}^{(v)}\beta_{jE}^{(Y)*}\beta_{mE'}^{(A)}\beta_{kE'}^{(s)*}).
\end{aligned}$$

Similarly we have

$$\begin{aligned}
&\langle\psi_m|\mathcal{M}^{(rr,AvsY)}(\rho_{\beta 0})|\psi_k\rangle \\
&= p_{k0} \sum_j \sum_{\substack{E,E':\overline{E}=E_j^{(v)} \\ \overline{E'}=E_k^{(s)}}} f_{E'E} \sum_C \mu(C)\left(c_{jm}c_{jk}^*\right) \cdot \Re\left(\beta_{jE}^{(v)}\beta_{jE}^{(Y)*}\beta_{mE'}^{(A)}\beta_{kE'}^{(s)*}\right) \\
&= \sum_j p_{k0} \cdot f_{E_k^{(s)}E_j^{(v)}} \sum_{\substack{E,E':\overline{E}=E_j^{(v)} \\ \overline{E'}=E_k^{(s)}}} \sum_C \mu(C)\left(c_{jm}c_{jk}^*\right) \cdot \Re\left(\beta_{jE}^{(v)}\beta_{jE}^{(Y)*}\beta_{mE'}^{(A)}\beta_{kE'}^{(s)*}\right).
\end{aligned}$$

Note that by definition of f and p_{ks} we always have

$$p_{ks} \cdot f_{E_k^{(s)}E_j^{(v)}} = p_{jv} \cdot f_{E_j^{(v)}E_k^{(s)}}, \tag{2.121}$$

$$p_{ks} = p_{k0} \cdot e^{-\beta s w}. \tag{2.122}$$

Thus

$$p_{k0} \cdot f_{E_k^{(s)} E_j^{(v)}} = e^{\beta s w} p_{ks} \cdot f_{E_k^{(s)} E_j^{(v)}} \tag{2.123}$$

$$= e^{\beta s w} p_{jv} \cdot f_{E_j^{(v)} E_k^{(s)}} \tag{2.124}$$

$$= e^{\beta (s-v) w} p_{j0} \cdot f_{E_j^{(v)} E_k^{(s)}}. \tag{2.125}$$

Note that the "error" $e^{\beta(s-v)w}$ is independent of j, m, k, thus

$$\langle \psi_m | \left(\mathcal{M}^{(a,AvsY)} - \mathcal{M}^{(rr,AvsY)} \right) (\rho_{\beta 0}) | \psi_k \rangle \tag{2.126}$$

$$= (1 - e^{\beta(s-v)w}) \sum_j p_{j0} \cdot f_{E_j^{(v)} E_k^{(s)}} \sum_{\substack{E, E' : \overline{E} = E_j^{(v)} \\ \overline{E'} = E_k^{(s)}}} \sum_C \mu(C) \left(c_{jm} c_{jk}^* \right) \cdot \Re(\beta_{jE}^{(v)} \beta_{jE}^{(Y)*} \beta_{mE'}^{(A)} \beta_{kE'}^{(s)*}) \tag{2.127}$$

$$= (1 - e^{\beta(s-v)w}) \langle \psi_m | \mathcal{M}^{(a,AvsY)} (\rho_{\beta 0}) | \psi_k \rangle . \tag{2.128}$$

Thus

$$\left(\mathcal{M}^{(a,AvsY)} - \mathcal{M}^{(rr,AvsY)} \right) (\rho_{\beta 0}) = (1 - e^{\beta(s-v)w}) \cdot \mathcal{M}^{(a,AvsY)} (\rho_{\beta 0}), \tag{2.129}$$

$$\left| \left(\mathcal{M}^{(a,AvsY)} - \mathcal{M}^{(rr,AvsY)} \right) (\rho_{\beta 0}) \right|_1 \leq 4\beta w. \tag{2.130}$$

where for the last inequality, we use Lemma 28 and the fact that since $v, s \in \{0, 1\}$, we have $|s - v| \leq 1$ and $|1 - e^{\beta(s-v)w}| \leq 2\beta w$.

For (2). W.o.l.g. assume $X = \{\text{“}else''\}$. Other cases are similar. Note that

$$\left(\sum_j \left| c_{jm} c_{jk}^* \right| \right)^2 \leq \left(\sum_j |c_{jm}|^2 \right) \left(\sum_j \left| c_{jk}^* \right|^2 \right) \leq 1 \tag{2.131}$$

$$\left(\sum_E |\beta_{jE}^{(B)} \beta_{jE}^{(Y)*}| \right)^2 \leq \left(\sum_E \left| \beta_{jE}^{(B)} \right|^2 \right) \left(\sum_E \left| \beta_{jE}^{(Y)*} \right|^2 \right) \leq 1 \cdot 1 \tag{2.132}$$

$$\left(\sum_{E'} |\beta_{mE'}^{(A)} \beta_{kE'}^{(X)*}| \right)^2 \leq \left(\sum_{E'} \left| \beta_{mE'}^{(A)} \right|^2 \right) \left(\sum_{E'} \left| \beta_{kE'}^{(X)*} \right|^2 \right) \leq 2^{-g/5} \cdot 1, \tag{2.133}$$

where the first equality comes from the fact that C is a unitary. The second equality comes from $\{\beta_{jE}^{(B)}\}_E$ is a subset of $\{\beta_{jE}\}_E$, and $\{\beta_{jE}\}_E$ is the amplitude of a quantum state. For the third equality, recall that $X = \{\text{“}else''\}$, notice that for any non-zero $\beta_{mE'}^{else}$, by definition we have $\overline{E'} \notin \{E_j^{(0)}, E_j^{(1)}\}$. Then the third equality comes from property of BQPE, which is Lemma 3.

Besides, note that

$$|\Re(\beta_{jE}^{(B)}\beta_{jE}^{(Y)*}\beta_{mE'}^{(A)}\beta_{kE'}^{(X)*})| \leq |\beta_{jE}^{(B)}\beta_{jE}^{(Y)*}\beta_{mE'}^{(A)}\beta_{kE'}^{(X)*}|. \tag{2.134}$$

Since $2\beta w \leq 1$, by Lemma 22 we have $p_{j0} \in [0,2]$. Note that $f_{EE'} \in [0,1]$ and $\sum_C \mu(C) = 1$, then from Eq. (2.118) we have for any m, k,

$$|\langle\psi_m|\mathcal{M}^{(a,ABXY)}(\rho_{\beta 0})|\psi_k\rangle| \leq 2\sum_C \mu(C)\sum_j \left|c_{jm}c_{jk}^*\right| \cdot \sum_E |\beta_{jE}^{(B)}\beta_{jE}^{(Y)*}| \cdot \sum_{E'} |\beta_{mE'}^{(A)}\beta_{kE'}^{(X)*}| \tag{2.135}$$

$$\leq 2\cdot 2^{-g/10}. \tag{2.136}$$

Then by triangle inequality,

$$\left|\mathcal{M}^{(a,ABXY)}(\rho_{\beta 0})\right|_1 \leq \sum_{m,k} |\langle\psi_m|\mathcal{M}^{(a,ABXY)}(\rho_{\beta 0})|\psi_k\rangle| \cdot ||\psi_m\rangle\langle\psi_k||_1 \leq 2\cdot 2^{-g/10+2n}. \tag{2.137}$$

The proof for $\left|\mathcal{M}^{(rr,ABXY)}(\rho_{\beta 0})\right|_1$ is similar.

■

Gibbs state as Approximate Fixed Point

We are ready now to complete the proof of Lemma 21, which provides an upper bound on $\left|\mathcal{L}(\rho_\beta)\right|_1$.

Proof:[of Lemma 21] If $r \geq r_{\beta H}$, we have $2\beta w \leq 1$. From Lemma 23 and Lemma 24 we have

$$\left|\mathcal{L}(\rho_\beta)\right|_1 \leq 2\left|\mathcal{M}^{(a)}(\rho_{\beta 0}) - \mathcal{M}^{(rr)}(\rho_{\beta 0})\right|_1 + 4\cdot 2\beta w. \tag{2.138}$$

Note that for any matrices $\{N^{(uvst)}\}_{uvst}$, by triangle inequality we have

$$\left|\sum_{u,v,s,t\in T} N^{(uvst)}\right|_1 \leq \left|\sum_{\substack{v=else\\u,s,t\in T}} N^{(uvst)}\right|_1 + \left|\sum_{\substack{v\in\{0,1\},s=else\\u,t\in T}} N^{(uvst)}\right|_1 + \sum_{v,s\in\{0,1\}}\left|\sum_{u,t\in T} N^{(uvst)}\right|_1.$$

Then by Lemma 26 and Lemma 29, we have

$$\left|\mathcal{M}^{(a)}(\rho_{\beta 0}) - \mathcal{M}^{(rr)}(\rho_{\beta 0})\right|_1 \leq 2\cdot(2\cdot 2^{-g/10+2n} + 2\cdot 2^{-g/10+2n}) + 4\cdot 4\beta w\cdot \tag{2.139}$$

Substituting $w = \kappa_H \cdot 2^{-r}$, we complete the proof. ■

2.8 Proofs of Theorem 4 and Theorem 6

Proof:[of Theorem 4] The fact that one step of the algorithm can be expressed as $\mathcal{E}[\tau](\rho) = \left(\mathcal{I} + \tau^2\mathcal{L} + \tau^4\mathcal{J}[\tau]\right)(\rho)$, where $|\mathcal{J}[\tau]|_\star \leq 4$ is proven in Lemma 16 in Section 2.5. The bound on $\left|\mathcal{L}(\rho_\beta)\right|_1$ is proved in Lemma 21 in Section 2.7. The Uniqueness and Relaxation property is proved in Theorem 18 in Section 2.6. ∎

Before proving theorem 6 we give a bound for the evolution.

Lemma 30 (Evolution) *For any $t \in \mathbb{R}$, $\tau \in (0, 1]$. If $K = t/\tau^2$ is an integer, then*

$$\left|\mathcal{E}^K - e^{t\mathcal{L}}\right|_\star \leq 2e^4 K\tau^4. \tag{2.140}$$

Proof: Note that

$$\mathcal{E}^K - (e^{\tau^2\mathcal{L}})^K = \sum_{k=0}^{K-1} (e^{\tau^2\mathcal{L}})^k \left(\mathcal{E} - e^{\tau^2\mathcal{L}}\right) \mathcal{E}^{K-k-1}. \tag{2.141}$$

Note that $\mathcal{E}$ is CPTP by definition. By Lemma 19 we know $e^{\tau^2\mathcal{L}}$ is also CPTP. Thus $|\mathcal{E}|_\star$ and $\left|e^{\tau^2\mathcal{L}}\right|_\star$ are bounded by 1 by Lemma 38 in Appendix 2.10. Then we have

$$\left|\mathcal{E}^K - (e^{\tau^2\mathcal{L}})^K\right|_\star \leq K \cdot 1 \cdot \left|\mathcal{E} - e^{\tau^2\mathcal{L}}\right|_\star \cdot 1 \leq K \cdot \tau^4 2e^4, \tag{2.142}$$

where for the last inequality, recall that $\mathcal{E} = \mathcal{I} + \tau^2\mathcal{L} + \tau^4\mathcal{J}[\tau]$ and the Taylor expansion of $e^{\tau^2\mathcal{L}}$, we get

$$\left|\mathcal{E} - e^{\tau^2\mathcal{L}}\right|_\star \leq |\mathcal{J}[\tau]|_\star \tau^4 + \sum_{k=2}^{\infty} \frac{\tau^{2k} |\mathcal{L}|_\star^k}{k!} \leq \tau^4 2e^4. \tag{2.143}$$

where we use $|\mathcal{J}|_\star \leq 4$ and $|\mathcal{L}|_\star \leq 4$ from Lemma 16. ∎

We then prove that the distance between ρ_β and $\rho_{\mathcal{L}}$ can be bounded in terms of the mixing time.

Lemma 31

$$\left|\rho_{\mathcal{L}} - \rho_\beta\right|_1 \leq \epsilon + \left|\mathcal{L}(\rho_\beta)\right|_1 \cdot t_{mix}(\mathcal{L}, \epsilon)$$

Proof: We abbreviate $t_{mix}(\mathcal{L}, \epsilon)$ as t. Let $\tau > 0$ be a parameter such that $K = t/\tau^2$ is an integer. We have

$$\left|\rho_{\mathcal{L}} - \rho_\beta\right|_1 \leq \left|\rho_{\mathcal{L}} - e^{t\mathcal{L}}(\rho_\beta)\right|_1 + \left|e^{t\mathcal{L}}(\rho_\beta) - \rho_\beta\right|_1 \leq \epsilon + \left|e^{t\mathcal{L}}(\rho_\beta) - \rho_\beta\right|_1, \tag{2.144}$$

where the last inequality comes from the definition of mixing time. Besides, from Lemma 16 we have $|\mathcal{L}|_\star \leq 4$. Expand $e^{\tau^2\mathcal{L}}$ as Taylor series in τ, and use triangle inequality we get

$$\left|\left(e^{\tau^2\mathcal{L}} - \mathcal{I}\right)(\rho_\beta)\right|_1 \leq \tau^2 \left|\mathcal{L}(\rho_\beta)\right|_1 + \tau^4 \sum_{k=2}^{\infty} \frac{|\mathcal{L}|_\star^k}{k!} \tag{2.145}$$

$$\leq \tau^2 \left|\mathcal{L}(\rho_\beta)\right|_1 + \tau^4 e^4. \tag{2.146}$$

Thus

$$\left|e^{K\tau^2\mathcal{L}}(\rho_\beta) - \rho_\beta\right|_1 \leq \sum_{k=K}^{1} \left|e^{(k-1)\tau^2\mathcal{L}}\left(e^{\tau^2\mathcal{L}} - \mathcal{I}\right)(\rho_\beta)\right|_1 \tag{2.147}$$

$$\leq \sum_{k=K}^{1} \left|\left(e^{\tau^2\mathcal{L}} - \mathcal{I}\right)(\rho_\beta)\right|_1 \tag{2.148}$$

$$\leq K\tau^2 \left|\mathcal{L}(\rho_\beta)\right|_1 + K\tau^4 e^4, \tag{2.149}$$

where the second inequality comes from $e^{(K-1)\tau^2\mathcal{L}}$ is CPTP by Lemma 19, thus its $|\cdot|_\star$ is bounded by 1 by Lemma 38 in Appendix 2.10. Thus we finish the proof by substituting $K\tau^2$ with the mixing time t and take the limit $\tau \to 0$. ∎

Proof:[of Theorem 6] In the proof we abbreviate $t_{mix}(\mathcal{L}, \epsilon)$ as t and $K = t/\tau^2$. We will analyze the following terms,

$$\left|\mathcal{E}^K(\rho) - \rho_\beta\right|_1 \leq \left|\mathcal{E}^K(\rho) - e^{t\mathcal{L}}(\rho)\right|_1 + \left|e^{t\mathcal{L}}(\rho) - \rho_{\mathcal{L}}\right|_1 + \left|\rho_{\mathcal{L}} - \rho_\beta\right|_1. \tag{2.150}$$

By Lemma 30 we can bound the first term by $2e^4 K\tau^4$. By the definition of mixing time, we can bound the second term by ϵ. The bound for last term comes from Lemma 31 and Theorem 4. Thus we get the desired error bounds by substituting $K\tau^2$ with the mixing time t.

∎

Acknowledgement

Part of this work was conducted while the author was visiting the Simons Institute for the Theory of Computing, supported by DOE QSA grant #FP00010905. We thank Yu Tong for the helpful discussion on analyzing mixing time by spectral gap of $\mathcal{L}^{(s)}$.

2.9 Appendix: More Details on Quantum Phase Estimation

(1) Adapting quantum phase estimation to estimating Hamiltonian eigenvalues. Firstly we recall the standard quantum phase estimation in Section 5.2 in [NC10]:

suppose a unitary V has an eigenstate $|\phi\rangle$ with eigenvalue $e^{2\pi\mu}$, where $\mu \in [0,1)$ is unknown. Quantum Phase estimation is a quantum algorithm which has access to $|\phi\rangle$ and V, and outputs an estimate of μ.

More precisely, let $|\phi\rangle := \left|\psi_j\right\rangle$, $U := e^{iHt}$, $t := \frac{2\pi}{\kappa_H}$, where $\kappa_H = poly(n)$ is a power of two that upper bounds $\|H\|$. For example, for local Hamiltonian $H = \sum_{i=1}^{m} H_i, \|H_i\| \leq 1$, one can set κ_H to be the least integer which is a power of two and is greater than m. Then $\left|\psi_j\right\rangle$ is an eigenstate of U with eigenvalue $e^{2\pi\mu}$, for

$$\mu := \frac{E_j t}{2\pi} = \frac{E_j}{\kappa_H} \in [0,1).$$

Let r be an integer. For any $\boldsymbol{b} \in \{0,1\}^r$, with some abuse of notations, we use $\boldsymbol{b}$ both for the binary string and the integer $\sum_{j=1}^{r} b_j 2^{j-1}$. Denote $\boldsymbol{b}^{(j)} \in \{0,1\}^r$ be the integer such that $\boldsymbol{b}^{(j)}/2^r$ is the best r bit approximation to $\frac{E_j}{\kappa_H}$ which is less than $\frac{E_j}{\kappa_H}$. Then if $\boldsymbol{b}$ is a good approximation to $\boldsymbol{b}^{(j)}$, we have $E(\boldsymbol{b})$ defined below, which is a good approximation of E_j:

$$E(\boldsymbol{b}) := \kappa_H \cdot \boldsymbol{b}/2^r. \tag{2.151}$$

Note that since κ_H is a power of 2, the binary representation of $\boldsymbol{b}$ can also be viewed as the binary representation of $E(\boldsymbol{b})$ by shifting the decimal point by $\log(\kappa_H)$ bits.

The quantum phase estimation w.r.t precision r in [NC10] is a unitary, denoted as QPE, which outputs a distribution highly peaked at $\boldsymbol{b}^{(j)}$,

$$\mathrm{QPE}\left|\psi_j\right\rangle|0^r\rangle = \left|\psi_j\right\rangle \sum_{l=-2^{r-1}+1}^{2^{r-1}} \gamma_{jl} \left|\boldsymbol{b}^{(j)} + l \mod 2^r\right\rangle, \tag{2.152}$$

$$\text{where } \gamma_{jl} := \frac{1}{2^r} \sum_{k=0}^{2^r-1} \left(e^{2\pi i(\epsilon_j - l/2^r)}\right)^k = \frac{1}{2^r}\left(\frac{1-e^{2\pi i(2^r\epsilon_j - l)}}{1-e^{2\pi i(\epsilon_j - l/2^r)}}\right), \tag{2.153}$$

$$\epsilon_j := E_j/\kappa_H - b^{(j)}/2^r. \tag{2.154}$$

Lemma 32 *Note that*

$$|\gamma_{j0}|^2 + |\gamma_{j1}|^2 \geq 0.8. \tag{2.155}$$

Proof: By definition of $\boldsymbol{b}^{(j)}$ and ϵ_j we have that $\epsilon_j \in [0, 2^{-r})$. Note that if $\epsilon_j = 0$, then $|\gamma_{j0}|^2 = 1$, so for the remainder of the proof, we will assume $\epsilon_j \in (0, 2^{-r})$. Define $\theta = 2^r\epsilon_j \in (0,1)$. Note that

$$\gamma_{jl} = \frac{1}{2^r}\left(\frac{1-e^{2\pi i(\theta-l)}}{1-e^{2\pi i(\theta-l)/2^r}}\right) = \frac{1}{2^r}\frac{e^{\pi i(\theta-l)}}{e^{\pi i(\theta-l)/2^r}} \cdot \frac{\sin(\pi(\theta-l))}{\sin(\pi(\theta-l)/2^r)}. \tag{2.156}$$

So the goal is to lower bound:

$$|\gamma_{j0}|^2 + |\gamma_{j1}|^2 = \frac{1}{4^r}\frac{\sin^2(\theta\pi)}{\sin^2(\theta\pi/2^r)} + \frac{1}{4^r}\frac{\sin^2((\theta-1)\pi)}{\sin^2((\theta-1)\pi/2^r)}. \tag{2.157}$$

Since $\sin(\theta) \leq \theta$ for $\theta \geq 0$, we can use this lower bound in the denominator for each term:

$$|\gamma_{j0}|^2 + |\gamma_{j1}|^2 \geq \frac{1}{4^r}\frac{\sin^2(\theta\pi)}{(\theta\pi/2^r)^2} + \frac{1}{4^r}\frac{\sin^2((\theta-1)\pi)}{((\theta-1)\pi/2^r)^2} \tag{2.158}$$

$$= \frac{1}{\pi^2}\left(\frac{\sin^2(\theta\pi)}{\theta^2} + \frac{\sin^2(\theta-1)\pi)}{(\theta-1)^2}\right). \tag{2.159}$$

Note that as shown in Figure 2.2 for $\theta \in (0,1)$, the function in Eq. (2.159) is symmetric around $\theta = 1/2$ and is minimized for $\theta = 1/2$. The value obtained by plugging $\theta = 1/2$ into Eq. (2.159) is at least 0.8 .

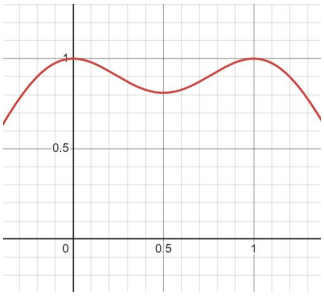


Figure 2.2: Approximating the amplitudes of the quantum phase estimation. The function shown in Equation (2.159) is minimized for $\theta = 1/2$.

■

Note that γ_{j0} and γ_{j1} are the amplitudes of $\boldsymbol{b}^{(j)}$ and $\boldsymbol{b}^{(j)}+1$, which are the best two r-bit approximation to $\frac{E_j}{\kappa_H}$. Denote the best two r-bit approximation to E_j as

$$\lfloor E_j \rfloor := E(\boldsymbol{b}^{(j)}), \quad \lceil E_j \rceil := E(\boldsymbol{b}^{(j)}+1). \tag{2.160}$$

Since there is a one-to-one correspondence between r-bit string $\boldsymbol{b}$ and $E(\boldsymbol{b}) \in S(r)$, to ease notations, we rewrite Eqs. (2.152)(2.155) as

$$\mathrm{QPE}\left|\psi_j\right\rangle|0^r\rangle = \left|\psi_j\right\rangle \sum_{\nu\in S(r)} \beta_{j\nu}|\nu\rangle, \tag{2.161}$$

$$\text{where } |\beta_{j\lfloor E_j\rfloor}|^2 + |\beta_{j\lceil E_j\rceil}|^2 \geq 0.8. \tag{2.162}$$

In addition, one can check that $\forall j$,

$$\sum_{\nu\in S(r)} \beta_{j\nu} = \sum_{l=-2^{r-1}+1}^{2^{r-1}} \gamma_{jl} \tag{2.163}$$

$$= \frac{1}{2^r} \sum_{k=0}^{2^r-1} \sum_{l=-2^{r-1}+1}^{2^{r-1}} \left(e^{2\pi i\epsilon_j}\right)^k \left(e^{2\pi i(-l/2^r)}\right)^k \tag{2.164}$$

$$= \frac{1}{2^r} \sum_{k=0}^{2^r-1} \left(e^{2\pi i\epsilon_j}\right)^k \cdot 2^r \delta_{k0} \tag{2.165}$$

$$= 1. \tag{2.166}$$

(2) Boosted Quantum Phase Estimation. Let r, g be two integer parameters. Taking input as $\left|\psi_j\right\rangle |0^{rg}\rangle$, the Boosted quantum phase estimation (BQPE) in this manuscript is just viewing the $|0^{rg}\rangle$ as g copies of $|0^r\rangle$, and sequentially perform the standard Quantum Phase Estimation QPE w.r.t. $\left|\psi_j\right\rangle$ and each copy of $|0^r\rangle$,

$$\text{BQPE}\left|\psi_j\right\rangle |0^{gr}\rangle = \left|\psi_j\right\rangle \sum_{\nu_1,\ldots,\nu_g\in S(r)} \beta_{j\nu_1}\ldots\beta_{j\nu_g} \left|\nu_1\ldots\nu_g\right\rangle, \tag{2.167}$$

$$= \left|\psi_j\right\rangle \sum_{E\in S(r)^{\otimes g}} \beta_{jE} |E\rangle, \tag{2.168}$$

where in the last equality to ease notations we write $E := \nu_1\ldots\nu_g$ and $\beta_{jE} := \beta_{j\nu_1}\ldots\beta_{j\nu_g}$. We use $\overline{E}$ to denote the median of $\nu_1, \ldots, \nu_g$. By Eq. (2.162) and Chernoff bound, we have

$$\sum_{\substack{E\in S(r)^{\otimes g}:\\ \overline{E}\neq\lfloor E_j\rfloor, \overline{E}\neq\lceil E_j\rceil}} |\beta_{jE}|^2 \leq 2^{-g/5}. \tag{2.169}$$

Besides, by Eq. (2.166) we have

$$\sum_{E\in S(r)^{\otimes g}} \beta_{jE} = \left(\sum_{\nu\in S(r)} \beta_{j\nu}\right)^g = 1. \tag{2.170}$$

(3) Flipped Boosted Quantum Phase Estimation. As explained in Section 5.2 of [NC10], the circuit for the standard Quantum Phase Estimation QPE is, (a) First applying a layer of Hadamard gates. (b) Then apply a sequence controlled-U operators, with U raised to successive powers of 2. Denote the operator as CU. (c) Then apply inverse quantum Fourier transform.

The Flipped Quantum Phase Estimation FQPE is similar to QPE, with the difference that each time FQPE adds a conjugate phase: (a) First applying a layer of Hadamard

gate. (b) Then apply $(CU)^\dagger$. (c) Then apply quantum Fourier transform rather than inverse quantum Fourier transform. It is worth noting that FQPE $\neq$ QPE†.

Recall that from (2) the Boosted Quantum Phase estimation is just running QPE for g times. The Flipped Boosted Quantum Phase estimation (FBQPE) is just running FQPE for g times.

2.10 Appendix: Matrix Norm Properties

In this section we list properties on matrix norms.

Lemma 33 (Cauchy-Schwarz inequality) *For any $M, N \in \Xi(m)$,*

$$|tr(A^\dagger B)| \leq \sqrt{tr(A^\dagger A)}\sqrt{tr(B^\dagger B)}. \tag{2.171}$$

Lemma 33 is a well-known fact thus we skip the proof.

Lemma 34 (Variational Characterization of Trace Norm) *For $M \in \Xi(m)$, the following variational characterization of the trace norm holds,*

$$|M|_1 = \max_U |tr(MU)|, \tag{2.172}$$

where the maximization is over all m-qubit unitary operators U.

Proof: Consider singular value decomposition of M as WDV, with W, V unitaries and D a diagonal matrix of singular values. Using Cauchy-Schwart inequality Lemma 33 one gets

$$\begin{aligned} |tr(MU)| &= |tr(\sqrt{D}\sqrt{D}VUW)| \\ &\leq \sqrt{tr(\sqrt{D}\sqrt{D})}\sqrt{tr\left\{\left(\sqrt{D}VUW\right)^\dagger\left(\sqrt{D}VUW\right)\right\}} \\ &= tr(D). \end{aligned}$$

Note that $tr(D) = |M|_1$. On the other hand, $|M|_1$ can be obtained by choosing $U = V^\dagger W^\dagger$. ∎

Corollary 35 *Let $|\phi\rangle, |\varphi\rangle$ be two unit vectors on registers a, b. Let P, Q be linear operators on registers a, b with spectrum norm bounded by 1. Then*

$$|tr_b(P|\phi\rangle\langle\varphi|Q)|_1 \leq 1. \tag{2.173}$$

In particular, let $P = Q = I$ we get

$$|tr_b(|\phi\rangle\langle\varphi|)|_1 \leq 1. \tag{2.174}$$

Proof: By Lemma 34, we have that

$$|tr_b(P|\phi\rangle\langle\varphi|Q)|_1 = \max_{\text{unitary } U \text{ on register } a} |tr_a(tr_b(P|\phi\rangle\langle\varphi|Q)U)| \tag{2.175}$$

$$= |tr_{a,b}(P|\phi\rangle\langle\varphi|QU)| \tag{2.176}$$

$$= |\langle\varphi|QUP|\phi\rangle| \tag{2.177}$$

$$\leq 1. \tag{2.178}$$

■

Lemma 36 *Let $\mathcal{E} : \Xi(m) \to \Xi(m)$ be a linear map. If for any unit vector $|u\rangle, |v\rangle$, we have $|\mathcal{E}(|u\rangle\langle v|)|_1 \leq c$. Then $|\mathcal{E}|_\star \leq c$.*

Proof: Given any $N \in \Xi(m)$ with $|N|_1 \leq 1$, consider its spectrum decomposition

$$N = \sum_i \gamma_i |u_i\rangle\langle v_i|, \tag{2.179}$$

where $\gamma_i \geq 0$ are singular values. $|u_i\rangle, |v_i\rangle$ are unit vectors. $\sum_i \gamma_i \leq |N|_1 = 1$. Then by triangle inequality we have that

$$|\mathcal{E}(N)|_1 \leq \sum_i \gamma_i \cdot |\mathcal{E}(|u\rangle\langle v|)|_1 \leq c. \tag{2.180}$$

■

Lemma 37 *Let P, Q be operators on register $1,2,3,4$ with spectrum norm bounded by 1. Let $|w\rangle$ be a unit vector on register $2,3,4$. For any operator N on register 1, define*

$$\mathcal{F}(N) := tr_{2,3,4}(P[N \otimes |w\rangle\langle w|]Q). \tag{2.181}$$

Then $|\mathcal{F}|_\star \leq 1$.

Proof: By Lemma 36 it suffices to assume $N = |u\rangle\langle v|$ for unit vectors $|u\rangle, |v\rangle$ and prove $|\mathcal{F}(N)|_1 \leq 1$. Since $|u\rangle|w\rangle$ and $|v\rangle|w\rangle$ are unit vectors on register $1,2,3,4$. By Corollary 35 we have $|\mathcal{F}|_\star \leq 1$. ■

Lemma 38 *[Trace-Norm non-increasing of CPTP] Let $\mathcal{E} : \mathcal{H}(m) \to \mathcal{H}(m)$ be a CPTP map. We have $|\mathcal{E}|_\star \leq 1$. That is, for any Hermitian operator $M \in \mathcal{H}(m)$, $|\mathcal{E}(M)|_1 \leq |M|_1$.*

Proof: Consider the spectrum decomposition of $M = \sum_j \lambda_j \left|\phi_j\right\rangle \left\langle\phi_j\right|$. We have that

$$|\mathcal{E}(M)|_1 \leq \sum_j |\lambda_j| \cdot \left|\mathcal{E}(\left|\phi_j\right\rangle \left\langle\phi_j\right|)\right|_1 = \sum_j |\lambda_j| = |M|_1 , \tag{2.182}$$

where the first equality comes from the fact that $\mathcal{E}(\left|\phi_j\right\rangle \left\langle\phi_j\right|)$ is a quantum mixed state, and thus its trace norm is equal to 1. ∎

Lemma 39 (Connecting trace norm and σ-norm) *Let σ be a Hermitian matrix where $\sigma \geq I$. Then $\forall M \in \Xi(n)$, we have that $|M|_1^2 \leq 2^n |M|_\sigma^2$.*

Proof: Note that for any two positive Semi-definite Hermitian matrices $A \geq 0, B \geq 0$, we have $tr(AB) \geq 0$. Further note that $\sigma^{1/2} - I \geq 0$, $M^\dagger \sigma^{1/2} M \geq 0$, and both of them are Hermitian. Thus we have

$$|M|_\sigma^2 = tr(IM^\dagger IM) + tr(IM^\dagger \left(\sigma^{1/2} - I\right) M) + tr((\sigma^{1/2} - I) \cdot M^\dagger \sigma^{1/2} M) \tag{2.183}$$

$$\geq tr(IM^\dagger IM). \tag{2.184}$$

Denote the singular values of M as $\{a_j\}_j$. (By definition $a_j \geq 0$). By Cauchy inequality

$$\left(\sum_j a_j\right)^2 \leq 2^n \sum_j a_j^2 \quad \Rightarrow \quad |M|_1^2 \leq 2^n tr(IM^\dagger IM) \leq 2^n |M|_\sigma^2 . \tag{2.185}$$

∎

2.11 Appendix: Bounding Mixing time w.r.t. spectral gap of $\mathcal{L}^{(s)}$

This section is based on [Wol12] and private communications. This section is primarily for the $\mathcal{L}$ in our algorithm, while here we write a slightly more general proof for Lindbladian $\mathcal{L}$ satisfying the following assumptions

Assumption 1 *We assume $e^{t\mathcal{L}}$ is CPTP for $t \geq 0$ and satisfies*

- $\mathcal{L}(\rho) = \sum_j L_j \rho L_j^\dagger - \frac{1}{2} \left\{L_j^\dagger L_j, \rho\right\}_+$ *for some matrices $\{L_j\}_j \subseteq \Xi(n)$.*
- *$\{L_j\}_j$ generates the full algebra $\Xi(n)$. There exists a unique $\rho_\mathcal{L}$ satisfying $\mathcal{L}(\rho_\mathcal{L}) = 0$. Besides, this $\rho_\mathcal{L}$ is a full-rank quantum state.*

We use σ to denote $\rho_\mathcal{L}^{-1}$. Note that $\mathcal{L}$ in our algorithm satisfies Assumption 1 by Lemma 17 and proofs in Section 2.6.

Notations. Here we define two different inner products on $\Xi(n)$, that is the Schmidt-Hilbert inner product and the weighted inner product w.r.t. σ: For any $M, N \in \Xi(n)$,

$$\langle M, N\rangle := tr\left(M^\dagger N\right), \tag{2.186}$$

$$\langle M, N\rangle_\sigma := tr\left(\sigma^{1/2} M^\dagger \sigma^{1/2} N\right). \tag{2.187}$$

We use $\mathcal{L}^\diamond$ to denote the dual map w.r.t. Schmidt inner product. We use $\mathcal{L}^*$ to denote the dual map w.r.t. $\langle , \rangle_\sigma$. That is

$$\langle M, \mathcal{L}^\diamond(N)\rangle = \langle \mathcal{L}(M), N\rangle, \tag{2.188}$$

$$\langle M, \mathcal{L}^*(N)\rangle_\sigma = \langle \mathcal{L}(M), N\rangle_\sigma. \tag{2.189}$$

$\langle , \rangle_\sigma$ induces a norm where $|N|_\sigma^2 := \langle N, N\rangle_\sigma$. One can check that for any $N \in \Xi(n)$,

$$\mathcal{L}^*(N) = \sigma^{-\frac{1}{2}} \mathcal{L}^\diamond\left(\sigma^{\frac{1}{2}} N \sigma^{\frac{1}{2}}\right) \sigma^{-\frac{1}{2}} \tag{2.190}$$

$$\mathcal{L}^\diamond(N) = \sum_j L_j^\dagger N L_j - \frac{1}{2}\left\{L_j^\dagger L_j, N\right\}_+. \tag{2.191}$$

Define the symmetrized version of $\mathcal{L}$ as

$$\mathcal{L}^{(s)} := \frac{\mathcal{L} + \mathcal{L}^*}{2}.$$

By definition $\mathcal{L}^{(s)}$ is Hermitian w.r.t. $\langle , \rangle_\sigma$, and thus is diagonalizable and has a real spectrum.

Properties on $\mathcal{L}$ and $\mathcal{L}^{(s)}$.

In this section, we prove some basic properties of $\mathcal{L}$ and $\mathcal{L}^{(s)}$. We use the following theorem.

Theorem 40 (Generators for Semigroup of quantum channels, Theorem 7.1 in [Wol12]) *Consider a linear map $\mathcal{P} : \Xi(n) \to \Xi(n)$. Then for any $t \geq 0$, $e^{t\mathcal{P}}$ is CPTP if there exists a set of matrices $\{P_j \in \Xi(n)\}_j$ and a Hermitian H_{system} such that*

$$\mathcal{P}(\rho) = i[\rho, H_{system}] + \sum_j P_j \rho P_j^\dagger - \frac{1}{2}\left\{P_j^\dagger P_j, \rho\right\}_+. \tag{2.192}$$

Theorem 41 *For any $t \geq 0$,*

(i) $\mathcal{L}(\rho_{\mathcal{L}}) = 0$, $e^{t\mathcal{L}}$ is CPTP and $\mathcal{L}^\diamond(I) = 0$.

(ii) $e^{t\mathcal{L}^*}$ and $e^{t\mathcal{L}^{(s)}}$ are CPTP.

(iii) $\mathcal{L}^{(s)}$ *has a unique fixed point, that is* $\rho_{\mathcal{L}}$.

(iv) The spectrum of $\mathcal{L}^{(s)}$ *lies in* $[-\infty, 0]$.

(v) The spectral gap of $\mathcal{L}^{(s)}$ *is strictly greater than* 0*, which is denoted as* Υ *and equals to*

$$\Upsilon(\mathcal{L}^{(s)}) = \min_{\substack{\langle N,N\rangle_\sigma=1 \\ \langle N,\rho_{\mathcal{L}}\rangle_\sigma=0}} -\langle N, \mathcal{L}^{(s)}(N)\rangle_\sigma > 0. \tag{2.193}$$

Since $\mathcal{L}^{(s)}$ *is clear in the context we abbreviate* $\Upsilon(\mathcal{L}^{(s)})$ *as* Υ.

Proof: For (i): Besides, by Assumption 1 we have $\mathcal{L}(\rho_{\mathcal{L}}) = 0$. Besides

$$\mathcal{L}(N) = \sum_j L_j N L_j^\dagger - \frac{1}{2}\left\{L_j^\dagger L_j, N\right\}_+. \tag{2.194}$$

$$\mathcal{L}^\diamond(N) = \sum_j L_j^\dagger N L_j - \frac{1}{2}\left\{L_j^\dagger L_j, N\right\}_+. \tag{2.195}$$

One can directly check $\mathcal{L}^\diamond(I) = 0$. Besides $e^{t\mathcal{L}}$ is CPTP by Theorem 40 by setting $H_{system} = 0$.

For (ii): we have

$$\mathcal{L}^*(N) = \sigma^{-\frac{1}{2}}\mathcal{L}^\diamond\left(\sigma^{\frac{1}{2}} N \sigma^{\frac{1}{2}}\right)\sigma^{-\frac{1}{2}} \tag{2.196}$$

$$= \sum_j \sigma^{-\frac{1}{2}} L_j^\dagger \sigma^{\frac{1}{2}} \cdot N \cdot \sigma^{\frac{1}{2}} L_j \sigma^{-\frac{1}{2}} - \sigma^{-\frac{1}{2}} \frac{1}{2}\left\{L_j^\dagger L_j, \sigma^{\frac{1}{2}} N \sigma^{\frac{1}{2}}\right\}_+ \sigma^{-\frac{1}{2}} \tag{2.197}$$

$$= \sum_j \sigma^{-\frac{1}{2}} L_j^\dagger \sigma^{\frac{1}{2}} \cdot N \cdot \sigma^{\frac{1}{2}} L_j \sigma^{-\frac{1}{2}} - \frac{1}{2}\left(\sigma^{-\frac{1}{2}} L_j^\dagger L_j \sigma^{\frac{1}{2}} \cdot N + N \cdot \sigma^{\frac{1}{2}} L_j^\dagger L_j \sigma^{-\frac{1}{2}}\right). \tag{2.198}$$

We will prove by properly defining a Hermitian H_{system}, the $\mathcal{L}^*$ can be written in the form of Eq. (2.192) in Theorem 40, which will imply $\mathcal{L}^*$ is CPTP. Denote

$$O_j := \sigma^{-\frac{1}{2}} L_j^\dagger \sigma^{\frac{1}{2}}, \quad K := \frac{1}{2}\sigma^{-\frac{1}{2}} L_j^\dagger L_j \sigma^{\frac{1}{2}}, \tag{2.199}$$

$$V := \frac{1}{2}\sum_j O_j^\dagger O_j, \quad H_{system} := \frac{1}{2i}\left(K - K^\dagger\right). \tag{2.200}$$

Recall that $\mathcal{L}(\rho_{\mathcal{L}}) = 0$ and $\sigma = \rho_{\mathcal{L}}^{-1}$, we have

$$\sum_j L_j \sigma^{-1} L_j^\dagger - \frac{1}{2}\left(L_j^\dagger L_j \sigma^{-1} + \sigma^{-1} L_j^\dagger L_j\right) = 0. \tag{2.201}$$

$$\Rightarrow \sigma^{\frac{1}{2}}\left(\sum_j L_j \sigma^{-1} L_j^\dagger - \frac{1}{2}\left(L_j^\dagger L_j \sigma^{-1} + \sigma^{-1} L_j^\dagger L_j\right)\right)\sigma^{\frac{1}{2}} = 0. \tag{2.202}$$

$$\Rightarrow \sum_j \sigma^{\frac{1}{2}} L_j \sigma^{-\frac{1}{2}} \cdot \sigma^{-\frac{1}{2}} L_j^\dagger \sigma^{\frac{1}{2}} - \frac{1}{2}\left(\sigma^{\frac{1}{2}} L_j^\dagger L_j \sigma^{-\frac{1}{2}} + \sigma^{-\frac{1}{2}} L_j^\dagger L_j \sigma^{\frac{1}{2}}\right) = 0. \tag{2.203}$$

Thus

$$V = \frac{1}{2}\left(K^\dagger + K\right) \quad \Rightarrow \quad K = V + iH_{system}, \tag{2.204}$$

where the $\Rightarrow$ we use $H_{system} := \frac{1}{2i}\left(K - K^\dagger\right)$. One can verify

$$\mathcal{L}^*(N) = \sum_j O_j N O_j^\dagger - \left(KN + NK^\dagger\right) \tag{2.205}$$

$$= \sum_j O_j N O_j^\dagger - (VN + NV) + i[N, H_{system}] \tag{2.206}$$

$$= i[N, H_{system}] + \sum_j O_j N O_j^\dagger - \frac{1}{2}\left\{O_j^\dagger O_j, N\right\}_+ \tag{2.207}$$

where the first equality comes from Eqs. (2.198)(2.199)(2.200), the second equality comes from (2.204), and the last equality comes from definition of V in Eq. (2.200). One can check that H_{system} is Hermitian. Since $\mathcal{L}^{(s)} = \frac{1}{2}(\mathcal{L} + \mathcal{L}^*)$, from Eqs. (2.194)(2.207) we known that $\mathcal{L}^{(s)}$ can also be written as Eq. (2.192), where the Lindbladian jump operator is $\{\frac{1}{\sqrt{2}}L_j\} \cup \{\frac{1}{\sqrt{2}}O_j\}$. Thus by Theorem 40 we conclude that $e^{t\mathcal{L}^{(s)}}$ is CPTP for $t \geq 0$.

For (iii): From Assumption 1 we already have $\{\frac{1}{\sqrt{2}}L_j\}$ generates the full algebra. Thus so does $\{\frac{1}{\sqrt{2}}L_j\} \cup \{\frac{1}{\sqrt{2}}O_j\}$. By Theorem 20 we conclude that there exists a unique ρ such that $\mathcal{L}^{(s)}(\rho) = 0$. Besides, from Eq. (2.198) one can directly verify that $\mathcal{L}^*(\rho_{\mathcal{L}}) = 0$, and thus $\mathcal{L}^{(s)}(\rho_{\mathcal{L}}) = 0$.

For (iv): By definition of $\mathcal{L}^{(s)} : \mathcal{H}(n) \to \mathcal{H}(n)$ it is Hermitian w.r.t. $\langle , \rangle_\sigma$. Thus $\mathcal{L}^{(s)}$ is diagonalizable and the spectrum is real. With contradiction assume there exists an eigenstate $N \in \mathcal{H}(n), N \neq 0$ with eigenvalue $\lambda > 0$. Then $\left|e^{t\mathcal{L}^{(s)}}(N)\right|_1 = e^{\lambda t}\,|N|_1$ goes to infinity as $t \to \infty$, which contradicts with the fact that $e^{t\mathcal{L}^{(s)}}$ is CPTP from (ii), where CPTP map does not increase trace norm of Hermitian operators (Lemma 38 in Appendix 2.10).

For (v): By (iii)(iv) we know the spectral gap is strictly greater than 0, and the minimum eigenvector is $\rho_{\mathcal{L}}$ and has eigenvalue 0. Then Eq. (2.193) is just by definition of spectral gap. ■

Proof of Theorem 7

Proof:[of Theorem 7] All the properties of $\mathcal{L}$ we use are summarized in Assumption 1. Define

$$h_t = e^{t\mathcal{L}}(\rho) - \rho_{\mathcal{L}}.$$

Since $\mathcal{L}(\rho_{\mathcal{L}}) = 0$, one can check that

$$\frac{d}{dt}\langle h_t, h_t\rangle_\sigma = 2\left\langle h_t, \mathcal{L}^{(s)}(h_t)\right\rangle_\sigma. \tag{2.208}$$

Further note that since $e^{t\mathcal{L}}$ is trace-preserving from Theorem 41 (i), we have $tr(e^{t\mathcal{L}}(\rho)) = 1$. One can verify that $\langle h_t, \rho_{\mathcal{L}}\rangle_\sigma = 0$. Then by Eq. (2.208) and Eq. (2.193), we have

$$\frac{d}{dt}\langle h_t, h_t\rangle_\sigma = 2\left\langle \frac{h_t}{|h_t|_\sigma}, \mathcal{L}^{(s)}\left(\frac{h_t}{|h_t|_\sigma}\right)\right\rangle_\sigma \cdot \langle h_t, h_t\rangle_\sigma \tag{2.209}$$

$$\leq -2\Upsilon \cdot \langle h_t, h_t\rangle_\sigma. \tag{2.210}$$

Then by Gronwall's inequality we have

$$\langle h_t, h_t\rangle_\sigma \leq \langle h_0, h_0\rangle_\sigma \cdot \exp(-2\Upsilon t). \tag{2.211}$$

Recall that $h_0 = \rho - \rho_{\mathcal{L}}$. Thus we have

$$\left|e^{t\mathcal{L}}(\rho) - \rho_{\mathcal{L}}\right|_1 \leq 2^{n/2}\left|e^{t\mathcal{L}}(\rho) - \rho_{\mathcal{L}}\right|_\sigma \tag{2.212}$$

$$\leq 2^{n/2}\left|\rho - \rho_{\mathcal{L}}\right|_\sigma \cdot \exp(-\Upsilon t) \tag{2.213}$$

$$\leq 2^{n/2} \cdot \sqrt{tr(\sigma^{\frac{1}{2}}\rho\sigma^{\frac{1}{2}}\rho)} \cdot \exp(-\Upsilon t), \tag{2.214}$$

where the first inequality comes from connecting trace-norm and $|\cdot|_\sigma$ by Lemma 39 in Appendix 2.10. Note that $\sigma \geq I$ since $\sigma = \rho_{\mathcal{L}}^{-1}$ and $\rho_{\mathcal{L}}$ is a quantum state. The second inequality comes from Eq. (2.211). The last inequality comes from $\sigma = \rho_{\mathcal{L}}^{-1}$ and $tr(\rho - \rho_{\mathcal{L}}) = 0$ since both ρ and $\rho_{\mathcal{L}}$ are quantum states, and

$$|\rho - \rho_{\mathcal{L}}|_\sigma^2 = tr(\sigma^{\frac{1}{2}}(\rho - \rho_{\mathcal{L}})\sigma^{\frac{1}{2}}(\rho - \rho_{\mathcal{L}})) = tr(\sigma^{\frac{1}{2}}\rho\sigma^{\frac{1}{2}}\rho) - 1. \tag{2.215}$$

■

Chapter 3

GIBBS STATE PREPARATION FOR COMMUTING HAMILTONIAN

3.1 Introduction

Gibbs state preparation is a key computational technique used in physics, statistics, and many other scientific fields. Given a local Hamiltonian H and an inverse temperature β, the Gibbs state $\rho_{\beta H} \sim \exp(-\beta H)$ describes the thermal equilibrium properties of quantum systems at finite temperature, making them essential for studying the phase diagram, stability of topological quantum memory [Has11; LP13] as well as the thermalization process [RGE12; Mül+15]. In addition to physics, Gibbs state preparation also has found various applications in optimization [Bra+19; Van+17] and Bayesian Inference [Ami+18; HS83]. Various Gibbs state preparation algorithms (or *Gibbs samplers*) have been proposed, including approaches inspired by the Davies generator [Che+23; Gil+24b; RWW23; DLL24], the Metropolis-Hasting type method [JI24; Tem+11], and ones based on Grover amplification [PW09] and the Quantum Singular Value Transform [Gil+19].

The key requirement for a good Gibbs sampler is *fast mixing*, that is, the algorithm prepares the Gibbs state in *polynomial* time. Gibbs samplers for classical Hamiltonians have been studied for decades and fast mixing algorithms have been successfully designed for various scenarios. In particular, Glauber dynamics yield fast mixing Gibbs samplers for 1D systems at any constant temperature [GZ03; Hol85; HS89] and for 2D systems at high temperature [MO94]. On the other hand, for 2D systems like the Ising model, Glauber dynamics-based samplers are known to suffer critical slow downs and are slow mixing at low temperature [CCS87; Ces+96; Sch87]. Instead, the Swendsen-Wang algorithm [SW87; FGW23] and approaches based on Barvinok's method [Bor+20] were proved to achieve fast mixing for low temperature 2D systems.

Recent efforts on developing fast mixing Gibbs samplers for quantum Hamiltonians have largely focused on commuting local Hamiltonians (CLHs). CLHs are an important subclass of quantum Hamiltonian which exhibits non-trivial quantum phenomenon. Different from classical Hamiltonians whose eigenstates are computational basis, the eigenstates of a CLH instance can be highly entangled and

cannot be prepared by any constant depth quantum circuit, as is true for the famous example of Kitaev's Toric code [Kit03a]. Besides, it was shown that Gibbs sampling of CLHs at constant temperature remains classically hard [BCL24; RW24]. Nonetheless, several aforementioned fast mixing results for classical Hamiltonians have been successfully generalized to the CLH case. In particular, multiple results utilize the Davies generator [Dav76; Dav79], which represents a quantum Markov chain (Lindbladian) for thermalization process in the weak coupling limit. This has yielded fast mixing Gibbs samplers for 1D CLH at any constant temperature [Bar+23; KB16] and 2D CLH at high temperature [CRF20; KB16]. The fast mixing proofs are obtained by generalizing classical techniques for analyzing the mixing time of transition matrices [GZ03; Hol85; HS89; MO94; MO94] to analyzing the mixing time of the Davies generator. These generalizations are highly non-trivial and very technical since analyzing the spectrum of a quantum operator (in this case the Davies generator) is generally hard. In the low temperature regime, fast mixing Gibbs samplers for CLHs on two or higher dimensions are only known for the standard 2D Toric code [AFH09; Din+24].

In this work, we introduce a new approach which does not use the Davies generator. Instead, we design new Gibbs samplers for various CLHs by giving a reduction from quantum Gibbs state preparation to classical Gibbs sampling. Combined with the existing fast mixing results for classical Hamiltonians, our Gibbs sampler is able to replicate the state-of-the-art performances mentioned above. Furthermore, our algorithm can prepare low temperature Gibbs states as long as there exists a fast mixing Gibbs sampler for the corresponding classical Hamiltonians. Our reductions are summarized in the following two theorems. More details and comparisons between previous results and the performance of our Gibbs sampler are contained in Section 3.1.

Roughly speaking, we say that there is a *Gibbs sampling reduction* from a quantum Hamiltonian H to a classical Hamiltonian $H^{(c)}$ if, given the existence of an algorithm that performs Gibbs sampling for $(H^{(c)}, \beta)$ in time T, there exists a quantum algorithm preparing the quantum Gibbs state for (H, β) in time T plus a small overhead polynomial in the number of qubits.[1] First we notice that the Structure Lemma, which is the key technique used in studying the complexity of CLHs [BV03; IJ23; AKV18; Sch11], and directly gives the desired reduction for 2-local CLHs.

[1] In our case, we will obtain scaling like $T + O(n)$ or $T + O(n^2)$.

Theorem 42 (Informal version of Theorem 49) *There is a Gibbs sampling reduction from 2-local qudit commuting Hamiltonians to 2-local qudit classical Hamiltonians.*

For more physically motivated 4-local CLHs such as the Toric code, the Structure Lemma can no longer transform 4-local CLHs to classical Hamiltonians. Instead, by leveraging a symmetry in the eigenspaces, we demonstrate that an oblivious randomized correction approach yields the desired reduction for qubit CLHs in 2D, via generalizing a ground state preparation algorithm [AKV18] to Gibbs state preparation. The locality of the resulting classical Hamiltonian depends on whether there are classical qubits in the CLHs. Roughly speaking a qubit is classical if by choosing proper basis of this qubit, all terms look like $|0\rangle\langle 0| \otimes ... + |1\rangle\langle 1| \otimes ...$ on this qubit.

Theorem 43 (Informal version of Theorem 58 and 76) *There is a Gibbs sampling reduction from 4-local qubit 2D commuting local Hamiltonian H to qudit classical Hamiltonians. In particular,*

- *If there are no classical qubits with respect to terms in H, then the classical Hamiltonian is a 2-local qudit classical Hamiltonian on a planar graph.*
- *When there are classical qubits but all quantum terms (terms far away from classical qubits) are uniformly correctable, then the classical Hamiltonian is a $O(1)$-local qudit classical Hamiltonian.*

Note that in the above theorem the quantum Hamiltonian is on qubits while the classical Hamiltonian is on qudits. An example of a qubit 2D CLH without classical qubits is the defected Toric code, a generalization of the Toric code with arbitrary complex coefficients. We will give a technical overview based on the defected Toric code in Section 3.1.

Our reduction also gives a quantum analogy of Stockmeyer's result [Sto83] for the complexity of quantum approximate counting. In particular, a fundamental result of Stockmeyer [Sto83] states that classical approximate counting (approximating the partition function of a classical Hamiltonian) is contained in the complexity class $\mathbf{BPP}^{\mathbf{NP}}$. It is natural to conjecture that the quantum approximate counting is upper bounded by a complexity class like $\mathbf{BQP}^{\mathbf{QMA}}$, but few results are known. By

the connection between quantum approximate counting and the Gibbs state preparation [Bra+21][2], our reduction shows that for various CLHs, the corresponding quantum approximate counting problem is contained in $\mathbf{BQP}^{\mathsf{CS}}$, where CS is an oracle which can perform arbitrary classical Gibbs sampling.

Comparison to previous work.

Recall that most of the previous work on Gibbs samplers for CLHs are based on simulating the Davies generator, which is a Lindbladian closely related to the thermalization process. In this section, we give a detailed comparison between previous results and our result, demonstrating that instead of using the Davies generator, our reduction gives a new Gibbs sampler for CLHs directly utilizing fast mixing Gibbs samplers for classical Hamiltonians. In particular, our reduction is able to replicate state-of-the-art results and also derive new results. Our results are summarized in Figure 3.1. In this section we discuss related results, and a more thorough discussion on Gibbs state preparation can be found in [CRF20, Section 3.3].

Remark 1 *Due to the relationship between Davies generator and thermalization process, previous works analyzing the Davies generator [Bar+23; KB16; AFH09] also yield insights into how thermal noise influences the quantum systems. Our reduction does not cover this implication. The following comparison is only for the task of preparing Gibbs state.*

Mixing time	1D any temp	2D (2-local)	2D (4-local)
Previous work	$poly(\log n)$[1]	high temp: $poly(\log n)$[2] low temp: unknown	$poly(n)$ for high temp[3] $O(n^4)$ mixing time for TC [4]
Our results	$poly(\log n)$	high temp: $poly(\log n)$ **low temp: *$poly(n)$**	***$poly(n)$** **$O(n^2)$ for DTC**

Figure 3.1: Mixing time of Gibbs samplers for 1D and 2D CLHs at different temperatures (temp). TC and DTC refer to the standard Toric code and the more general defected Toric code respectively. Improved results are in **bold**. *The results marked "*" indicate we achieve this mixing time, when efficient samplers for the corresponding* classical *Hamiltonians are known.* Our $O(n^2)$ result for DTC is the total runtime rather than just mixing time. References: [1] [CRF20]; [2] [Koc+24; CRF20]; [3] [KB16]; [4] [Din+24].

Review of Markov chains, Lindbladians and mixing time We briefly review some key concepts. Consider an n-qubit local Hamiltonian H and an inverse

[2]Lemma 12 in [Bra+21], where the k-QMV can be estimated by measuring the Gibbs state .

temperature β. We assume $\beta \in O(1)$ unless further specified. We will first assume H is classical and introduce key concepts for classical Gibbs sampling. Then we will generalize to quantum Hamiltonians.

Suppose H is a classical Hamiltonian which diagonalizes in the computational basis, and the task is performing classical Gibbs sampling for (H, β). The target is the Gibbs distribution π which samples computational basis states $|x\rangle$ with probability proportional to $\exp(-\beta\langle x|H|x\rangle)$. The goal of classical Gibbs sampling is to design a classical process which drives any distribution ν to π. The commonly used method is the classical Metropolis algorithm [Met+53], which is a discrete-time Markov chain described by a transition matrix P, such that π is the unique fixed point of P, i.e. $P\pi = \pi$. The mixing time $t(\epsilon)$ is the time needed to get ϵ-close to the invariant distribution π with respect to 1-norm (the total variation distance), that is

$$t(\epsilon) := \min\{t : \|P^t\nu - \pi\|_1 \leq \epsilon, \forall \text{ distribution } \nu\}. \tag{3.1}$$

In addition to this discrete-time Markov chain, one can also design a continuous-time Markov chain, described by a generator matrix G such that π is the unique invariant distribution of G, i.e. $G\pi = 0$ or equivalently $e^{Gt}\pi = \pi, \forall t$. Similarly to above, the mixing time is defined to be

$$t(\epsilon) := \min\{t : \|e^{Gt}\nu - \pi\|_1 \leq \epsilon, \forall \text{ distribution } \nu\}. \tag{3.2}$$

- The Markov chain is *poly-time mixing*, or *fast mixing*, if the the spectral gap of P or G is $\Omega(1/poly(n))$, which implies $t(\epsilon) = poly(n) \times \log\frac{1}{\epsilon}$.
- The Markov chain is *rapid mixing* if it reaches the invariant distribution in a time which scales logarithmically with the system size, that is $t(\epsilon) = poly(\log n) \times \log\frac{1}{\epsilon}$. Rapid mixing is typically proved by bounding the log-Sobolev constant [GZ03] for the continuous-time chain.

When H is a quantum Hamiltonian, we wish to prepare a quantum Gibbs state, defined as

$$\rho_{\beta H} := \rho(H, \beta) := \frac{1}{\mathrm{tr}[\exp(-\beta H)]}\exp(-\beta H).$$

The goal is to design a quantum process which drives any quantum state σ to $\rho_{\beta H}$. One commonly used method is to design a Lindbladian $\mathcal{L}$ such that $\rho_{\beta H}$ is the unique fixed point of $\mathcal{L}$, i.e. $\mathcal{L}(\rho_{\beta H}) = 0$ or, equivalently, $e^{\mathcal{L}t}(\rho_{\beta H}) = \rho_{\beta H}, \forall t$.

The Lindbladian is the quantum analogy of a continuous-time Markov chain. The mixing time $t(\epsilon)$ is defined to be

$$t(\epsilon) := \min\{t : \|e^{\mathcal{L}t}(\sigma) - \rho_{\beta H}\|_1 \leq \epsilon, \forall\sigma\}. \tag{3.3}$$

The notion of poly-time mixing and rapid mixing is defined similarly to the classical setting. One can prepare the quantum Gibbs state on a quantum computer by Lindbladian simulation techniques [CW16; Che+23]. For simplicity we will assume that $\epsilon = 1/poly(n)$ for the remainder of the section.

Replication of poly-log time mixing algorithm for 1D commuting Hamiltonians. For 1D classical Hamiltonians, it is well-known that there is no *computational phase transition* [GZ03; Hol85; HS89]. As a result, for any constant temperature, Glauber dynamics is rapid mixing for all translation-invariant, 1D classical Hamiltonians with finite-range interactions.

A large body of previous work has focused on generalizing the rapid mixing results from classical Hamiltonians to quantum CLHs. In particular, for 1D CLHs, [KB16] proved that the Davies generator has a constant spectral gap and thus is fast (poly-time) mixing. Then Bardet *et.al.* [Bar+23] strengthened the result to obtain rapid mixing. Specifically, they proved that for any constant temperature, the Davis generator $\mathcal{L}$ for any finite-range, translation-invariant 1D CLHs is rapid mixing. This is proved by generalizing the classical technique of bounding the log-Sobolev constant [GZ03; Hol85; HS89; Zeg90] to CLHs. Note that this generalization is highly non-trivial since $\mathcal{L}$ is a quantum operator and analyzing its spectral gap and log-Sobolev constant are complicated. Combining this bound with quantum simulations of the Lindbladian, Bardet *et.al.* gives a quantum Gibbs sampler with runtime

$$poly(\log n) \times f_1,$$

where f_1 is the overhead brought by simulating the Lindbladian evolution.

In contrast to [Bar+23], which obtained a rapidly mixing Gibbs sampler by developing sophisticated techniques to bound the log-Sobolev constant of the Davies generator, our reduction gives a Gibbs sampler of similar performance by directly using classical results

Lemma 44 (Informal version of Lemma 50) *There is a Gibbs sampling reduction from any finite-range, translation-invariant (TI) qudit 1D CLHs, to the finite-range,*

TI 1D classical Hamiltonians. Combined with the rapid mixing Gibbs sampler for finite-range, TI 1D classical Hamiltonian at any constant temperature [GZ03; Hol85; HS89], we prepare the Gibbs state in time,

$$poly(\log n) \times f_2 + O(n).$$

Here f_2 is the overhead incurred by simulating the classical Markov chain. $O(n)$ is the time needed to prepare a constant depth quantum circuit arising from the Structure Lemma which implements the quantum-to-classical reduction.

High and low temperature Gibbs state for 2-local CLHs on 2D. Recall that β is the inverse temperature, and thus low temperature corresponds to large β. We begin with a literature review for classical Hamiltonians. Unlike 1D classical Hamiltonians where the Glauber dynamics is rapid mixing for any constant temperature, 2-local 2D classical Hamiltonians exhibit a constant-temperature computational phase transition. For example, the ferromagnetic 2D Ising model has a constant *critical inverse temperature* β_c, such that

- For $\beta < \beta_c$ the Glauber Dynamics is poly-time mixing [MO94; MO94].
- For $\beta \geq \beta_c$, the Glauber dynamics meets a critical slow down where the spectral gap of the Glauber Dynamics is smaller than $\exp(-\alpha\sqrt{n})$ for $\alpha > 0$ [CCS87; Ces+96; Sch87].

Similar results also hold for the Potts model [Ull12; GL16]. To understand this phase transition intuitively, note that the Glauber dynamics is a Markov chain with local update rules. Intuitively an algorithm using local updates is good at solving a "local" problem. In the high temperature region, most spins will interact effectively weakly thus the Gibbs state has little entanglement [Bak+24] and a local optimization suffices. However, in the low temperature region, there are strong correlations in large regions and thus the Gibbs state is highly non-local.[3] Thus, to prepare low temperature Gibbs state , one needs to carefully design Markov chains with *non-local* update rules, such as the cluster updates in the Swendsen-Wang algorithm [SW87]; or uses other methods such as the Barvinok's method [Bor+20].

[3] More precisely there is an equivalence between the mixing time and the spatial decay of correlation in the Gibbs measure [Dye+04; Ces01; SZ92a; SZ92b]. In some classical literature, decay of correlation is referred to as mixing condition [GZ03] or spatial mixing [Dye+04].

In the quantum case, to the best knowledge of the authors, all previous work on Gibbs state preparation for 2D CLHs has focused on the high temperature region. In particular,

- [KB16] showed that there is a constant β_1 such that for $\beta \leq \beta_1$ the Davies generator is poly-time mixing.
- [CRF20] showed that there is a constant β_2 such that for $\beta \leq \beta_2$[4], the Schmidt generator defined in [CRF20] is rapid mixing.

Both the Davies generator and the Schmidt generator for 2D CLHs with respect to local jump operators are local Lindbladians. A simple adaption of the classical proofs [CCS87; Ces+96; Sch87] will show that they are slow mixing for 2D systems at low temperature. That is, there exists a constant β_3 such that for $\beta \geq \beta_3$, the spectral gap of any $O(\log n)$-local Lindbladian (not necessarily the Davies generator) which fixes the Gibbs state of 2D Ising model at inverse temperature β has an exponentially-small spectral gap.

Our Gibbs sampler improves on the existing results in two main ways. First, in the high temperature region our reduction again gives a way to directly utilize classical results [Ces01] and obtain a Gibbs sampler of similar performance as the best prior work [CRF20] (i.e., rapid mixing), without involving heavy proofs for analyzing the log-Sobolev constant of the Schmidt generator like [CRF20].

Lemma 45 *There is a Gibbs sampling reduction from any* 2-local qudit 2D CLHs *to 2-local qudit 2D classical Hamiltonians. Thus for high enough temperature where there exists rapid mixing classical Gibbs samplers for the corresponding classical Hamiltonians (e.g. from [Ces01] or Chapter 9 of [GZ03]), we can prepare the Gibbs state for the 2D CLHs in time*

$$poly(\log n) \times f_2 + O(n)\,,$$

where f_2 is the overhead incurred by simulating the classical Markov chain.

Our second contribution is in the low temperature regime. Unlike [KB16; CRF20] which only work for the high temperature region, our reduction allows us to prepare low-temperature Gibbs states by utilizing classical techniques such as the Swendsen-Wang algorithm. To the best knowledge of the authors, prior work has not addressed

[4] We did not check whether β_1, β_2 and the later mentioned high temperatures are equal.

low-temperature Gibbs samplers for 2-local CLHs. As an example, with our reduction we can obtain the following result.

Lemma 46 (Informal version of Lemma 51) *There is a Gibbs sampling reduction from translation-invariant qubit (2-local) 2D CLHs to 2D Ising model with magnetic fields. Then one can prepare the Gibbs state for the corresponding CLH at low temperature in $poly(n)$ time whenever there are poly-time mixing Gibbs sampler for the corresponding Ising model at low temperature like [FGW23].*

A key feature of our reduction is that it is agnostic to the underlying classical Gibbs sampler. In the low temperature regime, when applied to qudit 2-local 2D CLHs with *large* constant qudit dimension d, the Swendsen-Wang algorithm will also mix slowly and have an exponentially-small spectral gap [Bor+99]. However, our reduction allows us to substitute in other samplers, such as the Gibbs sampler for the low temperature Potts model based on Barvinok's method, which remains poly-time for large d [Bor+20].

4-local, 2D commuting Hamiltonian. The best prior work is due to [KB16], who proved that for high enough temperature, the Davies generator is poly-time mixing for 4-local 2D CLHs. Unlike their result, the mixing time of our algorithm is dependent on the classical Hamiltonian produced by the reduction and thus our results are not directly comparable.

For the *standard* Toric code, a concurrent work [Din+24] showed that for any inverse temperature $\beta < +\infty$ (not necessarily constant), Lindbladian dynamics with nonlocal jump operators prepares the Gibbs state efficiently (for very low temperature, the mixing time is approximately $O(n^4)$). Our Gibbs sampler is based on different techniques and gives a $O(n^2)$-time Gibbs state preparation algorithm for the general defected Toric code at any non-zero temperature, where the defected Toric code is the Toric code with arbitrary coefficients. Our Gibbs sampler is based on generalizing the standard ground state preparation algorithm for the Toric code (which measures all stabilizers) to the task of Gibbs state preparation via an oblivious randomized correction technique. We will give a technical overview based on the example of defected Toric code[5] in Section 3.1 and 3.1. We remark that since our Gibbs sampler is not based on Lindbladian, our results do not offer additional insights into Lindbladian dynamics, unlike in [Din+24]. Another related work [GOL24] uses

[5] In fact, beyond the defected Toric code, our approach can also be applied to prepare the Gibbs states of other error-correcting codes, such as the 4D Toric code.

classical Monte Carlo methods to simulate the Gibbs states of t-doped stabilizer Hamiltonians, although without discussing convergence guarantees.

Besides the defected Toric code, our Theorem 43 also works for more general qubit CLHs and can prepare the corresponding Gibbs state as long as there exists efficient algorithm for the corresponding classical Gibbs sampling task.

In addition to the defected Toric code, Theorem 43 also works for more general families of qubit CLHs and can prepare the corresponding Gibbs state as long as there exists an efficient algorithm for the corresponding classical Gibbs sampling task.

Technical overview

Recall that in Theorem 42, we construct a Gibbs sampling reduction from 2-local qudit CLHs to 2-local qudit classical Hamiltonians,

Theorem 42 (Informal version of Theorem 49) *There is a Gibbs sampling reduction from 2-local qudit commuting Hamiltonians to 2-local qudit classical Hamiltonians.*

The proof is primarily based on the Structure Lemma [BV03; IJ23; AKV18; Sch11]. The Structure Lemma has been the principle tool in studying the complexity of CLHs and, intuitively, says that one can transform a 2-local qudit CLH $H^{(2)}$ to a 2-local qudit classical Hamiltonian $H^{(2c)}$ via a constant depth quantum circuit C_H. In other words, there is a *one-to-one* correspondence between the computational basis of the classical Hamiltonian $H^{(2c)}$ and the eigenstates of the quantum Hamiltonian $H^{(2)}$. By this observation, there is a simple procedure to sample from the Gibbs state of $H^{(2)}$. Frst, we take a sample $|\psi\rangle$ from the Gibbs distribution of $H^{(2c)}$ (via a classical Gibbs sampler). Then, applying a constant depth quantum circuit to $|\psi\rangle$ yields a sample from the Gibbs distribution of $H^{(2)}$.

For Hamiltonians of higher locality, the exact correspondence present in the 2-local case does not hold. Nonetheless, we show in Theorem 43 that we can extend our techniques beyond 2-local Hamiltonians. We demonstrate a reduction from Gibbs sampling of 4-local qubit CLHs in 2D to classical Gibbs sampling.

Theorem 43 (Informal version of Theorem 58 and 76) *There is a Gibbs sampling reduction from 4-local qubit 2D commuting local Hamiltonian H to qudit classical Hamiltonians. In particular,*

- *If there are no classical qubits with respect to terms in H, then the classical Hamiltonian is a 2-local qudit classical Hamiltonian on a planar graph.*
- *When there are classical qubits but all quantum terms (terms far away from classical qubits) are uniformly correctable, then the classical Hamiltonian is a $O(1)$-local qudit classical Hamiltonian.*

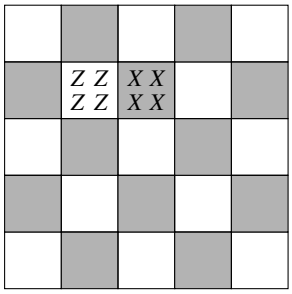
The case "without classical qubits" is the simpler setting. Still, even in this case, we can no longer straightforwardly apply the Structure Lemma as is possible for 2-local Hamiltonians. This is not due to a deficiency in our techniques; rather, 4-local Hamiltonians can exhibit topological order [Kit03a] and there cannot be a constant depth quantum circuit C_H as in the previous theorem. However, we observe that the eigenspace of qubit CLH is symmetric in some sense, and via an *oblivious randomized correction* technique we can adapt an algorithm for preparing ground state (as given in [AKV18]) to preparing a Gibbs state. In particular, [AKV18] proves that any 2D qubit CLH without classical qubits is equivalent to a defected Toric code permitting boundaries. That is, the "interior" terms look like Pauli X or Pauli Z terms and terms on the "boundary" have more freedom. The presence of boundaries makes it non-trivial to utilize this equivalence to design a Gibbs sampler for general 2D qubit CLH. We will use the defected Toric code as an example to explain our Gibbs sampler in Section 3.1.

When the initial Hamiltonian has classical qubits, the situation becomes more complex, as the connection from [AKV18] between 2D qubit CLH and the Toric code only applies when there are no classical qubits. This does not pose a problem in [AKV18] as they simply want to verify ground energy, and an **NP** prover can provide a *recursive* restriction of classical qubits consistent with some ground state. This effectively removes all classical qubits and the resulting Hamiltonian can be translated into a defected Toric code permitting boundaries.

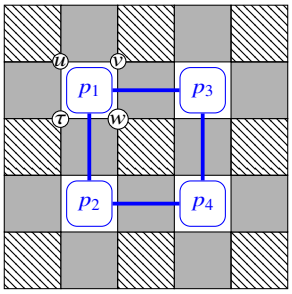
In our case, we would like to recover the *distribution* over eigenstates, and thus we cannot perform the same recursive restriction. Additionally, we need our reduction to be efficient and should not depend on the power of an prover. We develop a propagation lemma to characterize the limits of the recursive restriction. Combined with an assumption that all fully quantum terms[6] are uniformly correctable (see Assumption 2), we will argue that the statement of [AKV18] can be modified to

[6]Intuitively, this is the set of terms which remains quantum under any recursive restriction of the classical qubits; see Definition 75.

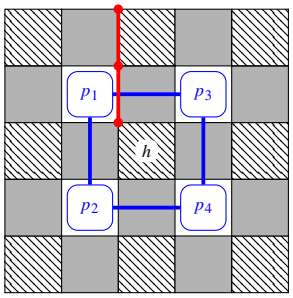
obtain a Gibbs sampling reduction from any 2D qubit CLH to a constant-locality classical Hamiltonian.



(a) Partition the plaquettes as Black $\mathcal{B}$ and $\mathcal{W}$. Put Z terms on white plaquettes and put X terms on black plaquettes then we get the defected Toric code H_{DT}.



(b) Remove the white terms on the odd lines, H_{DT} will become $H^{(2)}_{DT}$ which can be viewed as a qudit 2-local CLH.



(c) A correction operator $L_h = X^{\otimes 3}$ (in red) for the removed white term h.

Figure 3.2: Illustration of Gibbs states preparation for the Toric code.

Extensions to more general 2D Hamiltonians For simplicity, the theorems above are proved in the setting when the underlying Hamiltonian is placed on a $2D$ lattice. However, in [AKV18] the authors consider a more general setting of Hamiltonians on *polygonal complexes*. A straightforward generalization of our proofs works in this setting as well.

Case study: the defected Toric code

To illustrate our Gibbs sampler for 2D qubit CLH, that is Theorem 43, we consider the restricted setting of the defected Toric code. In the remainder of this section, we assume that the inverse temperature is $\beta < +\infty$ (not necessarily a constant). We will formally define the defected Toric code and first give a $O(n^2)$-time algorithm to prepare its Gibbs states via an oblivious randomized correction idea. This algorithm is specific to the defected Toric code. Then we describe a slightly different algorithm which is not as fast as the first algorithm, but by using the tools from [AKV18] it can be extended to prepare Gibbs state for general qubit 2D CLHs.

The defected Toric code H_{DT} is embedded on a 2D, $L \times L$ square lattice, with qubits placed on the vertices. Terms are grouped into "black" terms $\mathcal{B}$ and "white" terms $\mathcal{W}$, as in Figure 3.2a. Formally, we define

$$H_{DT} = \sum_{p \in \mathcal{B}} c_p X^p + \sum_{p \in \mathcal{W}} c_p Z^p, \tag{3.4}$$

where X and Z are the standard Pauli X and Z operators. For a given term p acting on qubits $q_1, \ldots, q_4$, X^p denotes $X_{q_1} \otimes \cdots \otimes X_{q_4}$ (and same for Z^p). The coefficients c_p can be any real number (whereas $c_p = -1$ in the standard Toric code).

A $O(n^2)$-time algorithm for the defected Toric code.

First, we explain the $O(n^2)$-time algorithm to prepare the Gibbs states of the defected Toric code. Let us temporarily assume that we have removed two white terms p_1, p_2 and two black terms p_a, p_b in H_{DT} to obtain a new Hamiltonian H'_{DT} with holes in the lattice corresponding to the removed plaquettes $S := \{p_1, p_2, p_a, p_b\}$. For any $p \notin S$, there is a correction operator L_p (realized as a tensor product of Pauli X or Pauli Z's), which anti-commutes with p and commutes with all other terms not in S. To prepare the Gibbs states for the punctured Hamiltonian H'_{DT}, we initialize our state as the maximally mixed state, then sequentially measure and correct each plaquette term p in H'_{DT}. That is, if we measure the plaquette operator p and get measurement outcome $\lambda \in \{+c_p, -c_p\}$, then we perform the following oblivious randomized correction:

- With probability $prob := \frac{\exp(-\beta\lambda)}{\exp(\beta\lambda)+\exp(-\beta\lambda)}$ we do nothing.
- With probability $1 - prob$ we apply the correction operation L_p.

The algorithm correctly prepares the Gibbs states of H'_{DT} because L_p *bijectively* maps the eigenspace associated with measurement outcome $+c_p$ to that of $-c_p$ and vice versa. A more detailed description of this procedure and proof of its correctness can be found in Section 3.4. To obtain the Gibbs state for the defected Toric code, one can perform a similar measure-and-correct operation on the current state but for p_1, p_2 simultaneously,[7] followed by a symmetric correction for p_a, p_b.

We emphasize that the above algorithm *does not* work for zero temperature (when $\beta = +\infty$).[8] Furthermore, the above process only performs the correction *once* for

[7] More specifically, there is a "string" correction operator L_{p_1,p_2} which anti-commutes with p_1, p_2 and commutes with any other terms. Then one does the following on the current state (1) Measure the Gibbs states of H'_{DT} with respect to commuting observables p_1, p_2 at the same time. (2) Suppose the measurement outcome of p_1, p_2 is λ_1, λ_2. Define $\lambda := \lambda_1 + \lambda_2$ and perform a similar randomized correction by using L_{p_1,p_2}.

[8] Besides, we emphasize that this algorithm is unique to the defected Toric code. Even for ground energy, other 2D CLH like the 2D Ising model with arbitrary coefficients is hard to compute, whereas the ground energy (ground state) of the defected Toric code can be easily computed (prepared), as explained in [AKV18, Appendix E].

each plaquette term, and thus should not be interpreted as an iterative, randomized Accept/Reject process as done in the general Metropolis algorithm.

A generalizable algorithm

For general 2D qubit CLH (without classical qubits), not every plaquette term has a correction operator; in fact, the presence of a correction operator qualitatively characterizes the correctable interior terms and the non-correctable boundary terms. The proof that the interior terms are correctable uses techniques from [AKV18], and the exterior terms are handled by a reduction to classical Gibbs sampling. The details are as below. For simplicity, we assume in this section that H_{DT} is embedded on a plane rather than torus, and the initial boundary of the lattice naturally plays the roles of the puncture terms S in H_{DT}.

Reduction to Classical Hamiltonian As in [AKV18] the first step is to remove enough terms such that the resulting Hamiltonian can be viewed as 2-local. In the case of H_{DT}, we can simply remove alternating rows of white terms, as in Figure 3.2b. Finally, grouping all qubits on a white term as a single 2^4-dimensional qudit, we see that the white terms become 1-local and the black terms are all 2-local. In Figure 3.2b, we group qubits u, v, w, τ to form the qudit p_1. Similarly we form the qudits p_2, p_3, p_4. Then, the black term to the right of qudit p_1 becomes 2-local, acting on p_1 and p_3. Call this 2-local Hamiltonian $H^{(2)}_{DT}$. The Structure Lemma of [BV03] gives a way to transform $H^{(2)}_{DT}$ to a 2-local classical Hamiltonian. By working out the details (see Section 3.6), it turns out that in this 2-local classical Hamiltonian we obtain three distinct "types" of terms:

- h_{vert}, 2-local terms corresponding to black terms acting on vertically arranged white terms (e.g. between p_1 and p_2),
- h_{horiz}, 2-local terms corresponding to black terms acting on horizontally arranged white terms (e.g. between p_1 and p_3), and
- h_w, 1-local terms corresponding to the white terms.

The final classical Hamiltonian is then

$$H^{(2c)}_{DT} = \sum_{\text{vertical} p_i, p_j} (h_{\text{vert}})_{p_i,p_j} + \sum_{\text{horizontal} p_i, p_j} (h_{\text{horiz}})_{p_i,p_j} + \sum_{p} (h_w)_p.$$

Preparation of Quantum Gibbs State. So far, we've removed terms from H_{DT} to obtain $H_{DT}^{(2)}$, then argued that we can view this as a classical Hamiltonian $H_{DT}^{(2c)}$. Assume we are able to perform classical Gibbs sampling at a given temperature on $H_{DT}^{(2c)}$. To obtain a sampler for our original Hamiltonian, we need to reverse each step of the reduction. First, since the transformation from $H_{DT}^{(2)}$ to $H_{DT}^{(2c)}$ is via a low-depth quantum circuit, we can easily obtain a Gibbs state of $H_{DT}^{(2)}$ from the Gibbs state on $H_{DT}^{(2c)}$. The primary challenge is to correct for the terms we have removed to make H_{DT} 2-local.

Suppose we want to correct for a remove white term $p \in \mathcal{W}$. We first measure the current state ψ with respect to the term p. If we only need to obtain some state with the correct eigenvalues, whenever we obtain an incorrect outcome, we could simply perform the correction operator L_p depicted in Figure 3.2c. To obtain L_p, we find a path from a corner of p to the boundary of the lattice, and apply a Pauli X on each qubit along the path. However, this does not immediately work when we are trying to sample from the Gibbs distribution.

Denote $\Pi_{+c_p}^p |\psi\rangle$ be state if we get measurement outcome $+c_p$ when measuring p. Similarly for $\Pi_{-c_p}^p |\psi\rangle$. There are two challenges in preparing the Gibbs state. First, we need to maintain the proper distribution over $\Pi_{+c_p}^p |\psi\rangle$ and $\Pi_{-c_p}^p |\psi\rangle$. Second, applying the correction L_p *after measuring* $\Pi_{+c_p}^p$ may not yield $\Pi_{-c_p}^p |\psi\rangle$, i.e.

$$L_p \Pi_{+c_p}^p |\phi(\boldsymbol{y})\rangle \not\propto \Pi_{-c_p}^p |\phi(\boldsymbol{y})\rangle . \tag{3.5}$$

Nonetheless, we show that this can be done via an *oblivious randomized correction* technique. That is, based on the measurement outcome, we apply the correction operation L_p with some probability μ. At a high level, the correctness of the idea comes from the symmetry of the eigenspace; despite Equation (3.5), we *do have* that

$$L_p \Pi_{+c_p}^p \Pi_{\lambda(\psi)} \Pi_{+c_p}^p L_p = \Pi_{-c_p}^p \Pi_{\lambda(\psi)} \Pi_{-c_p}^p$$

where $\Pi_{\lambda(\psi)}$ is an eigenspace of the *non-removed* operators corresponding to $|\psi\rangle$. We will leverage this fact by applying a correction uniformly across this eigenspace, and the correction probability μ will depend only $\Pi_{\lambda(\psi)}$ rather than $|\psi\rangle$ itself.

Conclusion and future work

In this manuscript, we give a reduction from Gibbs state preparation for various families of CLHs to the task of Gibbs sampling for classical Hamiltonians. In particular, based on the Structure Lemma we show that there is a Gibbs sampling

reduction from 2-local qudit CLHs to 2-local qudit classical Hamiltonians. Based on the symmetry in qubit CLH and the idea of oblivious randomized correction we give a Gibbs sampling reduction from various 2D qubit CLH to qudit classical Hamiltonians. This approach yields a Gibbs sampler based on techniques very different than those traditionally used, such as analyzing the Davies generator. We also demonstrate that combined with existing fast mixing results for classical Hamiltonians, our Gibbs sampler matches the performance of state-of-the-art results in [CRF20; Bar+23; KB16].

A natural direction to explore is whether our reduction can be generalized to other CLHs, especially those for which the complexity of proving ground energy is in **NP**, such as the factorized qudit CLH on 2D lattice [IJ23], the factorized CLH on any geometry [BV03] and the qutrit CLH on 2D [IJ23].

Our work also gives an interesting characterization for the complexity of quantum approximate counting for specific CLHs. It is well-known that classical approximate counting is in $\mathbf{BPP}^{\mathbf{NP}}$ [Sto83]. Due to the connection between quantum approximate counting and the Gibbs state preparation [Bra+21], our work shows that quantum approximate counting for various CLHs is contained in the complexity class $\mathbf{BQP}^{\mathsf{CS}}$, where CS is an oracle which can perform arbitrary classical Gibbs sampling. It would be interesting to explore whether there exist other families of quantum Hamiltonians where one can also upper bound the complexity of quantum approximate counting by $\mathbf{BQP}^{O}$ for some oracle O which is weaker than quantum approximate counting.

3.2 Preliminary

Notation

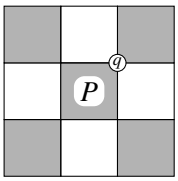
For two operators h and h', we use $[h, h']$ to denote the commutator $hh' - h'h$. We say that h and h' commutes if $[h, h'] = 0$. Given two n-qubit quantum states ρ and σ, we use $\|\rho - \sigma\|_1 := \frac{1}{2}\mathrm{tr}(|\rho - \sigma|)$ to denote their trace distance. Given two probability distributions $\mathcal{D}_1$ and $\mathcal{D}_2$ over $\{0,1\}^n$, we use $\|\mathcal{D}_1 - \mathcal{D}_2\|_1$ to denote the total variation distance, that is $\|\mathcal{D}_1 - \mathcal{D}_2\|_1 = \frac{1}{2}\sum_{x\in\{0,1\}^n} |\mathcal{D}_1(x) - \mathcal{D}_2(x)|$, where $\mathcal{D}_i(x)$ is the probability of sampling x in distribution $\mathcal{D}_i$. Given a graph $G = (V, E)$, for any vertex $v \in V$, we use $N(v)$ to denote the set of vertices which are adjacent to v (excluding v). For $v, w \in V$, we use $\{v, w\}$ and $\langle v, w\rangle$ for unordered set and ordered set respectively. For a positive integer $m \in \mathbb{N}$, we use $[m]$ to denote $\{1, 2..., m\}$.

Formal problem definitions

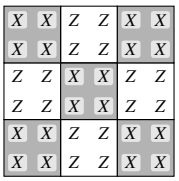
k-local Hamiltonians. We say an n-qudit Hermitian operator H is a k-local Hamiltonian, if $H = \sum_{i=1}^{m} h_i$ for $m = poly(n)$, and each h_i only acts non-trivially on at most k qudits. We allow different qudits to have different dimensions.

For the special case of $k = 2$, one can define H via a graph $G = (V, E)$. That is, on each vertex there is a qudit, and on each edge $\{v, w\}$ there is a Hermitian term h^{vw}, such that

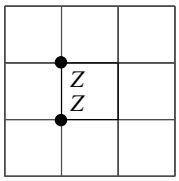
$$H_G = \sum_{\{v,w\} \in E} h^{vw}.$$



(a) 2D Hamiltonian



(b) Toric code



(c) 2D Ising Model

Figure 3.3: Examples of k-local Hamiltonians

2D Hamiltonians. Consider a 2D lattice $G = (V, E)$ as in Figure 3.3a, where qudits are placed on vertices. As above, we can define a 2-local Hamiltonian on G via $H_G = \sum_{\{v,w\} \in E} h^{v,w}$. We can also define a 4-local Hamiltonian over the 2D lattice by associating a Hermitian term to each plaquette P. We use $v \in P$ to denote that vertex v is in the plaquette P. With some abuse of notations, we also use v and P to denote the corresponding qubit and the Hermitian term. The 2D, 4-local Hamiltonian on the lattice is given by

$$H = \sum_{P} P\,.$$

For many of the proofs in this work, it will useful to partition the plaquette terms $\{P\}_P$ into a set of "black" terms $\mathcal{B}$ and "white" terms $\mathcal{W}$ by viewing the 2D lattice as a chess board, as in Figure 3.3a.

There is another natural notion of a 2D local Hamiltonian where the qudits are placed on the edges and Hermitian terms are corresponding to plaquettes and stars; this is the version which is primarily considered in [AKV18]. However, the two settings (where qudits are on vertices or are on edges) are equivalent when the underlying graph is the 2D square lattice (See Appendix C in [IJ23]).

Commuting and classical Hamiltonians. We say a k-local Hamiltonian $H = \sum_{i=1}^{m} h_i$ is a *commuting local Hamiltonian* (CLH) if $[h_i, h_j] = 0, \forall i, j$. Whenever we have a Hamiltonian $H = \sum_P P$ defined over the plaquettes of a 2D lattice, we have $[p, p'] = 0 \; \forall p, p'$. We say a k-local Hamiltonian $H = \sum_{i=1}^{m} h_i$ is *classical*, if each h_i is diagonalized in the computational basis.

Defected Toric code and 2D Ising model. Here we give two examples of qubit CLHs on 2D. For a vertex v in plaquette p, we use Z_v^p, X_v^p to denote the Pauli Z and Pauli X operator on the qubit v. When v uniquely identifies a vertex we abbreviate Z_v^p as Z_v and similarly for other Pauli operators. For a plaquette term p, we define $Z^p := \otimes_{v \in p} Z_v$ and X^p similarly.

As shown in Figure 3.3b, the *defected Toric code* is defined as

$$H = \sum_{p \in \mathcal{W}} c_p Z^p + \sum_{p \in \mathcal{B}} c_p X^p,$$

where $c_p \in \mathbb{R}$ can be arbitrary. The *standard Toric code* is a special case when all $c_p = -1$.

Denote the 2D lattice as $G = (V, E)$. As in Figure 3.3c, the *ferromagnetic 2D Ising model* is a 2-local Hamiltonian

$$H = \sum_{\{u,v\} \in E} Z_u \otimes Z_v.$$

The 2D Ising model is a classical Hamiltonian whose eigenstates are all computational basis. The defected Toric code is not a classical Hamiltonian. The ground state of the standard Toric code is highly entangled and cannot be prepared by any constant depth quantum circuit.

Gibbs state preparation. Given a k-local Hamiltonian $H = \sum_i h_i$, an inverse temperature $\beta < +\infty$, the *Gibbs state* with respect to (H, β) is defined as

$$\rho(H, \beta) = \frac{1}{\mathrm{tr}(\exp(-\beta H))} \exp(-\beta H). \tag{3.6}$$

Given $\epsilon > 0$, we say an algorithm $\mathcal{A}$ prepares $\rho(H, \beta)$ within precision ϵ, if $\mathcal{A}$ outputs a state ρ such that

$$\|\rho - \rho(H, \beta)\|_1 \leq \epsilon. \tag{3.7}$$

When H is classical, the *classical Gibbs distribution* with respect to (H, β) is denoted as $\mathcal{D}_{\beta H}$, which samples a classical string $x \in \{0,1\}^n$ with probability $\exp(-\beta\langle x|H|x\rangle)/\mathrm{tr}(\exp(-\beta H))$. We say an algorithm $\mathcal{A}$ performs classical Gibbs sampling $\mathcal{D}_{\beta H}$ with precision ϵ, if $\mathcal{A}$ outputs a distribution $\mathcal{D}$ such that

$$\|\mathcal{D} - \mathcal{D}_{\beta H}\|_1 \leq \epsilon. \tag{3.8}$$

Note that Eq. (3.8) is equivalent to Eq. (3.7) when ρ and $\rho(H, \beta)$ are diagonal matrices.

3.3 Reduction for 2-local qudit CLHs

In this section, we prepare the Gibbs state for qudit 2-local CLHs. In particular, in Section 3.3 we will prove the Gibbs sampling reduction for general 2-local qudit CLHs in Theorem 49. Then in Section 3.3 we will give several examples, whose proofs are put into Appendix 3.7.

General case

Recall that a 2-local CLHs $H_G^{(2)}$ is defined on a graph $G = (V, E)$, where

$$H_G^{(2)} = \sum_{\{v,w\}\in E} h^{vw}.$$

Here $\{h^{vw}\}_{\{v,w\}\in E}$ are Hermitian terms and commute with each other. The superscript (2) is to emphasize that the Hamiltonian is 2-local. In this section, we assume on each vertex there is a *qudit* rather than a qubit, and we allow G to be an arbitrary graph rather than just a 2D lattice.

The Gibbs sampling reduction for 2-local CLHs comes from the Structure Lemma, which was originally developed by [BV03] to study the computational complexity of commuting Hamiltonians. A constructive proof of the Structure Lemma can be found in Section 7.3 of [Gha+15]. Intuitively, the Structure Lemma says that one can decouple all commuting 2-local terms. This will allow us to identify eigenstates of a 2-local CLH $H_G^{(2)}$ with the computational basis of a classical Hamiltonian $H_G^{(2c)}$ defined later. In addition, each such eigenstate can be prepared by a constant depth quantum circuit. Thus, to prepare the Gibbs state of $H_G^{(2)}$, it suffices to first do classical Gibbs sampling for $H_G^{(2c)}$, yielding a distribution over computational bassi states, then prepare the corresponding eigenstate of $H_G^{(2)}$ indexed by a sampled basis state via a constant depth quantum circuit.

We first give the formal statement of the Structure Lemma.

Lemma 47 (Rephrasing of the Structure Lemma [BV03]) *Consider a vertex $v \in V$ and denote the Hilbert space of the qudit on v as $\mathcal{H}^v$. Consider the commuting Hermitian terms $\{h^{vw}\}_{w\in N(v)}$. There exists a direct sum decomposition of $\mathcal{H}^v$,*

$$\mathcal{H}^v = \bigoplus_{j_v=1}^{J_v} \mathcal{H}^v_{j_v}, \tag{3.9}$$

such that $\forall j_v$, all terms $\{h^{vw}\}_{w\in N(v)}$ keeps the subspace $\mathcal{H}^v_{j_v}$ invariant. Furthermore, each $\mathcal{H}^v_{j_v}$ has a tensor product factorization:

$$\mathcal{H}^v_{j_v} = \bigotimes_{w\in N(v)} \mathcal{H}^{\langle v,w\rangle}_{j_v}, \tag{3.10}$$

such that for all neighbors $w \in N(v)$, the term $h^{vw}|_{j_v}$ (the restriction of h^{vw} onto $\mathcal{H}^v_{j_v}$) acts non-trivially only on $\mathcal{H}^{\langle v,w\rangle}_{j_v}$, i.e.

$$h^{vw}|_{j_v} \subseteq \left(\bigotimes_{u\in N(v)/\{w\}} \mathcal{I}\left(\mathcal{H}^{\langle v,u\rangle}_{j_v}\right)\right) \otimes \mathcal{L}\left(\mathcal{H}^{\langle v,w\rangle}_{j_v} \otimes \mathcal{H}^w\right), \tag{3.11}$$

where $\mathcal{I}(\mathcal{H})$ is the identity operator on space $\mathcal{H}$, and $\mathcal{L}(\mathcal{H})$ is the set of all linear operators on $\mathcal{H}$.

The Structure Lemma can be understood via Figure 3.4. We can understand Equation (3.10) as the following: by choosing a proper local basis for the Hilbert space $\mathcal{H}^v_{j_v}$, it is equivalent to the Hilbert space of $|N(v)|$ distinct new qudits.

If for every qudit v, one applies Lemma 47 and chooses an index $j_v \in [J_v]$ and corresponding subspace $\mathcal{H}^v_{j_v}$, this will decouple all terms in H_G. Each term h^{vw} restricted to the subspaces $\mathcal{H}^v_{j_v} \otimes \mathcal{H}^v_{j_w}$ will act on distinct qudits and

$$h^{vw} \in \mathcal{L}\left(\mathcal{H}^{\langle v,w\rangle}_{j_v} \otimes \mathcal{H}^{\langle w,v\rangle}_{j_w}\right).$$

To define the corresponding classical Hamiltonian $H^{(2c)}_G$, we use the indices $\{j_v\}_{v\in V}$ to index the eigenstates of $H^{(2)}_G$. Denote

$$h^{vw}|_{j_v j_w} := \text{ restriction of } h^{vw} \text{ onto } \mathcal{H}^{\langle v,w\rangle}_{j_v} \otimes \mathcal{H}^{\langle w,v\rangle}_{j_w}.$$

Note that eigenstates of $h^{vw}|_{j_v j_w}$ might not be computational basis states (and in particular could be entangled). Nonetheless, we have shown that under the restriction corresponding to $\{j_v\}_{v\in V}$, all terms $h^{vw}|_{j_v,j_w}$ act on distinct qudits and

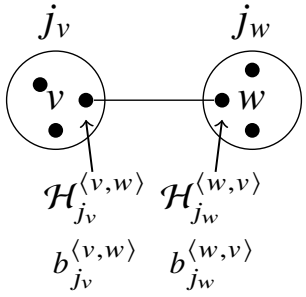


Figure 3.4: Illustration of decoupling the commuting terms. The figure contains two vertex v (on the left) and w (on the right). The Structure Lemma says that after choosing subspace j_v for $\mathcal{H}^v$ and j_w for $\mathcal{H}^w$, all terms are decoupled, that is they act on different qudits. More specifically, the qudits on v (the big ○) can be interpreted as the tensor product of several qudits of smaller dimension (the small •). Similarly for the qudit on w. The term h^{vw} only acts on the qudits correspond to $\{v, w\}$, that is $\mathcal{H}_{j_v}^{\langle v,w\rangle}$ and $\mathcal{H}_{j_w}^{\langle w,v\rangle}$, which are associated with edge $\{v, w\}$ and are not touched by any other terms. $b_{j_v}^{\langle v,w\rangle}$ and $b_{j_w}^{\langle w,v\rangle}$ are notations for the computational basis states for $\mathcal{H}_{j_v}^{\langle v,w\rangle}$ and $\mathcal{H}_{j_w}^{\langle w,v\rangle}$ respectively.

we can use the computational basis to index the eigenstates. As shown in Figure 3.4, let $D_{j_v}^{\langle v,w\rangle} := \dim(\mathcal{H}_{j_v}^{\langle v,w\rangle})$ and write the basis of each subspace $\mathcal{H}_{j_v}^{\langle v,w\rangle}$ as $\left|b_{j_v}^{\langle v,w\rangle}\right\rangle$ where $b_{j_v}^{\langle v,w\rangle}$ ranges over $[D_{j_v}^{\langle v,w\rangle}]$. Then $\bigotimes_{w\in N(v)} \left|b_{j_v}^{\langle v,w\rangle}\right\rangle$ is a computational basis state of $\mathcal{H}_{j_v}^{v}$. For each edge $\{v, w\} \in E$, the term $h^{vw}|_{j_v j_w}$ is Hermitian and thus can be diagonalized; the computational basis states $\left|\boldsymbol{b}_{j_v j_w}^{vw}\right\rangle$ are used to index the eigenstates. Thus, a basis for the full eigenspace of $h^{v,w}|_{j_v,j_w}$ is given by

$$\left|\boldsymbol{b}_{j_v j_w}^{vw}\right\rangle := \left|b_{j_v}^{\langle v,w\rangle}, b_{j_w}^{\langle w,v\rangle}\right\rangle, \quad \boldsymbol{b}_{j_v j_w}^{v,w} \in [D_{j_v}^{\langle v,w\rangle} \times D_{j,w}^{\langle w,v\rangle}]\,. \tag{3.12}$$

Given an index $\boldsymbol{b}_{j_v j_w}^{v,w}$, the corresponding eigenstate is denoted $\psi(\boldsymbol{b}_{j_v j_w}^{vw})$ and the eigenvalue $\lambda(\boldsymbol{b}_{j_v j_w}^{vw})$. The classical Hamiltonian is defined by substituting the eigenstate with its index, that is

$$H_G^{(2c)} := \sum_{\{v,w\}\in E} \sum_{j_v,j_w} \sum_{\boldsymbol{b}_{j_v j_w}^{vw}} \lambda(\boldsymbol{b}_{j_v j_w}^{vw}) \left|\boldsymbol{b}_{j_v j_w}^{vw}\right\rangle \left\langle \boldsymbol{b}_{j_v j_w}^{vw}\right|. \tag{3.13}$$

Following the usual convention, each term in the summand is implicitly padded with identities as necessary.

In this way, the eigenstates of $H_G^{(2c)}$ are given by specifying an index j_v for each vertex $v \in V$, then a basis state $b_{j_v j_w}^{\langle v,w\rangle}$ for each of the decoupled Hilbert spaces

$\mathcal{H}_{j_v}^{\langle v,w\rangle} \otimes \mathcal{H}_{j_w}^{\langle w,v\rangle}$

$$\boldsymbol{j} := \{j_v\}_v, \tag{3.14}$$

$$\boldsymbol{b_j} := \{b_{\boldsymbol{j}_v,\boldsymbol{j}_w}^{\langle v,w\rangle}\}_{\langle v,w\rangle}. \tag{3.15}$$

Thus, by construction, we have the following.

Lemma 48 *$H_G^{(2c)}$ is 2-local classical Hamiltonian on the graph G.*

We can also easily map eigenstates of $H_G^{(2c)}$ to eigenstates of $H_G^{(2)}$ via

$$\lambda(\boldsymbol{b_j}) := \sum_{\{v,w\}\in E} \lambda(\boldsymbol{b}_{j_v j_w}^{vw}) \tag{3.16}$$

$$\left|\psi(\boldsymbol{b_j})\right\rangle := \bigotimes_{\{v,w\}\in E} \left|\psi(\boldsymbol{b}_{j_v j_w}^{vw})\right\rangle , \tag{3.17}$$

and any classical Gibbs sampling procedure for $H_G^{(2c)}$ yields a Gibbs sampler for $H_G^{(2)}$.

Theorem 49 *For any inverse temperature β, if one can do classical Gibbs sampling w.r.t $\left(H_G^{(2c)},\beta\right)$ within precision ϵ in classical time T, then one can prepare the quantum Gibbs state w.r.t. $\left(H_G^{(2)},\beta\right)$ within precision ϵ in quantum time $T + O(m)$, where m is the the number of edges in graph G, by firstly using the classical Gibbs sampling w.r.t. $\left(H_G^{(2c)},\beta\right)$ to sample the index $\boldsymbol{b_j}$, then prepare the product state $\left|\psi(\boldsymbol{b_j})\right\rangle$ in time $O(m)$.*

Proof: It suffices to notice that by construction, $\boldsymbol{b_j}$ indexes the eigenvector of $H_G^{(2)}$ of eigenvalue $\lambda(\boldsymbol{b_j})$, that is $\left|\psi(\boldsymbol{b_j})\right\rangle$. ∎

Examples

In this section, we write down the Gibbs sampling reduction for some specific Hamiltonians as illustrative examples. All the proofs are deferred to Section 3.7.

We first consider an n-qudit Hamiltonian on a 1D chain

$$H = \sum_i h_i, \tag{3.18}$$

we say that H is r-range if h_i only acts non-trivially on qudits $i, i+1, .., i+r-1$. For simplicity we assume n is an integer multiple of r and the qudit dimension

d is a power of 2. We say that H is finite-range if r is a constant, and H is translation-invariant if all the terms h_i are the same.

By coarse-graining H, we can always assume H is 2-local: group each consecutive set of r qudits as a new qudit so that each h_i acts non-trivially on at most two (grouped) qudits. For each pair of new qudits $\{j, j+1\}$, we associate the new term $H_{j,j+1}$, which is a sum of all terms from H acting on the corresponding qudits. Thus H can be viewed as a 2-local qudit Hamiltonian on 1D written as $H = \sum_j H_{j,j+1}$. Note the terms $H_{j,j+1}$ can also be made translation-invariant.

Lemma 50 *Consider a finite-range translation-invariant qudit CLH on 1D chain, denoted as H_{1D}. Then the corresponding classical Hamiltonian $H_{1D}^{(c)}$ can be made as 1D finite-range translation-invariant Ising model.*

Combined with the rapid mixing Gibbs sampler for 1D finite-range, translation-invariant Ising model for any constant inverse temperature β [GZ03; Hol85; HS89] which performs classical Gibbs sampling to precision ϵ in time $T(\beta, \epsilon)$, Theorem 49 implies that one can prepare the Gibbs state on (H_{1D}, β) to precision ϵ in quantum time $T(\beta, \epsilon) + O(n)$.

We also give another example of a Hamiltonian on a 2D lattice. Recall our first definition of a 2-local Hamiltonian on 2D (see Section 3.2)

$$H_{2D} = \sum_{\{v,w\} \in E} h^{vw}.$$

As usual, H_{2D} is translation-invariant if all terms h^{vw} are the same. We say that a 2D lattice has periodic boundary condition if it can be embedded onto torus; we will assume a periodic boundary for simplicity.

Lemma 51 *Consider a translation-invariant, 2-local 2-dimensional* qubit *CLH $H_{2D} = \sum_{\{v,w\} \in E} h^{vw}$ with a periodic boundary condition. Then the classical Hamiltonian $H_{2D}^{(c)}$ can viewed as a 2D Ising model under a magnetic field.*

Set the precision to be $1/poly(n)$. If the corresponding 2D Ising model $H_{2D}^{(c)}$ is ferromagnetic with a consistent field, then there exists poly-time mixing Gibbs sampler using the Swendsen-Wang dynamics for any constant temperature (as in [FGW23]). Via our quantum-to-classical Gibbs sampling reduction, we can prepare the Gibbs state for the corresponding CLH H_{2D} in quantum polynomial time.

3.4 Reduction for 2D 4-local qubit CLH without classical qubits

In this section we describe how to prepare the Gibbs state of 2D qubit CLHs. In Section 3.4 we first review the canonical form of the 2D qubit CLH as developed in [AKV18], who establishes a connection between 2D qubit CLHs and the defected Toric code. Based on this connection and an observation on the symmetry of the ground space, we use an oblivious randomized correction technique to generalize the Gibbs state preparation algorithm for the defected Toric code presented in Section 3.1 to prepare the Gibbs state for the more general family of 2D qubit CLHs without classical qubits.[9]

A canonical form

This section is primarily a review of the results in [AKV18]. In that work, the authors prove that 2D qubit CLH[10] without classical qubits (which we will define shortly) is in some sense equivalent to the defected Toric code. [AKV18] used this connection to show that one can prepare the ground state of 2D qubit CLHs similar to the way ground states of the defected Toric code are prepared; this is via the measure and correct approach mentioned in Section 3.1.

We summarize necessary definitions and theorems which will be used in later sections. Recall that a 2D qubit CLH is defined as $H = \sum_{p \in P} p$, where P is the set of plaquettes of the lattice.

Definition 52 (Boundary and interior) *A qubit is in the boundary of the Hamiltonian, if it is acted trivially by at least one of the four adjacent plaquette terms. All other qubits are said to be in the interior. A plaquette term p which acts only on interior qubits is said to be in the interior of the Hamiltonian.*

Definition 53 (Classical qubit) *A qubit is classical if its Hilbert space can be decomposed into a direct sum of 1-dimensional subspace, which are invariant under all terms $\{p\}_{p \in P}$. We say that there is no classical qubit if and onlf if all qubits in the system are not classical.*

In other words a qubit q is classical if under some basis for the qubit, all terms look like $|0\rangle\langle 0|_q \otimes \ldots + |1\rangle\, 1_q \otimes \ldots$.

[9] The formal definition of classical qubits is given in Definition 53.

[10] The definition of a "2D qubit CLH" in [AKV18] is slightly different; qubits are put on the edges of 2D lattice while we put qubits on the vertices. However, the two settings are equivalent, as explained in Appendix C of [IJ23].

Definition 54 (Access to boundary) *We say that a plaquette term p has access to the boundary if there exists a path (a sequence of adjacent vertices) γ_p starting from a vertex of p and ending at a vertex correspoding to a boundary qubit. Morever, there should be some choice of local unitary U_v on each vertex of the path such that the operator*

$$L_p := \otimes_{v\in\gamma_p} U_v X_v U_v^\dagger$$

anti-commutes with p, and commute with all other terms. Note that by construction, $L_p^2 = I$.

Lemma 55 (Interior term) *Suppose there is no classical qubit, and a term p is in the interior of the Hamiltonian. Then by choosing a proper basis for each qubit, we have*

$$p = a_p \mathcal{I} + c_p Z^p, \tag{3.19}$$

with $a_p, c_p \in \mathbb{R}, c_p \neq 0$. Note that replacing p with $p - a_p\mathcal{I}$ does not change the Gibbs state and thus we may assume $a_p = 0$.

Even when all terms are in the interior, Lemma 55 does not imply all the terms should be a tensor product of Pauli Z. To be written in the form of Eq. (3.19), adjacent plaquettes may need different choices of basis. An example is the Toric code, where plaquettes are alternate $X^{\otimes 4}$ and $Z^{\otimes 4}$.

Recall the chessboard partition of the 2D lattice into black plaquettes $\mathcal{B}$ and white plaquettes $\mathcal{W}$.

Lemma 56 (Rephrased from Theorem 5.3 and Lemma 6.2 [AKV18]) *Consider a 2D qubit CLH $H = \sum_{p\in P} p$. Suppose there are no classical qubits. Then,*

(i) If the set of boundary qubits is not empty, then for any adjacent plaquette terms $p \in \mathcal{B}, \hat{p} \in \mathcal{W}$ such that p and $\hat{p}$ are both in the interior, either p or $\hat{p}$ has access to the boundary.

(ii) If there are no boundary qubits, then H is equivalent to the defected Toric code: by a choosing proper basis for each qubit, we have $\forall p \in \mathcal{B}$, p is of the form $a_p\mathcal{I} + c_p X^p$ and $\forall p \in \mathcal{W}$, p is of form $a_p\mathcal{I} + c_p Z^p$.

In [AKV18] the authors use Lemma 56 to reduce the task of preparing the ground state of any (4-local) 2D qubit CLH to the task of preparing the ground state of a 2-local qudit CLH. This reduction is characterized by the following Corollary.

Corollary 57 ([AKV18]) *Consider a 2D qubit CLH $H = \sum_{p\in P} p$. Suppose there are no classical qubits, and the set of boundary qubits is not empty. Then there exists a partition of all the terms $\{p\}_p$ as $\mathcal{P}$ (punctured terms) and $\mathcal{R}$ (terms with access to the boundary) such that,*

(1) After grouping some qubits into qudits, $H_{\mathcal{P}} := \sum_{p\in\mathcal{P}} p$ can be viewed as a 2-local qudit *CLH on a constant-degree planar graph $G = (V, E)$. When viewed as a 2-local Hamiltonian, we also write $H_{\mathcal{P}}$ as $H_G^{(2)} = \sum_{\{v,w\}\in E} h^{vw}$.*

(2) All terms in $\mathcal{R}$ are in the interior of the Hamiltonian and have access to the boundary.

For instance, in Figure 3.2 from the technical overview, $\mathcal{P}$ is the set of all black terms $\mathcal{B}$ and non-removed white terms $\mathcal{O}$. Then, $H_{\mathcal{P}}$ is exactly the 2-local Hamiltonian $H_{DT}^{(2)}$.

Reduction to a classical Hamiltonian

In this section, we describe how to reduce the task of Gibbs state preparation for qubit CLH without classical qubits to the task of classical Gibbs sampling. An explicit and canonical example is the Gibbs state preparation for the defected Toric code, which is described in Section 3.6. For general qubit CLHs without classical qubits, our result is summarized in the following theorem.

Theorem 58 *Given an 2D n-qubit CLH $H = \sum_{p\in P} p$, suppose there are no classical qubits, and the set of boundary qubits is not empty. Let $\mathcal{P}$, $\mathcal{R}$, $H_{\mathcal{P}} := \sum_{p\in\mathcal{P}} p$ and $H_G^{(2)}$ be as defined in Corollary 57.*

Let $H_G^{(2c)}$ be the 2-local classical Hamiltonian derived from $H_G^{(2)}$ as in Section 3.3. Then for any inverse temperature β and precision ϵ, if one can perform classical Gibbs sampling on $(H_G^{(2c)}, \beta)$ to precision ϵ in classical time T, then one can prepare the Gibbs state on (H, β) on a quantum computer in time $T + O(n^2)$.

On the other hand, if the set of boundary qubits is empty, then by Lemma 56 item (ii) the Hamiltonian is equivalent to the defected Toric code and the Gibbs state can be prepared as described in Section 3.1.

We begin with some notation. Recall that the set of all plaquette terms P is partition into black and white terms, i.e. $P = \mathcal{B} \cup \mathcal{W}$. In Corollary 57 we defined $\mathcal{R}$ as the set of terms which have access to the boundary. Let $Q \subseteq \mathcal{B} \cup \mathcal{W}$ be an arbitrary

subset of the plaquette terms. We write $\mathcal{R} \setminus Q$ as the set difference of $\mathcal{R}$ and Q. For $p \in \mathcal{R}$ define the correction operator L_p as in Definition 54.

Define $\boldsymbol{\lambda}_Q := \{\lambda_p\}_{p\in Q}$ to be a set of real values, where each λ_p coresponds to an eigenvalue of $p \in Q$. Let $\lambda(Q) := \sum_{p\in Q} \lambda_p$. Recall that all plaquette terms are commuting. Thus, the terms are simulataneously diagonalizable and the common eigenspace of each $p \in Q$ with eigenvalue λ_p is well defined; we denote this as $\mathcal{H}^Q_{\boldsymbol{\lambda}_Q}$. Formally,

$$\mathcal{H}^Q_{\boldsymbol{\lambda}_Q} := \{|\phi\rangle \mid p\,|\phi\rangle = \lambda_p\,|\phi\rangle\,, \forall p \in Q\}. \tag{3.20}$$

Let $\Pi^Q_{\boldsymbol{\lambda}_Q}$ be the projection onto $\mathcal{H}^Q_{\boldsymbol{\lambda}_Q}$. Again by commutation, $\Pi^Q_{\boldsymbol{\lambda}_Q}$ is equal to the product of the individual projectors:

$$\Pi^Q_{\boldsymbol{\lambda}_Q} = \prod_{p\in Q} \Pi^p_{\lambda_p}\,. \tag{3.21}$$

Note that for a general 2D qubit CLH, a plaquette term p can be any arbitrary 4-qubit operator. For example one can set $H = p_0 + \mathcal{I}$ where p_0 is an arbitrary operator on one plaquette. Nonetheless, one can show that all terms in $\mathcal{R}$ are in a sense quite regular.

Lemma 59 *Each $p \in \mathcal{R}$ has exactly two eigenvalues $\pm c_p$.*

Proof: By definition of $\mathcal{R}$, i.e. Corollary 57 (2), all terms in $\mathcal{R}$ are in the interior of the Hamiltonian. Then Lemma 59 is true by Lemma 55. ∎

Additionally, the eigenspaces corresponding to each eigenvalue of p are symmetric. Lemma 60 is the key observation which leads to the oblivious randomized correction idea.

Lemma 60 *For any subset $Q \subseteq \mathcal{B} \cup \mathcal{W}$ and for any $p \in \mathcal{R}\setminus Q$, we have that*

$$L_p \Pi^p_{+c_p} \Pi^Q_{\boldsymbol{\lambda}_Q} \Pi^p_{+c_p} L_p^\dagger = \Pi^p_{-c_p} \Pi^Q_{\boldsymbol{\lambda}_Q} \Pi^p_{-c_p} \tag{3.22}$$

$$L_p \Pi^p_{-c_p} \Pi^Q_{\boldsymbol{\lambda}_Q} \Pi^p_{-c_p} L_p^\dagger = \Pi^p_{+c_p} \Pi^Q_{\boldsymbol{\lambda}_Q} \Pi^p_{+c_p} \tag{3.23}$$

Proof: We prove the first equality by moving L_p on the very left of the LHS through $\Pi^p_{+c_p}$ and $\Pi^Q_{\boldsymbol{\lambda}_Q}$ until we can cancel it with $L_p^\dagger$. The second equality follows from the first and the fact that $L_p^2 = \mathcal{I}$ and $L_p = L_p^\dagger$. By Lemma 59, we have

$$\Pi^p_{\pm c_p} = \frac{1}{2c_p}\left(\pm p + c_p \mathcal{I}\right). \tag{3.24}$$

By Definition 54, we have L_p anti-commutes with p. Thus we have

$$L_p \Pi^p_{+c_p} = L_p \frac{1}{2c_p} \left(+p + c_p \mathcal{I}\right) \tag{3.25}$$

$$= \frac{1}{2c_p} \left(-p + c_p \mathcal{I}\right) L_p \tag{3.26}$$

$$= \Pi^p_{-c_p} L_p, \tag{3.27}$$

and we can rewrite $L_p \Pi^p_{+c_p} \Pi^Q_{\lambda_Q} \Pi^p_{+c_p} L_p^\dagger$ as $\Pi^p_{-c_p} L_p \Pi^Q_{\lambda_Q} \Pi^p_{+c_p} L_p^\dagger$. Next, by Definition 54 we have that L_p commutes with all terms in Q, and thus L_p also commutes with each eigenspace projectors $\Pi^{p'}_{\lambda_{p'}}$, $\forall p' \in Q$. Recalling the definition of $\Pi^Q_{\lambda_Q}$ in Equation (3.21), this means that L_p commutes with $\Pi^Q_{\lambda_Q}$, and we can move L_p through $\Pi^Q_{\lambda_Q}$. To conclude, we once again use Equation (3.27) and the fact that L_p is unitary (i.e., $L_p L_p^\dagger = \mathcal{I}$), obtaining

$$L_p \Pi^p_{+c_p} \Pi^Q_{\lambda_Q} \Pi^p_{+c_p} L_p^\dagger = \Pi^p_{-c_p} \Pi^Q_{\lambda_Q} \Pi^p_{-c_p}\,, \tag{3.28}$$

as desired. ∎

A consequence of this lemma is that the the the eigenspace corresponding to λ_Q is balanced across any p's $+c_p$ and $-c_p$ eigenspaces.

Lemma 61 $\mathrm{tr}\left(\Pi^p_{+c_p} \Pi^Q_{\lambda_Q} \Pi^p_{+c_p}\right) = \frac{1}{2}\mathrm{tr}\left(\Pi^Q_{\lambda_Q}\right)$.

Proof: Note that

$$\mathrm{tr}\left(\Pi^p_{+c_p} \Pi^Q_{\lambda_Q} \Pi^p_{+c_p} + \Pi^p_{-c_p} \Pi^Q_{\lambda_Q} \Pi^p_{-c_p}\right) = \mathrm{tr}\left((\Pi^p_{+c_p})^2 \Pi^Q_{\lambda_Q} + (\Pi^p_{-c_p})^2 \Pi^Q_{\lambda_Q}\right) \tag{3.29}$$

$$= \mathrm{tr}\left(\Pi^Q_{\lambda_Q}\right), \tag{3.30}$$

where the last equality comes is because Π_{+c_p}, Π_{-c_p} are projections, and $\Pi_{+c_p} + \Pi_{-c_p} = \mathcal{I}$.

Since L_p is a unitary, by Lemma 60 we have that

$$\mathrm{tr}(\Pi^p_{+c_p} \Pi^Q_{\lambda_Q} \Pi^p_{+c_p}) = \Pi^p_{-c_p} \Pi^Q_{\lambda_Q} \Pi^p_{-c_p}\,. \tag{3.31}$$

Thus $\mathrm{tr}(\Pi^p_{+c_p} \Pi^Q_{\lambda_Q} \Pi^p_{+c_p}) = \frac{1}{2}\mathrm{tr}(\Pi^Q_{\lambda_Q})$. ∎

Underyling the Theorem 58 is the following algorithm. We will prove Theorem 58 by proving correctness via Lemma 62. Recall that by Corollary 57 the Hamiltonian $H^{(2)}_G$ is 2-local after we group some qubits into qudits. Using the notation from

Section 3.3, $H_G^{(2)}$ and $H_G^{(2c)}$ denote the 2-local CLH and the corresponding classical Hamiltonian, and $\{\boldsymbol{b}_j\}_{\boldsymbol{b}_j}$ denote the computational basis of the grouped qudits. Since G is planar (Corollary 57 item (1)) the number of edges m is $O(n)$, with n being the number of vertices. As usual, we assume access to a classical Gibbs sampler which obtains a computational basis state $\left|\psi(\boldsymbol{b}_j)\right\rangle$ with probability $p(\boldsymbol{b}_j)$ in time $T + O(m) = T + O(n)$. That is, we prepare a state

$$\rho(\mathcal{P}) := \sum_{\boldsymbol{b}_j} p(\boldsymbol{b}_j) \left|\psi(\boldsymbol{b}_j)\right\rangle \left\langle\psi(\boldsymbol{b}_j)\right| \tag{3.32}$$

$$\text{such that } \left\|\rho(\mathcal{P}) - \rho(H_G^{(2)}, \beta)\right\|_1 \leq \epsilon\,. \tag{3.33}$$

Here we did not write down the explicit formula for $p(\boldsymbol{b}_j)$ since we will not use it. In the second step of the algorithm, we sequentially measure the current state with respect to each removed term $p \in \mathcal{R}$ and perform an oblivious randomized correction. The details are in Algorithm 2.

Algorithm 2 Oblivious Randomized Correction

1: Set the current completed set as $Q = \mathcal{P}$.
2: Set the current state as $\rho(Q) \leftarrow \left|\psi(\boldsymbol{b}_j)\right\rangle$. {Prepare $\rho(\mathcal{P})$.} $p \in \mathcal{R}$
3: Measure the current state $\rho(Q)$ w.r.t measurement p. the outcome is $\lambda_p \in \{\pm c_p\}$
4: w.p. $\frac{\exp(-\beta\lambda_p)}{\exp(-\beta\lambda_p)+\exp(\beta\lambda_p)}$ do nothing and w.p. $\frac{\exp(\beta\lambda_p)}{\exp(-\beta\lambda_p)+\exp(\beta\lambda_p)}$ apply L_p to the measured states.
5: Set $Q \leftarrow Q \cup \{p\}$. Denote the current state as $\rho(Q)$.

We prove correctness by induction. Define $H_Q := \sum_{p\in Q} p$. We claim the following.

Lemma 62 *Assume $\boldsymbol{b}_j$ are sampled from the correct classical Gibbs distribution over $H_G^{(2c)}$. At the end of each **for** iteration in Algorithm 2, the current state $\rho(Q)$ satisfies*

$$\|\rho(Q) - \rho(H_Q, \beta)\|_1 \leq \epsilon\,. \tag{3.34}$$

Proof: When $Q = \mathcal{P}$, note that by definition $H_{\mathcal{P}} = H_G^{(2)}$. Thus Lemma 62 holds by assumption on the initial distribution, as given in Equation (3.33).

Suppose Lemma 62 holds for a set Q. Denote Λ_Q be the set of distinct vectors $\boldsymbol{\lambda}_Q$ where $\Pi_{\lambda_Q}^Q$ is not 0. Note that

$$\rho(H_Q, \beta) = \sum_{\lambda_Q \in \Lambda_Q} \frac{\exp(-\beta\lambda(Q))}{Z(Q)} \cdot \Pi_{\lambda_Q}^Q, \tag{3.35}$$

where $Z(Q)$ is the partition function for H_Q at inverse temperature β,

$$Z(Q) := \sum_{\lambda_Q \in \Lambda_Q} \exp(-\beta\lambda(Q)) \cdot \mathrm{tr}\left(\Pi^Q_{\lambda_Q}\right) . \tag{3.36}$$

Now, consider the next iteration where we measure some $p \in \mathcal{R}\backslash Q$. Let us first assume that in Algorithm 2 line 3, we are measuring the exact Gibbs state

$$\hat{\rho}(Q) := \rho(H_Q, \beta)$$

rather than $\rho(Q)$. We may represent the operation performed during each iteration as a quantum channel $\mathcal{N}_p$. Then, the state at the end of the iteration is

$$\hat{\rho}(Q \cup \{p\}) := \mathcal{N}_p(\hat{\rho}(Q)) \tag{3.37}$$

$$\begin{aligned}
&= \sum_{\lambda_Q} \sum_{\lambda_p \in \{\pm c_p\}} \left[\frac{\exp(-\beta\lambda(Q))}{Z(Q)} \cdot \Pi^p_{\lambda_p} \Pi^Q_{\lambda_Q} \Pi^p_{\lambda_p} \cdot \frac{\exp(-\beta\lambda_p)}{\exp(-\beta\lambda_p) + \exp(\beta\lambda_p)} \right. \\
&\quad \left. + \frac{\exp(-\beta\lambda(Q))}{Z(Q)} \cdot L_p \Pi^p_{\lambda_p} \Pi^Q_{\lambda_Q} \Pi^p_{\lambda_p} L_p^\dagger \cdot \frac{\exp(\beta\lambda_p)}{\exp(-\beta\lambda_p) + \exp(\beta\lambda_p)} \right] \\
&= \sum_{\lambda_Q} \sum_{\lambda_p \in \{\pm c_p\}} \left[\frac{\exp(-\beta\lambda(Q))}{Z(Q)} \cdot \Pi^p_{\lambda_p} \Pi^Q_{\lambda_Q} \Pi^p_{\lambda_p} \cdot \frac{\exp(-\beta\lambda_p)}{\exp(-\beta\lambda_p) + \exp(\beta\lambda_p)} \right. \\
&\quad \left. + \frac{\exp(-\beta\lambda(Q))}{Z(Q)} \cdot \Pi^p_{-\lambda_p} \Pi^Q_{\lambda_Q} \Pi^p_{-\lambda_p} \cdot \frac{\exp(\beta\lambda_p)}{\exp(-\beta\lambda_p) + \exp(\beta\lambda_p)} \right]
\end{aligned} \tag{3.38}$$

$$= \sum_{\lambda_Q} \sum_{\lambda_p \in \{\pm c_p\}} \frac{2\exp(-\beta\lambda(Q))}{Z(Q)} \frac{\exp(-\beta\lambda_p)}{\exp(-\beta\lambda_p) + \exp(\beta\lambda_p)} \cdot \Pi^p_{\lambda_p} \Pi^Q_{\lambda_Q} \Pi^p_{\lambda_p} \tag{3.39}$$

where Equation (3.38) comes from Lemma 60, and Equation (3.39) comes from renaming the $-\lambda_p$ to λ_p in the second half of Equation (3.38).

We next argue that $\hat{\rho}(Q \cup \{p\})$ is equal to $\rho(H_{Q\cup\{p\}}, \beta)$. First, notice that since the terms in $Q \cup \{p\}$ are commuting, we have

$$\Pi^p_{\lambda_p} \Pi^Q_{\lambda_Q} \Pi^p_{\lambda_p} = \Pi^{Q\cup\{p\}}_{\lambda_{Q\cup\{p\}}} . \tag{3.40}$$

Since $\hat{\rho}(Q \cup \{p\})$ is a positive linear combination of positive operators, we have

that $\hat{\rho}(Q \cup \{p\}) \succeq 0$. Moreover, it is correctly normalized:

$$\mathrm{tr}(\hat{\rho}(Q \cup \{p\})) := \sum_{\lambda_Q} \sum_{\lambda_p \in \{\pm c_p\}} \frac{2\exp(-\beta\lambda(Q))}{Z(Q)} \frac{\exp(-\beta\lambda_p)}{\exp(-\beta\lambda_p) + \exp(\beta\lambda_p)} \cdot \mathrm{tr}(\Pi^p_{\lambda_p} \Pi^Q_{\lambda_Q} \Pi^p_{\lambda_p}) \tag{3.41}$$

$$= \sum_{\lambda_Q} \sum_{\lambda_p \in \{\pm c_p\}} \frac{2\exp(-\beta\lambda(Q))}{Z(Q)} \frac{\exp(-\beta\lambda_p)}{\exp(-\beta\lambda_p) + \exp(\beta\lambda_p)} \cdot \frac{1}{2}\mathrm{tr}(\Pi^Q_{\lambda_Q}) \tag{3.42}$$

$$= \sum_{\lambda_Q} \frac{\exp(-\beta\lambda(Q))}{Z(Q)} \mathrm{tr}(\Pi^Q_{\lambda_Q}) \sum_{\lambda_p \in \{\pm c_p\}} \frac{\exp(-\beta\lambda_p)}{\exp(-\beta\lambda_p) + \exp(\beta\lambda_p)} \tag{3.43}$$

$$= \mathrm{tr}(\rho(H_Q, \beta)) \tag{3.44}$$

$$= 1\,, \tag{3.45}$$

where Equation (3.42) comes from Lemma 61. In summary,

- By Equation (3.45), $\hat{\rho}(H \cup \{p\})$ is a quantum state (this can also be inferred from the definition of the algorithm. However, the above calcuations also show this explicitly.)
- Equations Equation (3.39) and Equation (3.40) imply that $\hat{\rho}(H \cup \{p\})$ can be block-diagonalized with respect to the projectors $\Pi^{Q\cup\{p\}}_{\boldsymbol{\lambda}_{Q\cup\{p\}}}$, as should be true for a genuine Gibbs state. Additionally, within a fixed $\boldsymbol{\lambda}_Q$, the term $\exp(-\beta\lambda_p) + \exp(\beta\lambda_p)$ in the denominator of the weights in Equation (3.39) takes the same value for each $\lambda_p \in \{\pm c_p\}$. Thus, the eigenvalues are proportional to $\exp(-\beta\lambda(Q) - \beta\lambda_p)$, which shows that $\hat{\rho}(H \cup \{p\})$ also has the correct weights.

We conclude that

$$\mathcal{N}_p(\hat{\rho}(Q)) = \hat{\rho}(Q \cup \{p\}) = \rho(H_{Q\cup\{p\}}, \beta)\,. \tag{3.46}$$

Of course, in Algorithm 2 we start the **for** loop with the state $\rho(Q)$, which may not be the exact Gibbs state $\hat{\rho}(Q)$. Nonetheless, for the final state $\rho(Q \cup \{p\})$, we derive

$$\|\rho(Q \cup \{p\}) - \rho(H_{Q\cup\{p\}}, \beta)\|_1 = \|\mathcal{N}_p(\rho(Q)) - \mathcal{N}_p(\hat{\rho}(Q))\|_1 \tag{3.47}$$

$$\leq \|\rho(Q) - \hat{\rho}(Q)\|_1 \tag{3.48}$$

$$\leq \epsilon. \tag{3.49}$$

Equation (3.48) comes from the monotonicity of trace distance under quantum channels, and Equation (3.49) comes from the induction hypothesis that Equation (3.34) holds at the beginning of each iteration. ■

Proof:[Proof of Theorem 58.] Theorem 58 is just a corollary of Lemma 62. The runtime of Algorithm 2 is the sum of the time used for preparing the Gibbs state of $H_G^{(2)}$ and the time used to perform the randomized correction for $\mathcal{R}$. Recall that since G is a constant-degree planar graph, the number of edges is $O(m) = O(n)$. Thus the total runtime of Algorithm 2 is

$$T + O(m) + |\mathcal{R}| \times O(n) = T + O(n^2), \tag{3.50}$$

where the $O(n)$ is the cost for the correction operation L_p which is a tensor product state on at most n qubits. $|\mathcal{R}| = O(n)$ is the size of the set $\mathcal{R}$.

■

3.5 Reduction for 2D (4-local) qubit CLH with classical qubit

A review of the techniques of Aharonov et. al [AKV18]

Recall that [AKV18] studied the structure of qubit 2D commuting local Hamiltonians to argue that preparing the ground state can be done in NP. The two main technical results of our work (Theorems 58 and 76) require opening up this result so that we are not only able to prepare a single ground state, but sample from the Gibbs state of the Hamiltonian. This requires a good understanding of their correction operators, as well as the role classical qubits play in their proof. As such, we dedicate this section to reviewing the techniques used in their paper.

C*-algebras and the Structure Lemma

The primary technical tool used in [AKV18] (and in nearly all other works on commuting local Hamiltonians [BV03; Sch11; IJ23]) is the Structure Lemma for C* algebras.

Definition 63 (C*-algebra) *For any Hilbert space $\mathcal{H}$, let $\mathcal{L}(\mathcal{H})$ be the set of all linear operators over $\mathcal{H}$. Then, a* C**-algebra is any complex algebra $\mathcal{A} \subseteq \mathcal{L}(\mathcal{H})$ that is closed under the $\dagger$ operation (playing the role of complex conjugation) and includes the identity.*

Definition 64 (Commuting algebras) *Let $\mathcal{A}$ and $\mathcal{A}'$ be two* C**-algebras on $\mathcal{H}$. We say $\mathcal{A}$ and $\mathcal{A}'$* commute *if $[h, h'] = 0$ for all $h \in \mathcal{A}$ and $h' \in \mathcal{A}'$.*

The connection between local Hamiltonians and algebras is made through the concept of an "induced algebra".

Definition 65 (Induced algebra) *Let h be a Hermitian operator acting on two qudits q_1 and q_2. Consider the decomposition of h into*

$$h = \sum_{i,j} (h_{ij})_{q_1} \otimes (|i\rangle\langle j|)_{q_2} .$$

Here $\{|i\rangle_{q_2}\}_i$ is an orthogonal basis of $\mathcal{H}^{q_2}$, the Hilbert space of q_2. Then the induced algebra of h on q_1 is the C^*-algebra generated by $\{h_{ij}\}_{ij}$ and $\mathcal{I}$, denoted as

$$\mathcal{A}_{q_1}(h) := \left\langle \{(h_{ij})_{q_1}\}_{ij}, \mathcal{I} \right\rangle .$$

If the operator h is clear from context, we will abbreviate $\mathcal{A}_{q_1}(h)$ as $\mathcal{A}_{q_1}$.

The following lemma tells us that the induced algebra is independent of the choice of basis for q_2.

Lemma 66 (Claim B.3 of [AKV18]) *Let h be a Hermitian operator and consider two decompositions of h*

$$h = \sum_{i,j} (h_{ij})_{q_1} \otimes (g_{ij})_{q_2} = \sum_{i,j} (\hat{h}_{ij})_{q_1} \otimes (\hat{g}_{ij})_{q_2} ,$$

where both the sets $\{g_{ij}\}_{ij}$ and $\{\hat{g}_{ij}\}_{ij}$ are linearly independent. Then the C^*-algebra generated by $\{h_{ij}\}_{ij}$ and $\{\hat{h}_{ij}\}_{ij}$ are the same. In particular, the Schmidt decomposition of h is one way to generate an induced algebra.

The induced algebra gives us a tool to analyze whether two terms commute.

Lemma 67 *Let $(h_1)_{q,q_1}$ and $(h_2)_{q,q_2}$ be two Hermitian operators. Then $[h_1, h_2] = 0$ if and only if $\mathcal{A}_q(h_1)$ commutes with $\mathcal{A}_q(h_2)$.*

Finally, we recall the Structure Lemma, which was first applied to understanding commuting local Hamiltonians by [BV03]. Since then, it has been the principal tool used by subsequent works on the CLH [AE13; AE11; Sch11; IJ23]. At a high level, the Structure Lemma says that algebras can be block-diagonalized, and that within each of these blocks, the algebra takes on a tensor product structure which identifies its commutant. For a proof of the lemma, see [Gha+15, Section 7.3].

Lemma 68 (The Structure Lemma) *Let $\mathcal{A} \subseteq \mathcal{L}(\mathcal{H}^q)$ be a C*-algebra on $\mathcal{H}^q$. Then there exists a direct sum decomposition $\mathcal{H}^q = \bigoplus_i \mathcal{H}_i^q$ and a tensor product structure $\mathcal{H}_i^q = \mathcal{H}_{(i,1)}^q \otimes \mathcal{H}_{(i,2)}^q$ such that*

$$\mathcal{A} = \bigoplus_i \mathcal{L}(\mathcal{H}_{(i,1)}^q) \otimes \mathcal{I}(\mathcal{H}_{(i,2)}^q) .$$

We remark that Lemma 68 is equivalent to Lemma 47 in Section 3.3.

A corollary of Lemma 68, and the reason why it is so useful for characterizing the properties of commuting local Hamiltonians, is that in order to commute with an algebra, another algebra must live entirely within the $\mathcal{H}_{(i,2)}^q$ subspaces. Formally, we have the following.

Corollary 69 *Let $(h)_{q,q_1}$ and $(h')_{q,q_2}$ be two Hermitian operators with $[h, h'] = 0$. Let $\mathcal{A}_q(h), \mathcal{A}_q(h') \subseteq \mathcal{L}(\mathcal{H}^q)$ be the induced algebras on $\mathcal{H}^q$. Suppose $\{\mathcal{H}_{(i,j)}^q\}_{i,j}$ is the decomposition induced by Lemma 68 applied to $\mathcal{A}_q(h)$. Then the following holds:*

$$\mathcal{A}_q(h) = \bigoplus_i \mathcal{L}(\mathcal{H}_{(i,1)}^q) \otimes \mathcal{I}(\mathcal{H}_{(i,2)}^q)$$
$$\mathcal{A}_q(h') \subseteq \bigoplus_i \mathcal{I}(\mathcal{H}_{(i,1)}^q) \otimes \mathcal{L}(\mathcal{H}_{(i,2)}^q) .$$

Crucially, all operators keep the decomposition $\mathcal{H}^q = \bigoplus_i \mathcal{H}_i^q$ invariant.

Restrictions on 2D commuting local Hamiltonians

With the Structure Lemma in hand, we can see how [AKV18] apply these tools to show that 2D qubit CLH is in NP.

We start with some definitions. First, we review the notion of a *classical qubit*, defined originally in Definition 53. For a qubit q and a commuting Hamiltonian H, let $\mathcal{N}(q)$ be the set of terms acting non-trivially on q. We say that q is *classical* if there is a non-trivial decomposition of $\mathcal{H}^q$ (i.e. $\mathcal{H}^q = \bigoplus_{i \in \ell} \mathcal{H}_i^q$, with $\ell > 1$) and each term $h \in \mathcal{N}(q)$ keeps this decomposition invariant. We define $C_0(H)$ as the set of classical qubits of H. If the Hamiltonian H is clear from context, we abbreviate $C_0(H)$ as C_0.

Definition 70 (Classical restriction) *Let H be a commuting local Hamiltonian with classical qubits C_0. Then, there exists a unitary $U = \mathcal{I}_{\overline{C_0}} \otimes \bigotimes_{q \in C_0} U_q$ such that*

$\widetilde{H} := UHU^\dagger$ is block diagonal with respect to the computational basis on C_0. A classical assignment to C_0 corresponds to a string $s \in \{0,1\}^{|C_0|}$, with restricted Hamiltonian,

$$H|_s = \Pi_s \widetilde{H} \Pi_s \quad \text{where } \Pi_s := \bigotimes_{q \in C_0} |s_q\rangle\langle s_q|_q \, .$$

Moreover, $H|_s$ is still a commuting Hamiltonian.

Proof: Any classical qubit $q \in C_0$ has a decomposition into $\{\pi_q, \mathcal{I} - \pi_q\}$ such that each term $h \in \mathcal{N}(q)$ commutes with $\pi_q = |\psi\rangle\langle\psi|$ and $\mathcal{I} - \pi_q = |\psi^\perp\rangle\langle\psi^\perp|$, with $\langle\psi|\psi^\perp\rangle = 0$. This implies there exists a unitary transformation U_q on q such that

$$U_q |\psi\rangle = |0\rangle \quad \text{and} \quad U_q \left|\psi^\perp\right\rangle = |1\rangle \, . \tag{3.51}$$

Applying this to each qubit $q \in C_0$ yields the desired unitary $U = \bigotimes_{q \in C_0} U_q$.

To show that $H|_s$ is commuting, we note that each h in the original Hamiltonian is block diagonal with respect to *every* $\{\pi_q, \mathcal{I} - \pi_q\}$, and using Equation (3.51), we see that $UhU^\dagger$ is block diagonal with respect to $|0\rangle\langle 0|$ and $|1\rangle\langle 1|$. Since Π_S is also diagonal in the computational basis, $UhU^\dagger$ commutes with $\Pi_S = \otimes_q |s_q\rangle\langle s_q|$. Thus, commutation of $\Pi_s UhU^\dagger \Pi_s$ and $\Pi_s Uh'U^\dagger \Pi_s$ reduces to the commutation of h and h'. ∎

Remark 2 *Since each term in $H|_s$ now acts as $|s_q\rangle\langle s_q|$ on any classical qudit q, we will treat each term $h \in H|_s$ as having support only on $sup(h) \cap \overline{C_0}$, where $sup(h)$ is the set of qubits on which h acts non-trivially.*

A key point is that an initial restriction $H|_s$, with $s \in \{0,1\}^{C_0}$ can lead to the creation of new classical qubits in $H|_s$. For instance, in Figure 3.5, setting $s_c = |1\rangle$ removes h from the Hamiltonian and causes q to become classical.

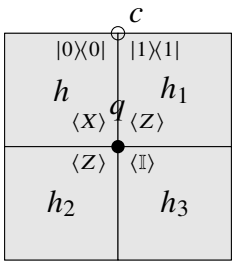


Figure 3.5: c is an initial classical qubit. If c is set to $|1\rangle$, then all terms acting on q have local algebras $\subseteq \langle Z \rangle$, rendering q classical.

Definition 71 (Propagated classical qubits) *For a commuting Hamiltonian H, let C_0 be the set of qubits which are classical with respect to H. Then given an assignment s_0 to C_0, we write $C_1(s_0)$ to denote the classical qubits of the restricted Hamiltonian $H|_{s_0}$. In general, we write $C_i(s_0, \ldots, s_{i-1})$ to refer to the classical qubits of*

$$H|_{(s_0,\ldots,s_{i-1})} := (((H|_{s_0})|_{s_1}) \ldots)|_{s_{i-1}} .$$

Definition 72 (Valid restriction) *A* valid *restriction of H is a sequence $\backslash s = (s_0, \ldots, s_i)$ and Hamiltonian $H|_{\backslash s}$ such that s_i is supported on the classical qubits from the prior restrictions $s_0, \ldots, s_{i-1}$, i.e. $s_i \in \{0,1\}^{C_i(s_0,\ldots,s_{i-1})}$. We say that a valid restriction is* terminating *if $H|_{\backslash s}$ has no classical qubits and $C_{i+1}(s_0, \ldots, s_i) = \emptyset$.*

When we refer to a terminating restriction, that restriction is implictly assumed to be valid.

Definition 73 (Possible Classical Qubits) *For a commuting Hamiltonian H, we write C to be the set of qubit q which are classical with respect to any valid restriction $\backslash s$ and the Hamiltonian $H_{\backslash s}$.*

In other words, C is the set of qubits which might become classical when we sequentially choose an assignment for the current classical qubits. For example in Figure 3.5 q is a "possible classical qubit" and both $c, q \in C$.

Now that we have characterized all the possible classical qubits, we define the notion of a "fully quantum" qubit (and term).

Definition 74 (Fully quantum qubit) *Given a commuting Hamiltonian H, we say that a qubit q is fully quantum if q is not a possible classical qubit (i.e. for any valid restriction $\backslash s$, the qubit q remains non-classical in $H|_{\backslash s}$).*

Definition 75 (Fully quantum term) *We say that a term h of H is fully quantum if all qubits q on which H acts non-trivially are fully quantum.*

With this notation, we can understand the first step of the algorithm of [AKV18] as applying a valid, terminating restriction $\backslash s = (s_0, \ldots, s_i)$ to the Hamiltonian.

Classical qudits

In this section we show a reduction from a CLH instance H to a classical Hamiltonian with constant locality, albeit with some (fully quantum) terms removed, but without

needing to first remove classical qubits. In general, the choice of classical qubits can affect the correction operators for a removed term. To deal with this issue, we make the following assumption.

Assumption 2 (Fully quantum terms are Uniformly Correctable) *Let H be an instance of CLH and suppose h is a fully quantum term (Definition 75). Consider a terminating restriction of the Hamiltonian $H|_s$, such that $H|_s$ has no classical qubits. Suppose that in $H|_s$, h is correctable via the correction operator L_h. Then we assume that h is correctable via the* same *correction operator L_h for any other terminating restriction of the Hamiltonian, $H|_t$, $t \neq s$.*

Formally, our result is stated as follows.

Theorem 76 *Let H be an instance of CLH with some classical qubits. Then, by removing a set of fully quantum terms $\mathcal{R}$, the resulting Hamiltonian can be converted to a $2+O(1)$-local classical Hamiltonian $H^{(c)}$ via a constant depth quantum circuit. Moreover, if*

- *we assume Assumption 2 and,*
- *there is an algorithm to prepare the Gibbs state ρ' of $H^{(c)}$ in time T within precision ϵ,*

then there is a quantum algorithm to prepare the Gibbs state ρ of the original Hamiltonian H in time $T + O(n^2)$ within precision ϵ.

We prove this theorem in two steps. First, in Section 3.5 we show how to modify the proof of [AKV18] so that classical qubits do not need to be removed when preparing a single ground state. Then in Section 3.5 we show how to extend this idea to Gibbs state sampling and get a proof for Theorem 76.

Characterization of classical qubits

Proving Theorem 76 will require characterizing the set of possible classical qubits C (Definition 73). We begin with a lemma about algebra on qubits in the interior (recall Definition 52).

Lemma 77 (Propagation of Classical Qubits) *Let $c \in C_0$ be a classical qubit in the original Hamiltonian H, which is acted on non-trivially by a term A. Suppose*

B is another term which interacts non-trivially with A on two other qubits $q, q' \neq c$. Furthermore, suppose that B is an interior term and is supported entirely outside of C_0. Then any projection π_q and $\pi_{q'}$ respecting the local algebras $\mathcal{A}_q(B)$ and $\mathcal{A}_{q'}(B)$ respectively satisfies

$$\mathcal{A}_r(B) = \mathcal{A}_r(\pi_q \pi_{q'} B \pi_{q'} \pi_q) \quad \textit{and} \quad \mathcal{A}_{r'}(B) = \mathcal{A}_{r'}(\pi_q \pi_{q'} B \pi_{q'} \pi_q)\,,$$

where r, r' are the other qubits in the support of B. In particular, "classical-ness" does not propagate from c through interior terms.

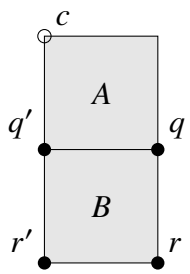


Figure 3.6: Illustration of the terms A and B and relevant qubits.

Proof: First, we claim that the assumption that B is in the interior and supported entirely on non-classical qubits implies that $\mathcal{A}(B) = \langle Z^{\otimes 4} \rangle$, under some change of basis. The proof is essentially by Theorem 5.3 of [AKV18], except that they prove the claim for *each* interior term in the entire Hamiltonian, with the assumption that all classical qubits have been removed. In our case, we only apply their proof for a single term which is not supported on any classical qubits. For completeness, we outline the required components.

Consider the sequence of qubits (q, q', r', r) acted on by B. Since each of these qubits is in the interior, this implies that the two "star" and two "plaquette" terms adjacent to each qubit act non-trivially on it, and thus induce a 2-dimensional algebra (Claim F.2 of [AKV18]). This implies (by Lemma F.5) that $\mathcal{A}_{p,p'}(B) = \langle Z \otimes Z \rangle$ for every length two subsequence of (q, q', r', r) (i.e. every edge of B). Now consider length three subsequences (p, p', p''). Lemma B.4 of [AKV18] tells us that

$$\mathcal{A}_{p,p',p''}(B) \subseteq \mathcal{A}_{p,p'}(B) \otimes \mathcal{A}_{p''}(B) = \langle Z \otimes Z \rangle \otimes \langle Z \rangle.$$

Similarly,

$$\mathcal{A}_{p,p',p''}(B) \subseteq \mathcal{A}_{p}(B) \otimes \mathcal{A}_{p',p''}(B) = \langle Z \rangle \otimes \langle Z \otimes Z \rangle.$$

By writing down the permissible expressions for $h \in \mathcal{A}_{p,p',p''}(B)$ subject to these conditions, we find that $\mathcal{A}_{p,p',p''}(B) = \langle Z \otimes Z \otimes Z \rangle$ (details can be found in [AKV18]). Extending this argument once more to length-4 sequences completes the claim.

In conclusion $\mathcal{A}(B) = \langle Z^{\otimes 4} \rangle$ and thus $B = \alpha \mathcal{I} + \beta Z^{\otimes 4}$ with $\beta \neq 0$. Consider the effect of applying projector π_q and $\pi_{q'}$ on qubits q and q'. Since $\mathcal{A}_q(B) = \mathcal{A}_{q'}(B) = \langle Z \rangle$, the fact that π_q and $\pi_{q'}$ respect the local algebras implies that π_q and $\pi_{q'}$ are in the Z-basis. If both projectors are trivial, the algebra of B on r, r' is unaffected. We give the proof for the case when both are non-trivial and dimension 1; the case when only one is dimension 1 and the other is trivial is similar. The result of applying π_q and $\pi_{q'}$ is

$$\pi_q \pi_{q'} B \pi_{q'} \pi_q = \alpha \pi_q \otimes \pi_{q'} \otimes \mathcal{I} + \beta((-1)^{b_1} \pi_q) \otimes ((-1)^{b_2} \pi_{q'}) \otimes Z^{\otimes 2} ,$$

where $b_1, b_2 \in \{0, 1\}$ indicate the possible phase (corresponding to whether $\pi_q, \pi_{q'}$ correspond to the $+1$ or -1 eigenspaces of Z). We case on the values of (b_1, b_2).

- If $b_1 = b_2$ then we obtain the Schmidt decomposition

  $$B = \pi_q \otimes \pi_q \otimes (\alpha \mathcal{I} + \beta Z^{\otimes 2}) ,$$

  and $\mathcal{A}_{r,r'}(\pi_q \pi_{q'} B \pi_{q'} \pi_q)$ remains $\langle Z^{\otimes 2} \rangle$.

- If $b_1 \neq b_2$ then $\pi_q \otimes \pi_{q'}$ and $(-1)^{b_1} \pi_q \otimes (-1)^{b_2} \pi_{q'}$ are orthogonal, and we have the Schmidt decomposition

  $$B = \pi_q \otimes \pi_q \otimes (\alpha \mathcal{I}) + (-1)^{b_1} \pi_q \otimes (-1)^{b_2} \pi_{q'} \otimes \beta Z^{\otimes 2} .$$

  Again, this leaves the local algebra unchanged.

Therefore, no choice of (consistent) projectors on q, q' change B's local algebra on $\{r, r'\}$. ∎

This lemma provides strong restrictions on the possible set of classical qubits after "propagating" the initial set of classical qubits.

Lemma 78 (Characterization of Classical Qubits) *Let C be the set of all possibly classical qubits (as in Definition 73). Then, each classical qubit $c \in C$ is either*

- *a classical qubit in the original Hamiltonian H, or*

- *supported on a boundary term or a term supported on C_0 (i.e. adjacent to an originally classical qubit).*

Proof: Assume $c \in C$ is not originally a classical qubit. Then, there must have been some set of projections $\pi_{c_1}, \ldots, \pi_{c_k}$ such that in the Hamiltonian $\pi_{c_k} \ldots \pi_{c_1} H \pi_{c_1} \ldots \pi_{c_k}$, the qubit c is a classical. Suppose c is acted on by $N(c) = \{h_1, h_2, h_3, h_4\}$. Certainly, if no projection π_{c_i} is applied to a qubit in the support of any $h \in N(c)$, it does not render c a classical qubit. Otherwise, Lemma 78 implies that if each $h \in N(c)$ is supported outside of C_0 and in the interior of the system, no projection applied to a qudit of h_i renders c classical. Thus, some term $h \in N(c)$ is either a boundary term or supported on C_0. ■

Uniform preparation of a ground state

In this section, we describe how to avoid restricting classical qubits in the proof of [AKV18].

Remark 3 (Comparison to the proof of Aharonov et. al [AKV18]) *We emphasize that this new proof does not qualitatively improve on their result as the initial Hamiltonian is block-diagonal with respect to C and thus for the task of* finding *a ground state, one can always assume the first step is to perform a classical restriction. Moreover, this new proof is* worse *in the sense that [AKV18] obtains a 2-local classical Hamiltonian, whereas, in general, our Hamiltonian can be k-local, for some $k \in O(1)$ depending on the stucture of the Hamiltonian. However, the ideas used here will be useful for the full proof of Theorem 76 in Section 3.5, which allows us to prepare the Gibbs state instead of only a single ground state.*

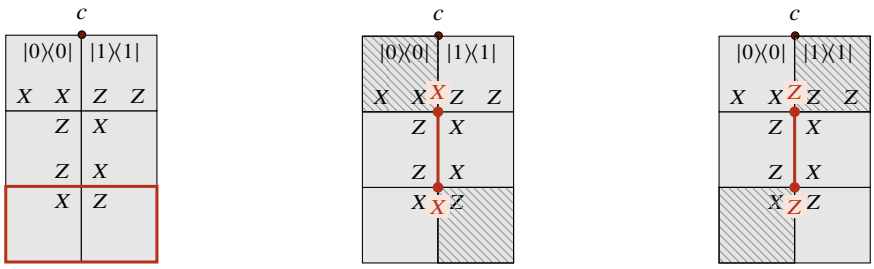
Let us first recall the high level proof of [AKV18], which can be boiled down to:

1. The prover provides a terminating restriction $\backslash s$ such that $H|_{\backslash s}$ contains a ground state of H. This removes all classical qubits.

2. Remove a set of terms $\mathcal{R}$ of $H|_{\backslash s}$ so that the Hamiltonian becomes two local.

3. Prepare a ground state of the two-local Hamiltonian via [BV03] (with the help of the prover).

4. Correct the removed operators in $\mathcal{R}$ to get a ground state for H.



(a) Goal: Remove one of the two terms highlighted in red.

(b) Suppose the classical qubit c is set to $|1\rangle$. This removes the top left term. Then, the correction operator $L_h = X \otimes X$ commutes with all remaining terms.

(c) Otherwise, if c is set to $|0\rangle$, we remove the top right term. Now, the correction operator $L_h = Z \otimes Z$ commutes with all remaining terms.

Figure 3.7: The choice of correction operator may depend on the choice of classical qubits.

Our first observation is that Item 2 can be performed without first removing classical qubits, as this step depends only on the geometric structure of the Hamiltonian. However, the issue comes in Item 4; depending on the choice of classical qubits, the set of "correctable" terms may be different. Therefore, if we perform Item 2 without considering the classical restriction and the resulting set of correctable terms, we may not be able to correct all terms in $\mathcal{R}$.

More precisely, [AKV18] characterize the set of correctable terms via paths to the boundary.

Lemma 79 (Access to the Boundary [AKV18]) *Let H be a CLH instance without any classical qubits, and let h, h' be terms in the interior of the system, such that h and h' share a single edge. Then* either *h or h' has a path to the boundary. If h (or h') has a path to the boundary, we say that h is* correctable.

These "paths to the boundary" yield correction operators, and the choice to remove h or h' depends on which term is correctable. For instance in Figure 3.7, we see that our choice for the classical qubit c changes which of the two boxed terms may be removed.

To avoid this issue, we take Assumption 2, which states that if some h is correctable under some valid, terminating restriction $H|_{\setminus s}$, then it is correctable under *any* other valid, terminating restriction $H|_{\setminus t}$. This implies that the classical restriction can be deferred until after Item 3 is performed. We will now formalize this idea.

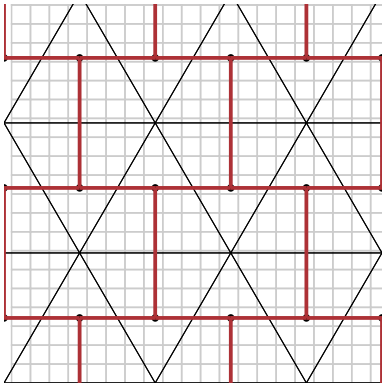


Figure 3.8: An example of a triangulation, followed by a co-triangulation. Within each triangle, a central point is identified. Each point is connected via one of the sides of the triangle to a center of an adjacent triangle. This yields "tiles", demarcated by the dark red lines. The set of qubits within a tile are identified as a single qubit. Within this grouping, the only original Hamiltonian terms that are more than 2-local are those at the centers of the triangles.

Choice of Removed Terms. In the proof of [AKV18], the authors triangulate the 2D complex, then construct a "co-triangulation", dividing the surface in tiles $T \in \mathcal{T}$ such that each qubit within a single tile T becomes a new qudit q_T in the transformed Hamiltonian (see Figure 3.8). Under this transformation, terms in the original Hamiltonian are either 1) internal to a single tile T (and are now 1-local), 2) cross between two adjacent tiles T, T' (and are now 2-local), or 3) on the corner of three tiles (and are now 3-local). Since the goal is to produce a 2-local Hamiltonian, the natural strategy is to simply set the set of removed terms $\mathcal{R}$ to be precisely these corner terms. However, we need to be a bit careful, since any given term is not necessarily correctable; all we know from Lemma 79 is that either the corner term h or its neighbor h' is correctable.[11] This is easy to handle. As long as the original triangulation has sufficient girth, the co-triangulation can be "shifted" so that its corners lie entirely in correctable terms.

In our case, we no longer have the ambiguity of which terms is correctable, but we instead need to deal with classical terms carefully. Consider a triangulation and corresponding tiling $\mathcal{T}$. Let h be an arbitrary term containing a corner of the tiling $\mathcal{T}$. We consider the following set of cases:

- **(Case 1)** h is a *fully quantum* term (Definition 75).

[11] Recall that in [AKV18] classical qubits have already been removed so h and h' are automatically quantum.

  - **(Case 1a)** h is an interior term.
  - **(Case 1b)** h is a boundary term.
- **(Case 2)** h is supported on an originally classical qubit $q \in C_0$.
- **(Case 3)** h is supported on a qubit $q \in C \setminus C_0$ (i.e. a qubit which is classical under some restriction but which is not classical in the original Hamiltonian).

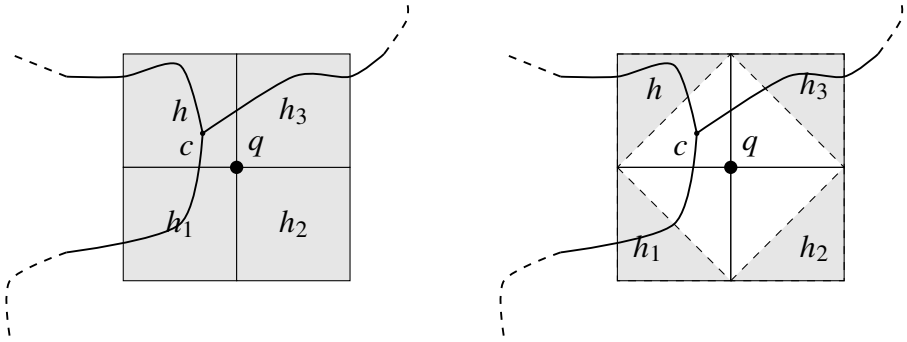


Figure 3.9: Illustration of (Case 2). When a center c is placed on a term supported on a classical qubit, any setting of the classical qubit induces a hole in the 2D structure.

In the first case, we apply the same logic as [AKV18]: for **Case 1a**, h is an interior term, it will be correctable (in our case by Assumption 2) and we added it to the removed set $\mathcal{R}$; in **Case 1b**, h is a boundary term which implies a neighbor acts trivially on one of its qubits. This neighbor results in a "hole" in the original surface. Place the corner of the co-triangulation in the hole.

In **Case 2**, we construct the co-triangulation such that one tile contains only the classical qubits of h; this will ensure that any assignment to the classical qubit renders this term 2-local as well. For **Case 3**, we appeal to Lemma 78; since q is not originally classical but becomes classical, it must be on a boundary term or a term with a classical qubit. In the boundary case, we apply the same argument as in **Case 1b**. In the classical case, apply the argument from **Case 2**. Finally, we denote the Hamiltonian with terms in $\mathcal{R}$ removed as $\widetilde{H}$.

Construction of the Classical Hamiltonian. Now that we have fixed the tiling $\mathcal{T}$ and chosen the set of removed terms $\mathcal{R}$, it remains to describe how to translate this into a classical Hamiltonian. To do so, we introduce some notation. For each tile $T \in \mathcal{T}$, we denote $Q(T)$ to refer to the qubits contained entirely within T and $H(T)$ as the set of original Hamiltonian terms in $\widetilde{H}$ overlapping T (i.e. acting non-trivially

on some $q \in Q(T)$). In addition to the qubits internal to T, the qubits "nearby" T are also important. Define

$$\overline{Q}(T) = \bigcup_{h \in H(T)} Q(T),$$

Additionally, let $\mathcal{N}(T)$ be the tiles neighboring T. Then, we define the Hilbert space of the tile qudit q_T as $\mathcal{H}_T = \otimes_{q \in T} \mathcal{H}_q$. To define the terms in the grouped Hamiltonian, we define the following sets:

- S_T is the set of Hamiltonian terms which act non-trivially on $Q(T) \setminus C_0$ (i.e. on a qudit in T which is not originally classical).
- $S_{T,T'} := S_T \cap S_{T'}$ are the terms acting on both T and T'.
- $h_{T,T'} = \sum_{h \in S_{T,T'}} h$ will be the 2-local terms in the grouped Hamiltonian.
- $h_T = \sum_{\substack{h \in S_T \\ \forall T' \neq T, h \notin S_{T'}}} h$ are the 1-local terms.

Then, the grouped Hamiltonian is denoted as $\widetilde{H}_{\mathcal{T}} = \sum_T h_T + \sum_{T' \neq T} h_{T,T'}$. Again, note that $h_{T,T'}$ is technically *not* 2-local; it only becomes 2-local once a classical restriction to C_0 is fixed. But given such a restriction, we have the following simple consequence of the Structure Lemma [BV03].

Lemma 80 *Any restriction $s \in \{0,1\}^{|C_0|}$ to the classical qubits in C_0 induces a 2-local structure in $H_{\mathcal{T}}$. Therefore, for each q_T, the terms $h_{T,T'}|_s, T' \in \mathcal{N}(T)$ are mutually commuting and induce a decomposition of the Hilbert space of q_T as*

$$\mathcal{H}_T = \bigoplus_{i=1}^{\ell_T} \mathcal{H}_T^{(i)} = \bigoplus_{i=1}^{\ell} \bigotimes_{T' \in \mathcal{N}(T)} \mathcal{H}_T^{(i,T')},$$

where within each subspace $\mathcal{H}_T^{(i)}$, the term $h_{T,T'}|_s$ acts non-trivially only on $\mathcal{H}_T^{(i,T')}$. To make the dependence on an assignment s to C_0 explicit, we refer to this decomposition as $\mathcal{D}_T^s$.

We can say something slightly stronger than this; for a tile T, the above decomposition only depends on terms $h_{T,T'}$ which intersect T (and the internal terms as well). Therefore, if we consider a restriction s' where a bit *outside* of $\overline{Q}(T)$ is flipped, this induces the same decomposition, i.e. $\mathcal{D}_T^{s'} = \mathcal{D}_T^s$. As a result, when specifying

a restriction in the context of a triangle, we implicitly imagine s is defined over $\{0,1\}^{|Q(T)\cap C_0|}$.

Recall our goal is to argue that $h_{T,T'}|_s$ is classical. Recalling Section 3.3, we see that the classical terms constructed for a term $h_{v,w}$ in Equation (3.13) are only a function of the decomposition on vertices v and w (in our case q_T and $q_{T'}$). In particular, for a fixed assignment s and term $h_{T,T'}|_s$, we obtain the classical term,

$$h_{T,T'}|_s := \sum_{\substack{\setminus j=(j_1,j_2)\\ \in[\ell_T]\times[\ell_{T'}]}} \sum_{\setminus b_{\setminus j}^{T,T'}} \lambda(\setminus b_{\setminus j}^{T,T'})|\psi(\setminus b_{\setminus j}^{T,T'})\rangle\langle\psi(\setminus b_{\setminus j}^{T,T'})|, \tag{3.52}$$

where $\setminus b_{\setminus j}^{T,T'}$ and $\lambda(\setminus b_{\setminus j}^{T,T'})$ are defined as in Equations 3.14 to 3.17, except that we use T, T' to refer to qudits, rather than v, w. Then, we have that the following 2-local Hamiltonian is equivalent to $\widetilde{H}|_s$,

$$\widetilde{H}^{(c,s)} := \sum_{T,T'} h_{T,T'}|_s + \sum_T h_T|_s.$$

Now, we have that by construction, any ground state $|\psi\rangle$ to $\widetilde{H}^{(c,s)}$ is (after some 1-local unitary transformations) equal to a ground state $|\phi\rangle$ of $H|_s$; in turn, we can obtain a ground state of the original Hamiltonian $\widetilde{H}$ as $|s\rangle\otimes|\phi\rangle$. In particular, we can write down a Hamiltonian equivalent to $\widetilde{H}$ as

$$\sum_{s\in\{0,1\}^{|C_0|}} |s\rangle\langle s|\otimes\widetilde{H}^{(c,s)} = \sum_{T,T'}\sum_{s\in\{0,1\}^{|C_0|}} |s\rangle\langle s|\otimes h_{T,T'}|_s + \sum_T \sum_{s\in\{0,1\}^{|C_0|}} |s\rangle\langle s|\otimes h_T|_s\,.$$

Unfortunately, since $|C_0|$ can be as large as $\mathsf{poly}(n)$, this yields an exponentially-large Hamiltonian. But we use our earlier observation, which is that the decompositions on T and T only depend on $\overline{Q}(T)\cap C_0$. Since the local decompositions are the same, so too are classical terms obtained in Equation (3.52). This means that for any T, T',

$$\sum_{s\in\{0,1\}^{|C_0|}} |s\rangle\langle s|\otimes h_{T,T'}^{(c,s)} = \sum_{s'\in\{0,1\}^{|C_0\cap(S_T\cup S_{T'})|}} |s'\rangle\langle s'|\otimes h_{T,T'}^{(c,s')},$$

and the resulting Hamiltonian is

$$\widetilde{H}^{(c)} = \sum_{T,T'}\sum_{s'\in\{0,1\}^{|C_T\cup C_{T'}|}} |s'\rangle\langle s'|\otimes h_{T,T'}^{(c,s')} + \sum_T\sum_{s'\in\{0,1\}^{|C_T|}} |s'\rangle\langle s'|\otimes h_T^{(c,s')}. \tag{3.53}$$

Since each tile T is of constant size, each set $C_T\cup C_{T'}$ has constant size as well. This implies we only get a constant blow-up in the number of terms of H, yielding a polynomial-size instance. The locality of the instance depends on the max size of

the set $C_T \cup C_{T'}$. Assuming this is bounded by k, we obtain an $2+k$-local classical Hamiltonian.

Correcting Removed Terms $\mathcal{R}$. As in the original proof, a ground state $|\psi\rangle$ for $\widetilde{H}^{(c)}$ can be constructed in NP. To obtain a ground state for the original Hamiltonian H, we need to correct the terms removed terms $\mathcal{R}$. The first step is to remove the classical qubits. We do this by successively measuring the classical qubits to obtain a terminating restriction $\backslash s = (s_0, \ldots, s_\ell)$.

Lemma 81 *Suppose the above operation yields a series of assignments* $\backslash s = (s_0, \ldots, s_\ell)$*, taking* $|\psi\rangle$ *to* $|\phi\rangle = |s_0, \ldots, s_\ell\rangle\,|\phi'\rangle$*. Then,* $|\phi\rangle$ *is a ground state for the Hamiltonian* $\widetilde{H}|_{\backslash s}$*.*

Proof: We show by induction. By assumption $|\psi\rangle$ is a ground state of $\widetilde{H}^{(c)}$. Suppose this holds for $s_0, \ldots, s_i$, with corresponding ground state $|\psi'\rangle$ of $H' := \widetilde{H}|_{s_0,\ldots,s_i}$. By definition $C' := C_i(s_0, \ldots, s_{i-1})$ are the classical qubits of H'. This implies that (after some single-qubit unitary transformation) each term $h \in H'$ can be written as $h = \sum_{s\in\{0,1\}^{|C'|}} |s\rangle\langle s|_{C'} \otimes \widetilde{h}_s$, and thus H' takes the form

$$\sum_{s\in\{0,1\}^{|C'|}} |s\rangle\langle s|_{C'} \otimes \left(\sum_{h\in H'} \widetilde{h}_s\right) = \sum_{s\in\{0,1\}^{|C'|}} \Pi_s \otimes \left(\sum_{h\in H'} \widetilde{h}_s\right)$$

i.e., H' is *block diagonal* w.r.t. the qubits in C'. Thus, if $|\psi'\rangle$ is a ground state of H', then,

$$0 = \langle\psi'|\, H' \,|\psi'\rangle = \sum_{s\in\{0,1\}^{|C'|}} \langle\psi'|\, \Pi_s H' \Pi_s \,|\psi'\rangle \overset{\Pi_s H'\Pi_s \succeq 0}{\iff} \forall s\, \langle\psi'|\, \Pi_s H' \Pi_s \,|\psi'\rangle = 0.$$

This implies that $\Pi_s\,|\psi'\rangle$ is a ground state of $H'|_s$ for any measurement outcome s. ■

Therefore, the above procedure yields a Hamiltonian $H' = \widetilde{H}|_{\backslash s}$ and a ground state $|\psi\rangle$ of H'. Now, recall that the removed term $h \in \mathcal{R}$ is an interior, fully quantum term of the Hamiltonian H. Assumption 2 implies that there is a correction unitary operator L_h anti-commuting with h and commuting with every other term in H'. Thus, we may perform the measurement $\mathcal{M} = \{\frac{1}{2}(\mathcal{I} + h), \frac{1}{2}(\mathcal{I} - h)\}$ and we can apply L_h to flip the measurement outcome if required.

Gibbs state sampling

In this section, we extend the previous proof so that rather than obtaining a single ground state of H, we obtain a sample from the Gibbs distribution. Initially, we apply exactly the same steps as in the previous section, until we obtain a classical Hamiltonian

$$\widetilde{H}^{(c)} = \sum_{T,T'} \sum_{s' \in \{0,1\}^{|C_T \cup C_{T'}|}} |s'\rangle\langle s'| \otimes h_{T,T'}^{(c,s')} + \sum_{T} \sum_{s' \in \{0,1\}^{|C_T|}} |s'\rangle\langle s'| \otimes h_{T}^{(c,s')} \tag{3.54}$$

equivalent to the original Hamiltonian with terms $\mathcal{R}$ removed. Now, rather than obtaining a *ground state* of $\widetilde{H}^{(c)}$, we assume that we can sample from the Gibbs distribution of $\widetilde{H}^{(c)}$, and then argue that we can recover the Gibbs distribution of the original Hamiltonian, by correcting for the terms $\mathcal{R}$. We will show how to correct these terms inductively. Let Q be the set of terms that were either not removed or already corrected, and let $H^Q = \sum_{h \in Q} h$. Assume we have the Gibbs state $\rho(Q)$ of H^Q

$$\rho(Q) := \frac{1}{Z} \sum_{s, \lambda_s^Q} e^{-\beta \lambda_s^Q} |s\rangle\langle s| \otimes \Pi_{\lambda_s}^Q,$$

where we use the fact that the Hamiltonian H^Q is diagonal w.r.t. $s \in \{0,1\}^{|C_0|}$. To correct terms $h \in \mathcal{R}$, we use the same observation from Lemma 81 that given some classical restriction s, the resulting Hamiltonian $H|_s$ is block-diagonal w.r.t. the qubits in $C_1(s)$. Thus, we may assume we see the state

$$\rho^{(s,t)}(Q) = \sum_{\lambda_{s,t}^Q} e^{-\beta \lambda_{s,t}^Q} |s,t\rangle\langle s,t| \Pi_{\lambda_{s,t}}^Q$$

with probability proportional to $\sum_{\lambda_{s,t}^Q} e^{-\beta \lambda_{s,t}^Q}$. Here, $\Pi_{\lambda_{s,t}}^Q$ is the projector onto the $\lambda_{s,t}^Q$-eigenspace of $H^Q|_{s,t}$. The idea is to apply the proof of Lemma 62 within the subspace $|s,t\rangle\langle s,t| \otimes \mathcal{I}$. Specifically, we make the following observations.

- Lemma 59 holds since each $h \in \mathcal{R}$ is a fully quantum term (for any classical restriction, h is in the interior).
- By Assumption 2 there is a correction operator L_h anti-commuting with h and commuting with all other terms of $H^Q|_{s,t}$, for any choice of s,t. This assumption means that L_h itself is only defined over non-classical qubits and thus for a particular choice of s,t, we can think of this operator as being $|s,t\rangle\langle s,t| \otimes L_h$. Then, Lemma 60 follows by tensoring both sides of Equations (3.22) and (3.23) with $|s,t\rangle\langle s,t|$.

- Similarly, Lemma 61 holds, again by tensoring both matrices with $|s,t\rangle\langle s,t|$.

These observations plus the proof of Lemma 62 implies that we can prepare the state,

$$\hat{\rho}(Q\cup\{h\}) = \frac{1}{Z(Q\cup\{h\})}\sum_{\lambda^Q_{s,t}}\sum_{\lambda_h\in\{\pm c_p\}} \exp(-\beta\lambda^Q_{s,t})\cdot\exp(-\beta\lambda_h)|s,t\rangle\langle s,t|\Pi^{Q\cup\{h\}}_{\lambda^Q_{s,t}+\lambda_h}$$

It's easy to see that *within* the subspace $|s,t\rangle\langle s,t|$ this is the correct Gibbs state. Moreover, across different $(s,t),(s',t')$ pairs, we retain the correct distribution. This is because correcting additional fully quantum terms h only applies an identical multiplicative factor across each of the $|s,t\rangle\langle s,t|$-subspaces.

Acknowledgement

We thank Dominik Hangleiter, Sandy Irani, Anurag Anshu, and Yongtao Zhan for the helpful discussion.

YH is supported by the National Science Foundation Graduate Research Fellowship under Grant No. 2140743; YH is also supported by NSF grant CCF-2430375. Any opinion, findings, and conclusions or recommendations expressed in this material are those of the authors(s) and do not necessarily reflect the views of the National Science Foundation. Jiaqing Jiang is supported by MURI Grant FA9550-18-1-0161 and the IQIM, an NSF Physics Frontiers Center (NSF Grant PHY-1125565).

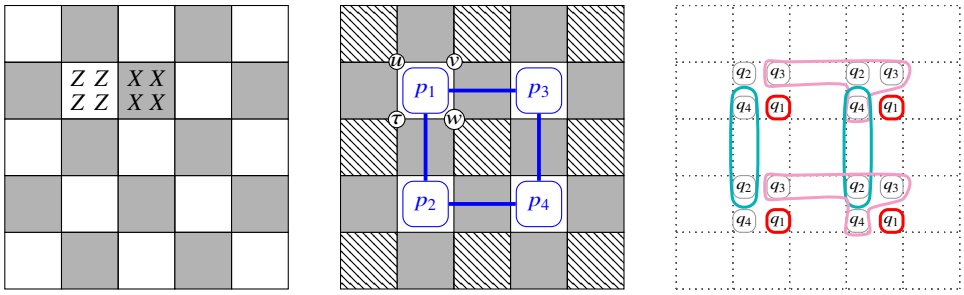
3.6 Appendix: Gibbs sampling reduction for defected Toric code

As described in the proof overview section (Section 3.1), our Gibbs sampler gives an $O(n^2)$-time algorithm for preparing the Gibbs states of the defected Toric code H_{DT}. In this section we instead describe a different and slower Gibbs sampler for H_{DT}, which achieves the following Claim 82 and captures the key ideas for our Gibbs sampler for general qubit 2D CLHs.

We recall the defected Toric code, H_{DT}. As shown in Figure 3.10a, imagine partitioning the plaquettes in the 2D lattice as Black $\mathcal{B}$ and White $\mathcal{W}$. The *defected Toric code* is putting Z and X terms in white and black plaquetes respectively:

$$H_{DT} = \sum_{p\in\mathcal{B}} c_p X^p + \sum_{p\in\mathcal{W}} c_p Z^p, \tag{3.55}$$

where c_p can be an arbitrary real number. In the standard Toric code $c_p = -1$ for any p. One can check that H_{DT} is a qubit 4-local CLH on 2D.



(a) Partition the plaquettes as Black $\mathcal{B}$ and $\mathcal{W}$. Put Z terms on white plaquettes and put X terms on black plaquettes then we get the defected Toric code H_{DT}.

(b) Remove the white terms on the odd lines, H_{DT} will become $H_{DT}^{(2)}$ which can be viewed as a qudit 2-local CLH.

(c) The final classical Hamiltonian. On each white plaquette which has not been removed, we put four qubits 1,2,3,4.

Figure 3.10: Recreation of Figure 3.2. Figure 3.2 is recreated here for convenience, with more detail on the resulting classical Hamiltonian $H_{DT}^{(2c)}$.

Claim 82 *For any inverse temperature β, if one can do classical Gibbs sampling with respect to $H_{DT}^{(2c)}$, β within precision ϵ in classical time T, then one can prepare the quantum Gibbs state with respect to H_{DT}, β within precision ϵ in quantum time $T + O(n^2)$.*

In this section we describe the algorithm and the reduction in details, while as the correctness proof is omitted, as is the same as in Section 3.4. The algorithm will require removing a set $\mathcal{R} \subseteq \mathcal{W}$ of terms from H_{DT}. In particular, if we remove alternating rows of white terms (as in Figure 3.10b) then treat the 4 qubits in the support of each remaining white term as a *single grouped qudit*, then the resulting Hamiltonian $H_{DT}^{(2)}$ is 2-local. We'll denote the remaining white terms as $\mathcal{O} = \mathcal{W} \setminus \mathcal{R}$.

Notation. For simplicity, here we assume the 2D lattice is a square $L \times L$ lattice embedded on a plain which has boundaries. The number of qubits is $n = L \times L$. For better illustration, here we denote the computational basis as $|x\rangle \in \{\pm 1\}^n$ rather than the conventional notation $|x\rangle \in \{0, 1\}^n$, where the Pauli X and Z operators act as

$$Z|1\rangle = |1\rangle, Z|-1\rangle = -|-1\rangle \tag{3.56}$$

$$X|1\rangle = |-1\rangle, X|-1\rangle = |1\rangle. \tag{3.57}$$

This notion is only used in this section. We say $|x\rangle$ is of even Hamming weight if there are even 1s in x.

The classical Hamiltonian $H_{DT}^{(2c)}$. To map $H_{DT}^{(2)}$ to the classical Hamiltonian $H_{DT}^{(2c)}$, we change the basis of the qudit. More specifically, for any term $p \in O$, consider the 4 qubits on p, as shown in Figure 3.10b, name them as u, v, w, τ . A natural basis for the 4 qubit Hilbert space is the computational basis, which are common eigenvalues of $\{Z_u\}_{u\in p}$. An alternative labeling is choosing another set of 4 independent stabilizers

$$S_1^p := Z^p, S_2^p := X_u \otimes X_v, S_3^p := X_v \otimes X_w, S_4^p := X_w \otimes X_\tau, \tag{3.58}$$

The 4 new stabilizers will specify a new basis for the 4-qubit Hilbert space indexed by $s \in \{\pm 1\}^4$, which can be viewed as the computational basis for 4 virtual qubits. The 4 new stabilizers act as Pauli Z on the 4 virtual qubits. That is if we use $(Z_{*i})^p$ to denote Pauli Z on virtual qubit i in plaquette p, then

$$(Z_{*i})^p = S_i^p.$$

Each $s \in \{\pm 1\}^4$ indicates the common eigenvector of S_i^p w.r.t eigenvalues s_i. Denote the corresponding vector as $|s^p\rangle^*$ where $*$ means $|s^p\rangle^*$ is the computational basis for the virtual qubits rather than the original qubits. As an example one can verify that $|\psi^p\rangle$ in Eq. (3.63) corresponds to $|1111^p\rangle^*$. One can check that in this virtual qubit basis, we have that the Hermitian terms[12] can be re-phrase as

$$\forall p \in O, p = c_p \cdot (Z_{*1})^p, \tag{3.59}$$

Besides, note that $X_u \otimes X_\tau = S_2^p \times S_3^p \times S_4^p$, and $X^{\otimes 4} = X^{\otimes 2} \otimes X^{\otimes 2}$, we have

$$\text{For horizontal } p \text{ connecting } p_1, p_3,\ p = c_p \cdot (Z_{*3})^{p_1} \otimes (Z_{*2} \otimes Z_{*3} \otimes Z_{*4})^{p_3}. \tag{3.60}$$

$$\text{For vertical } p \text{ connecting } p_1, p_2,\ p = c_p \cdot (Z_{*4})^{p_1} \otimes (Z_{*2})^{p_2}. \tag{3.61}$$

Thus the resulting Hamiltonian is exactly $H_{DT}^{(2c)}$ as shown in Figure 3.10c and Eq. (3.62), where we omitting the subscript * for simplicity,

$$H_{DT}^{(2c)} := \sum_{\text{horizontal } (p,p_i,p_j)} c_p \cdot (Z_3)^{p_i} \otimes (Z_2Z_3Z_4)^{p_j} + \sum_{\text{vertical } (p,p_i,p_j)} c_p \cdot (Z_4)^{p_i} \otimes (Z_2)^{p_j} + \sum_{p\in O} c_p Z_1^p \tag{3.62}$$

[12] With some abuse of notations, we use p to denote both the plaquette and the Hermitian term on the plaquette.

In summary, by choosing another basis each qudit can be viewed as four virtual qubits, and the 2-local Hamiltonian $H_{DT}^{(2)}$ becomes a classical Hamiltonian $H_{DT}^{(2c)}$ w.r.t those virtual qubits. The computational basis of the virtual qubits $|\boldsymbol{y}\rangle^*$ corresponds to an eigenvector of $H_{DT}^{(2)}$, denoted as $|\phi(\boldsymbol{y})\rangle$. Denote the eigenvalue w.r.t $|\boldsymbol{y}\rangle^*$ and $p \in \mathcal{B} \cup \mathcal{O}$ as $\lambda_p(\boldsymbol{y})$, and define

$$\lambda_{\mathcal{B}\cup\mathcal{O}}(\boldsymbol{y}) := \sum_{p\in\mathcal{B}\cup\mathcal{O}} \lambda_p(\boldsymbol{y}).$$

Preparation of a ground state

Before describing our Gibbs state sampler, we first review a folklore quantum algorithm for preparing the ground state of the standard Toric code.

For ease of presentation, for any subset $Q \subseteq \mathcal{W}$ or $Q \subseteq \mathcal{B}$, we use "terms in Q" to denote the operators $(-X^p)$ if $p \in \mathcal{B}$, and $(-Z^p)$ if $p \in \mathcal{W}$. Note that the ground state of the standard Toric code is the common (-1)-eigenvector of all terms in $\mathcal{W}$ and $\mathcal{B}$. The first step in preparing the ground state of H_{DT} is preparing the ground state of $H_{DT}^{(2)}$, where

$$H_{DT}^{(2)} := \sum_{p\in\mathcal{B}\cup\mathcal{O}} p$$

is two local.

The ground state of $H_{DT}^{(2)}$ is easy to describe. For each term $p \in \mathcal{O}$, denote $|\psi^p\rangle$ as the uniform superposition of basis states with even Hamming weights, that is

$$|\psi^p\rangle = \frac{1}{2} \sum_{x\in\{\pm1\}^p, |x| \text{ even}} |x^p\rangle\,, \tag{3.63}$$

where $\{\pm1\}^p$ is the computational basis of the four qubits of p. One can check that

$$\left|\psi^{\mathcal{B}\cup\mathcal{O}}\right\rangle := \otimes_{p\in\mathcal{O}} |\psi^p\rangle\,, \tag{3.64}$$

is the common (-1)-eigenstate of all terms in $\mathcal{B} \cup \mathcal{O}$.

Remark 4 *The underlying reason that the common (-1)-eigenstates of plaquette terms in $\mathcal{B} \cup \mathcal{O}$ can be chosen as a product of constant-qudit state, lies in the fact that the Hamiltonian $H_{DT}^{(2)}$ can be viewed as a 2-local qudit commuting Hamiltonian after grouping the four qubits in $p \in \mathcal{O}$ as a qudit. Thus by the Structure Lemma, one can prepare the ground state by constant depth quantum circuit.*

Next, we need to correct for the removed terms $p \in \mathcal{R}$. Starting with $\left|\psi^{\mathcal{B}\cup\mathcal{O}}\right\rangle$, we sequentially measure the current state w.r.t the measurement $-Z^p$, for each $p \in \mathcal{R}$.

(a) If we get -1, we know the current state is the common (-1)-eigenstate for all terms in $\mathcal{B} \cup \mathcal{O} \cup \{p\}$ and we move to the next $p \in \mathcal{R}$.

(b) Otherwise we perform a deterministic correction: as shown in Figure 3.2c, we can connect term p to the boundary by a path γ_p. The deterministic correction is done by applying a sequence of X operator, denoted as

$$L_p = \otimes_{v \in \gamma} X_v, \tag{3.65}$$

where X_v is the Pauli X on qubit v. Note that L_p anti-commutes with p, and commutes with all other terms. Thus the final state will again be the common (-1)-eigenstate for all terms in $\mathcal{B} \cup \mathcal{O} \cup \{p\}$.

Remark 5 *For a state $|\psi\rangle$ before the measurement for $p \in \mathcal{R}$ is applied, we end up with a -1-eigenstate whether we fall in the first or second case above. However, note that the exact states we obtain in each case may not be equal. This contributes to the difficulty of obtaining a Gibbs sampler.*

In summary, to prepare the ground state, one first prepares the ground state of a 2-local Hamiltonian $\sum_{p \in \mathcal{B}} -X^p + \sum_{p \in \mathcal{O}} -Z^p$. Then performs a *deterministic correction* to modify the ground state of the 2-local Hamiltonian to the ground state of the Toric code. The ground state preparation for the defected Toric code is similar.

Preparation of the Gibbs state

We now describe how to generalize the ground state preparation algorithm to prepare Gibbs state for the defected Toric code. At a high level, the idea is to:

- Step 1: First prepare the *Gibbs state* of the Hamiltonian

$$H_{DT}^{(2)} := \sum_{p \in \mathcal{B} \cup \mathcal{O}} p.$$

  by mapping it to the *classical 2-local* Hamiltonian $H_{DT}^{(2c)}$ defined in Eq. (3.62).

- Step 2: Perform a *randomized correction*, to modify the current Gibbs state for $H_{DT}^{(2)}$ to the final Gibbs state for H_{DT}.

Step 1: Gibbs state for $H_{DT}^{(2)}$. As in Figure 3.10b, $H_{DT}^{(2)}$ is 2-local in the sense that: if we group every 4 qubits in a plaquette $p \in \mathcal{O}$ as one qudit, then every $p \in \mathcal{O}$ only acts on one qudit, every $p \in \mathcal{B}$ acts on two qudits.

Moreover, since we only change local basis for every qudit, $|\phi(\boldsymbol{y})\rangle$ is in fact a tensor product of single-qudit state. Thus we can prepare the quantum Gibbs state of $H_{DT}^{(2)}$ by a simple algorithm: do classical Gibbs sampling for $H_{DT}^{(2c)}$, get a string $\boldsymbol{y}$, and prepare the tensor product state $|\phi(\boldsymbol{y})\rangle$.

Step 2: Correcting removed terms. Step 2 is more tricky. By Step 1, we assume that we can sample the string $\boldsymbol{y}$ with probability $\exp(-\beta\lambda_{\mathcal{B}\cup\mathcal{O}}(\boldsymbol{y}))/Z_{\mathcal{B}\cup\mathcal{O}}$, where $Z_{\mathcal{B}\cup\mathcal{O}}$ is the normalization factor.

Then we take a term in the removed set $p \in \mathcal{R}$, and try to prepare the Gibbs state w.r.t. $\left(H_{\mathcal{B}\cup\mathcal{O}\cup\{p\}}, \beta\right)$. Consider measuring $|\phi(\boldsymbol{y})\rangle$ w.r.t p. Denote the measurement outcome as $\lambda \in \{\pm c_p\}$. Denote the projectors to be

$$\Pi_{+c_p}^p := \frac{1}{2}\left(I + \frac{p}{c_p}\right), \quad \Pi_{-c_p}^p := \frac{1}{2}\left(I - \frac{p}{c_p}\right). \tag{3.66}$$

We have

$$Pr\ (\text{outcome is } \lambda) = \exp(-\beta\lambda_{\mathcal{B}\cup\mathcal{O}}(\boldsymbol{y}))/Z \cdot \langle\phi(\boldsymbol{y})|\Pi_\lambda^p|\phi(\boldsymbol{y})\rangle. \tag{3.67}$$

Recall that at this moment, the ideal distribution we want is

$$Pr_{ideal}\ (\text{outcome is } \lambda)\ \text{ is proportional to }\ \exp(-\beta\lambda_{\mathcal{B}\cup\mathcal{O}}(\boldsymbol{y})) \cdot \exp(-\beta\lambda). \tag{3.68}$$

Compared to the preparation of *ground state*, which only needs to correct the eigenvalue, in the task of Gibbs state preparation, to get Eq. (3.68) from Eq. (3.67), it seems that one needs to correct

- The probability incurred by measurement, that is $\langle\phi(\boldsymbol{y})|\Pi_\lambda^p|\phi(\boldsymbol{y})\rangle$.
- The probability incurred by the new energy $\exp(-\beta\lambda)$.

This first correction is tricky because it depends on $\phi(\boldsymbol{y})$, thus the probability to be corrected is different for every $\boldsymbol{y}$. We circumvent this problem by an intuitive oblivious *randomized correction*. That is if we get outcome λ then

- With probability $prob := \frac{\exp(-\beta\lambda)}{\exp(\beta\lambda)+\exp(-\beta\lambda)}$ we do nothing.
- With probability $1 - prob$ we apply the correction operation L_p defined in Eq. (3.65).

One may doubt whether the above algorithm successfully prepares the Gibbs state or not, since the correction only brings us back to the right eigenspace, but does not really help us get back to the right state, as noted in Remark 5. That is

$$L_p \Pi^p_{+c_p} |\phi(\boldsymbol{y})\rangle \not\propto \Pi^p_{-c_p} |\phi(\boldsymbol{y})\rangle , \tag{3.69}$$

where here $\not\propto$ means the two vectors are not proportional to each other. In fact, $L_p \Pi_{+c_p} |\phi(\boldsymbol{y})\rangle$ might not even be (proportional to) one of the eigenstates $\{\Pi^p_{+c_p} |\phi(\boldsymbol{y})\rangle\}_{\boldsymbol{y}} \cup \{\Pi^p_{-c_p} |\phi(\boldsymbol{y})\rangle\}_{\boldsymbol{y}}$. The key fact that makes this oblivious randomized correction work, is the symmetry of the eigenspace and the fact that L_p is a unitary. More specifically, let $\Pi^{\mathcal{B}\cup\mathcal{O}}_{\boldsymbol{y}}$ be the common eigenspace of $p' \in \mathcal{B} \cup \mathcal{O}$ w.r.t eigenvalue $\lambda_{p'}(\boldsymbol{y})$. Then

$$L_p \Pi^p_{+c_p} \Pi^{\mathcal{B}\cup\mathcal{O}}_{\boldsymbol{y}} \Pi^p_{+c_p} L_p = \Pi^p_{-c_p} \Pi^{\mathcal{B}\cup\mathcal{O}}_{\boldsymbol{y}} \Pi^p_{-c_p}. \tag{3.70}$$

Finally, note that all $p \in \mathcal{R}$ can be corrected independently. Thus it suffices to perform the same measure and randomized correction procedure sequentially.

For general qubit 2D CLH H without classical qubits, denote the removable terms $\mathcal{R}$ as the set of terms whose measurement value can be corrected without changing the value of other terms. We prepare the Gibbs state of H in a similar way as for the defected Toric code. That is we first remove terms in $\mathcal{R}$ and transform the remaining 2-local Hamiltonian $H^{(2)}$ to a classical Hamiltonian $H^{(2c)}$. Then we first do classical Gibbs sampling for $H^{(2c)}$, then prepare the Gibbs state of $H^{(2c)}$, then perform measurement and randomized correction to prepare the Gibbs state of H.

3.7 Appendix: Proofs of reductions for specific Hamiltonians

Proof:[Proof of Lemma 50]

By the argument in Section 3.3, we can always assume that H_{1D} is 2-local. To make the notation consistent with Section 3.3 we rename H_{1D} as $H^{(2)}_{1D}$. Then according to Section 3.3 we can construct a 1D qudit classical 2-local Hamiltonian $H^{(2c)}_{1D}$.

By assumption, we assumed that the dimension of qudit is $d = 2^k$ for some k, then every qudit can be viewed as k qubits and the computational basis can be written as $\{0,1\}^k$. Note that with an arbitrary ordering of the k qubits, we can view the n qudits on 1D chain as nk qubits on 1D chain. Let Z be the Pauli Z operator, note that

$$|0\rangle\langle 0| = \tfrac{1}{2}(Z + I), \quad |1\rangle\langle 1| = \tfrac{1}{2}(I - Z). \tag{3.71}$$

Thus we can rewrite classical Hamiltonian $H_{1D}^{(2c)}$ as sum of tensors of Z. that is

$$H_{1D}^{(2c)} = \sum_S a_S Z^S, \tag{3.72}$$

where S are subset of qubits, and $a_S = 0$ if $diam(S) \geq 2k$, where $diam$ is the diamater of S, that is the farthest distance between qubits in S w.r.t the 1D chain. Z^S is the tensor product of Pauli Z on qubits in S.

Besides, since H_{1D} is translation-invariant, so does $H_{1D}^{(2c)}$. Combined with the rapid mixing Gibbs sampler for 1D finite-range, translation-invariant Ising model [GZ03; Hol85; HS89], we complete the proof. ∎

The proof of Lemma 51 involves the notations of induced algebra, which should be read after reading section 3.5. **Proof:**[Proof of Lemma 51]

Let h be the translation-invariant term in the 2D Hamiltonian which acts on two systems (qubits) a, b. Let $\mathcal{A}_h^a$ and $\mathcal{A}_h^b$ as the induced algebra of h on systems a and b respectively.

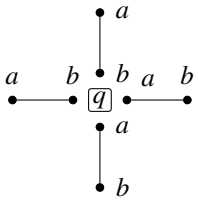


Figure 3.11: Using translational invariance. By translational invariance, each term acting on qubit q is equivalent to $h^{a,b}$.

As pictured in Figure 3.11, consider a qubit q on the 2D lattice, whose Hilbert space is denoted by $\mathcal{H}$. Since by assumption all terms are commuting, we have

$$[\mathcal{A}_h^a, \mathcal{A}_h^b] = 0.$$

By the Structure Lemma[13], that is Lemma 68, the algebra $\mathcal{A}_h^a$ induces a decomposition of $\mathcal{H}$,

$$\mathcal{H} = \bigoplus_{i=1}^{m} \mathcal{H}_i, \text{ where } \mathcal{H}_i = \mathcal{H}_{(i,1)} \otimes \mathcal{H}_{(i,2)}, \text{ and } \mathcal{A}_h^a = \bigoplus_{i=1}^{m} \mathcal{L}(\mathcal{H}_{(i,1)}) \otimes \mathcal{I}(\mathcal{H}_{(i,2)}).$$

Since q is a qubit thus $\dim(\mathcal{H}) = 2$, we know either of the following cases holds:

[13]Note that there are constructive proofs for the structure Lemma, as in Section 7.3 of [Gha+15].

(a) $m = 1$ and $\dim(\mathcal{H}_{(1,1)}) = 1$. Then $\mathcal{A}_h^a$ acts trivially on $\mathcal{H}$. In other words, h is a single-qubit term acts on system b. By definition of local Hamiltonian, h is a Hermitian term. We denote (give a name to) the eigenvalues and eigenstates as λ_0,$|0\rangle$ and λ_1,$|1\rangle$. One can check that

$$H_{2D} = 2 \sum_{q \text{ on 2D}} \lambda_0 |0\rangle\langle 0|_q + \lambda_1 |1\rangle\langle 1|_q.$$

(b) $m = 1$ and $\dim(\mathcal{H}_{(1,1)}) = 2$. By the Structure Lemma we know $\mathcal{A}_h^b$ acts trivially on $\mathcal{H}$. This case can be handled in the same way as (a).

(c) $m = 2$. In this case $\dim(\mathcal{H}_i) = 1$ for $i = 1, 2$. We denote the basis for $\mathcal{H}_1$, $\mathcal{H}_2$ as $|0\rangle, |1\rangle$. Then by the Structure Lemma, $\mathcal{A}_h^a$ keeps $|0\rangle, |1\rangle$ invariant. Since $\mathcal{A}_h^a, \mathcal{A}_h^b$ commute, by Corollary 69 $\mathcal{A}_h^b$ also keeps $|0\rangle, |1\rangle$ invariant. Thus h keeps the computational basis $\{0, 1\}^2$ invariant. In other words, h diagonalize in the computational basis, and thus is a linear combination of terms

$$|00\rangle\langle 00|, |01\rangle\langle 01|, |10\rangle\langle 10|, |11\rangle\langle 11|.$$

Note that

$$|0\rangle\langle 0| = \tfrac{1}{2}(Z + I) \text{ and } |1\rangle\langle 1| = \tfrac{1}{2}(I - Z).$$

We rewrite H as the 2D Ising model with magnetic fields

$$H_{2D} = \sum_{q \text{ on 2D}} \alpha_I I_q + \alpha_Z Z_q + \sum_{q,q' \text{ adjacent}} \beta Z_q \otimes Z_{q'}, \tag{3.73}$$

where we implicitly use the fact that H_{2D} is translation-invariant to get the same coefficient $\alpha_I, \alpha_Z, \beta$ for different q, q'.

■

Chapter 4

LOCAL HAMILTONIAN PROBLEM WITH SUCCINCT GROUND STATE IS **MA-complete**

4.1 Introduction

A fundamental question in quantum chemistry and condensed matter physics is finding the ground energy of a many-body system. A main obstacle to designing efficient classical algorithms for finding ground energy, is the need for exponentially many parameters to completely characterize the ground state. In practice, computational experts often make an additional assumption on the many-body system, that is, the ground state can be well-approximated by a succinct classical description. Here "succinct" refers to that there exists a poly-size classical circuit which computes the amplitude of the ground state on any computational basis. For instance, the Density Matrix Renormalization Group method (DMRG) [Whi92; Whi93; Sch05; LVV15], extensively used in quantum chemistry, operates under the assumption that the ground states can be represented by matrix-product states (MPS) [Vid03]. The Hartree-Fock method [Fis87], on the other hand, assumes that the ground states can be represented as Fermionic Gaussian states. For two-dimensional and higher-dimensional local Hamiltonians, researchers have devised successful heuristic algorithms by representing the ground states by contractible projected entangled pair states [VMC08; Cor16; Van+16]. Additionally, there have been endeavors to model ground states using neural networks [CT17; SSC22; GD17; Car+19].

In this work, we study the complexity-theoretic implications of the succinct ground state assumption, which is used in the above classical algorithms. Recall that the decision version of the ground energy finding problem, often referred to as the local Hamiltonian problem (LHP) [KKR06], is that given an n-qubit k-local Hamiltonian $H = \sum_{i=1}^{m} H_i$, two parameters a, b where $b - a \geq 1/poly(n)$, determine whether the ground energy of H, denoted as $\lambda(H)$, is less than a or greater than b. Here we introduce a variant of LHP, denoted as *LHP with succinct ground state*, where when $\lambda(H) \leq a$, we further assume that there exists a normalized ground state $|\psi\rangle$ which is *succinct*, that is, there exists a poly-size classical circuit $C_\psi(\cdot)$ which computes the amplitude of an unnormalized version of $|\psi\rangle$ as $C_\psi(x) = c \cdot \langle x|\psi\rangle$ for some c. It is well-established that the standard LHP is **QMA-complete** [KKR06]. The central

question (denoted as Q1) we ask is, how the assumption of a succinct ground state affects the complexity of LHP, or more precisely, what complexity class LHP with a succinct ground state is.

It is essential to note that the definition of succinct states is more general than MPS, Fermionic Gaussian states, and similar representations. For instance, the concept of a normalized state $|\phi\rangle$ being succinct does not necessarily imply efficient sample access to $|\phi\rangle$, that is, the ability to efficiently generate sample x with probability $|\langle x|\phi\rangle|^2$. This kind of sample access is a crucial element in many dequantization algorithms [Ten+95; Bra+23b], thus those dequantization algorithms do not apply to our setting. Besides, the Hamiltonian H we consider is general and may not be stoquastic, where stoquastic Hamiltonian [Bra+08] is known to be sign-free and general Hamiltonians suffer from the sign problem [Han+20]. Due to those difficulties, for a general succinct state $|\phi\rangle$, it is highly non-trivial to classically certify the energy, that is, $\langle\phi|H|\phi\rangle$. However, our main result, Theorem 83, demonstrates that we can still efficiently dequantize the quantum verifier in LHP. More connections between Theorem 83, dequantization algorithms and stoquastic Hamiltonians are put into Section 4.1.

Theorem 83 (Main theorem) *LHP with succinct ground state is **MA-complete**.*

In addition to exploring the underlying complexity aspects of classical algorithms, another motivation for LHP with succinct ground state is to gain insight into the boundaries of quantum advantage. Specifically, there is a widespread belief that quantum computers may provide an exponential advantage for quantum chemistry and condensed matter physics [Cao+19; McA+20; Bau+20]. One of the most promising pieces of evidence is the phase estimation algorithm [Til+22], which suggests that if we can prepare a *guided state*, that is, a state which has $1/poly(n)$ overlap with the ground state, then there is an efficient quantum algorithm to estimate the ground energy. The guided states are suggested to be obtained from existing classical algorithms like DMRG or Hartree-Fock, which has a succinct classical description. An important and natural question (denoted as Q2) that arises from this context is, with such classical access to the guided states, can we dequantize the phase estimation algorithm?

Existing literature partially refutes Q2, while many questions remain under research. Specifically, one can define the LHP with guided state [GL22] as a variant of LHP, where when $\lambda(H) \leq a$, we further assume that there exists a guided state. For

the standard definition of the guided states[1], existing works do indicate a potential quantum advantage: LHP with guided state is proved to be **BQP-complete** by Gharibian and Le Gall [GL22], when the guided state is given; and proved to be **QCMA-complete** by Weggemans, Folkertsma and Cade [WFC23], when the guided state is viewed as a witness. When relaxing the precision (the value of $b-a$ in LHP) from inverse-poly to constant, one can efficiently dequantize their algorithms, that is, in **BPP** [GL22] and in **NP** [WFC23] respectively. Conditioned on **BPP** $\subsetneq$ **BQP** and **NP** $\subsetneq$ **QCMA**, the above complexity results suggest that the quantum advantage for LHP is achieving higher precision.

However, these complexity results do not provide a comprehensive answer to Q2, since they overlooked the origin of the guided states. As numerically investigated in [Lee+23], if the guided states are obtained from existing classical algorithms, for Hamiltonians in practice it is possible that those guided states are much better than the standard guided states, which will enable not only an efficient quantum algorithm for the LHP, but also an efficient classical algorithm[2]. Inspired by [Lee+23], from a complexity view we raise the following question (denoted as Q3): Is there a definition of "strong guided states", such that the complexity of LHP with such guided state drops from quantum to classical?

Here quantum refers to **QCMA-complete** which is the complexity of LHP with standard guided states [WFC23], classical refers to **MA-complete**, and LHP with strong guided state is defined as a variant of LHP, where when $\lambda(H) \leq a$, we further assume that there exists a ground state which admits a strong guided state.

Recall that previous results [GL22; WFC23] suggest that the quantum advantage for LHP is achieving higher precision. Our result (Theorem 83) shows that even for inverse-poly precision, if the guided state in the Yes instance is extremely strong, i.e. is the ground state, the complexity of LHP with such guided states does drop to **MA-complete**. This opens the possibility that with a proper definition of strong guided states, even for inverse-poly precision, the LHP with strong guided states is **MA-complete**, which will give an affirmative answer to Q3. This **MA-complete** will imply that heuristic randomized classical algorithms might tackle

[1] We refer the standard guided state to be a state which has $1/poly(n)$ overlap with the ground state, and can be prepared by polynomial-size quantum circuit [GL22; WFC23].

[2] [Lee+23] also argued that with even standard guided states, for chemistry Hamiltonian which is a special case of the general local (Fermionic) hamiltonian, it is possible that the classical heuristics can efficiently estimate the ground energy to the desired precision. Philosophically, [Lee+23] argued that if a classical algorithm is good enough to get a good guided state, which is non-trivially close to a ground state, then the problem itself might be classically tractable.

the corresponding LHP when the strong guided state is given, thus giving a partial answer to Q2. Based on our work, we propose a definition of *strong guided states* to be the succinct approximation of ground state, which has entry-wise $1/poly(n)$ overlap with the ground state:

Definition 84 (Strong guided state) *We say an n-qubit normalized quantum state $|\phi\rangle$ admits a strong guided state, if there is a normalized state $|\eta\rangle$ which is succinct, and satisfies that*

$$\forall x, \langle\eta|x\rangle\langle x|\phi\rangle \geq \langle\phi|x\rangle\langle x|\phi\rangle/poly(n). \tag{4.1}$$

Note that $\langle x|\phi\rangle$ can be a complex number, $\langle\eta|x\rangle\langle x|\phi\rangle \geq \langle\phi|x\rangle\langle x|\phi\rangle/poly(n)$ implicitly implies that $|\eta\rangle$ has information about the "sign" of the ground state. We conjecture the following.

Conjecture 85 *LHP with strong guided state is **MA-complete**.*

Theorem 83 is a special case of Conjecture 85, since a succinct ground state admits a strong guided state as itself. We hope our techniques will be useful for proving Conjecture 85. Conjecture 85 is known to be true if the Hamiltonian in the LHP is stoquastic [Bra14]. The definition of strong guided states here is unarguably strong, which might not be the real reason that makes classical algorithms work [Lee+23]. Here we view the LHP with strong guided state as a starting point of quantitatively understanding the boundary of quantum advantage for LHP — With a standard guided state, there is an efficient quantum algorithm for the LHP. However, when the guided state is too strong, there might also be efficient classical algorithms. What is the precise definition of strong guided states that enable classical algorithms and thus invalidate the potential quantum advantage?

For clarity, we give a further remark on the conceptual relationship among previous work, Theorem 83 and Conjecture 85. Previous work [GL22; WFC23] suggests that quantum advantage for LHP is achieving higher precision. There are two intuitive reasons for such quantum advantage. The direct reason is that the overlap between the guided state and the ground state is $1/poly(n)$ which is relatively small compared to one, and a quantum algorithm which can directly perform quantum operations on guided states seems more powerful than a classical algorithm which can only query the amplitude one by one. A further reason is that general Hamiltonians can have sign problems which makes it hard for classical algorithms to utilize the guided

states. Otherwise, if the Hamiltonian is sign-free (stoquastic), even for inverse-poly precision, previous work [Bra14] already showed that the corresponding LHP with strong guided state could be dequantized, that is, **MA-complete** instead of **QCMA-complete**. Our result (Theorem 83) shows that for general Hamiltonians, the sign problem can be resolved with the help of extremely strong guided states. Our result suggests that the sign problem might also be resolved with the help of certain strong guided states which lie in between standard and extremely strong guided states.

Related works

In this section, we compare our work (Theorem 83) with some related works.

Stoquastic Hamiltonians and quantum Monte Carlo method. Technically, our work is closely related to the quantum Monte Carlo method for stoquastic Hamiltonian and their generalizations, especially the fixed node Monte Carlo method [Ten+95; Bra+23b]. For a Hamiltonian H with a ground state $|\psi\rangle$, the goal of the quantum Monte Carlo method mentioned here is to define a Markov chain which can efficiently sample from the ground state $|\psi\rangle$, that is, outputting x with probability $|\langle x|\psi\rangle|^2$. To ease notations, we denote those quantum Monte Carlo methods as *sampling algorithms*, and denote the runtime required for the Markov chain to be close to the ground state sampling as *mixing time*.

To begin with, Bravyi and Terhal [BT10] first proved that if the Hamiltonian is frustration-free, stoquastic, and has a $1/poly(n)$ spectral gap, one can design a sampling algorithm with $poly(n)$ mixing time. For a general Hamiltonian H, one can transform H to a stoquastic Hamiltonian with some heuristic information by the fixed node Monte Carlo method [Ten+95]. Roughly speaking, given a general Hamiltonian H and an arbitrary known (un-normalized) state $|\phi\rangle$ which we have query access to, [Ten+95] constructs a stoquastic Hamiltonian which is called the fixed node Hamiltonian $F^{H,\phi}$, such that

(1) The ground energy of $F^{H,\phi}$ is always an uppper bound of the ground energy of H, that is,
$$\lambda(F^{H,\phi}) \geq \lambda(H).$$

(2) If $|\phi\rangle$ is the ground state of H, then $|\phi\rangle$ is the ground state of $F^{H,\phi}$ and $\lambda(F^{H,\phi}) = \lambda(H)$.

The drawback of $F^{H,\phi}$ is that its norm can be exponentially large, which might influence the mixing time. Even assuming $|\phi\rangle$ is the ground state of H, there

is no rigorous bound for the mixing time of most sampling algorithms based on $F^{H,\phi}$ [Ten+95; BH18]. A breakthrough is made by Bravyi *et al.* recently [Bra+23b]. Instead of defining a discrete-time Markov chain like most sampling algorithms, from $F^{H,\phi}$ Bravyi *et al.* defined a continuous-time Markov chain. Furthermore, Bravyi *et al.* proved that if $|\phi\rangle$ is the true ground state of H, and the Hamiltonian has a $1/poly(n)$ spectral gap together with some other good conditions, then the continuous-time Markov Chain has $poly(n)$ mixing time, that is, can efficiently sample from the ground state.

Our work builds from the Markov Chain in [Bra+23b] and the properties of the fixed node Monte Carlo method [Ten+95; Bra+23b]. The main difference between our work and [Bra+23b] is that we work on different tasks. [Bra+23b] aimed for sampling from the ground state, with query access to a *trusted* ground state. While we aim for testing ground energy, with query access to an *adversarially* claimed "ground state". The fact that the claimed "ground state" can be adversarial is the main difficulty of solving LHP with succinct ground state. If the witness (the claimed "ground state" $|\phi\rangle$) is trusted to be the true ground state, then the LHP with succinct ground state can be trivially solved by computing the ground energy directly. That is, $\lambda(H) = \frac{\langle x|H|\phi\rangle}{\langle x|\phi\rangle}$ for x with $\langle x|\phi\rangle \neq 0$, which can be computed efficiently since H is sparse and $\langle x|H|\phi\rangle = \sum_y \langle x|H|y\rangle\langle y|\phi\rangle$.

Another remark is that since we work on a different task from [Bra+23b], we do not need the Hamiltonian to have a spectral gap or other additional conditions. Thus even in the Yes instance, one cannot efficiently sample from the ground state. Instead we utilize the promise gap to distinguish the Yes and No instances. Roughly speaking, we define a "Markov chain" that is well-defined for the Yes instances but becomes ill-defined for the No instances. Our verification algorithm distinguishes the Yes and No instances by testing whether the "Markov chain" is well-defined. More can be seen in the proof overview, that is, Section 4.1.

Dequantization algorithms. Conceptually, our work is close to several dequantization algorithms. In particular, Gharibian and Le Gall [GL22] proved when the promise gap is $1/poly(n)$, (promise gap is the value of $b - a$ in LHP), and for the setting where the guided state is given, that LHP with succinct *guided* state is **BQP-complete**. When the promise gap is a constant, one can efficiently dequantize the quantum phase estimation algorithm. Our main theorem—the **MA-complete** in Theorem 83—does not contradict with their result since we are working with succinct *ground* state rather than with succinct *guided* states, and on $1/poly(n)$ promise

gap rather than constant promise gap. Our setting is more similar to [Liu21], where Liu proved that stoquastic LHP with succinct ground state is **MA-complete**. The main difference between our work and Liu's is that we do not assume the Hamiltonian to be stoquastic.

Besides, although aiming for different tasks, one may wonder whether the dequantization techniques initiated by Tang [Tan19; Chi+22; Chi+20] work for our setting. Tang *et al.*'s settings are very different from ours, since they require sample access to the data, while the concept of succinct does not imply sample access. To illustrate this point, consider a 3SAT formula $S(x)$, where $x = x_1x_2...x_n$ are the values of the variables, $S(x) \in \{0, 1\}$ denotes whether x is a satisfying assignment. By definition, the n-qubit quantum state $|\phi\rangle$ which is a uniform superposition of all satisfying assignments is succinct, since the 3SAT formula can compute the amplitude of an unnormalized version of $|\phi\rangle$. However, it is not evident how to efficiently obtain uniform samples of all satisfying assignments from the 3SAT formula.

Verification of matrix products. There is an extensive study of the complexity of matrix verification [Fre79; BŠ06], that is, given query access to three matrices $A, B, C \in \mathbb{R}^{2^n \times 2^n}$, verifying whether $AB = C$. Here we use 2^n as the matrix size for convenience of comparison with our setting.

In particular, [Fre79] gave a classical algorithm with high probability in time proportional to 2^{2n}, and [BŠ06] gave a quantum algorithm in time $O(2^{5n/3})$. Our setting is conceptually related to verifying matrix product where the matrices have different sizes, that is, related to the question where $|\phi\rangle \in \mathbb{R}^{2^n}, H \in \mathbb{R}^{2^n \times 2^n}$, and testing whether the multiplication $\langle\phi|H|\phi\rangle \leq a$ or $\geq b$. In Theorem 83 we claim that we have a **MA**-type algorithm which only needs $poly(n)$-queries instead of $poly(2^n)$. Note that this is achieved because we are testing ground energy rather than testing matrix multiplication for arbitrary $|\phi\rangle, H$. Besides, in our setting the norm of $H \in \mathbb{R}^{2^n \times 2^n}$ is bounded by $poly(n)$. A more detailed generalization where our algorithm works can be seen in Appendix 4.5. This generalization only involves linear algebra and thus might be easier to understand for readers outside quantum information.

Proof overview

To prove Theorem 83, one needs to prove that LHP with succinct ground state is **MA-hard** and is inside **MA**. The **MA-hardness** directly comes from the Section 4 in [Bra+08], which is originally designed to prove that stoquastic Hamiltonian is

MA-hard. We explain why it implies LHP with succinct ground state is **MA-hard** in more detail in Appendix 4.6.

Our main contribution is proving LHP with succinct ground state is inside **MA**. For better illustration, we make some simplifications here. Note that if an n-qubit state $|\phi\rangle$ is succinct, one can efficiently compute the ratio of any two amplitudes, that is, $\frac{\langle x|\phi\rangle}{\langle y|\phi\rangle}$. Since we only use the ratios, to ease notations, here we assume that we can efficiently compute the amplitude $\langle x|\phi\rangle, \forall x$, that is, we have *query access* to $|\phi\rangle$. For simplicity, in this section, we also assume that $a = 0, b = 1/poly(n)$, and assume that H and its ground state are real-valued. We always use the notations $|\psi\rangle$ for the true ground state, and $|\phi\rangle$ for an arbitrary state.

To explain our **MA** verification protocol, we begin with a direct algorithm that fails, then illustrate the ideas to overcome the difficulties. Note that when $a = 0, b = 1/poly(n)$, given H and query access to a succinct state $|\phi\rangle$, the LHP with succinct ground state is equivalent to test

- Whether $H|\phi\rangle = 0$ with $\lambda(H) = 0$,
- Or $\langle\phi|H|\phi\rangle \geq 1/poly(n)$.

Here we assume $|\phi\rangle \neq 0$ which can be checked easily by providing a x where $\langle x|\phi\rangle \neq 0$.

One cannot directly compute $H|\phi\rangle$, since both H and $|\phi\rangle$ are of exponential size. However, note that since H is sparse, one can efficiently check every row of $H|\phi\rangle$, that is,

$$\langle x|H|\phi\rangle = \sum_y \langle x|H|y\rangle\langle y|\phi\rangle. \tag{4.2}$$

Utilizing this, one may immediately come up with the following verification algorithm:

Algorithm 3 A direct algorithm

1: **for** $t = 1$ to $poly(n)$ **do**
2: Randomly sample $x \leftarrow \{0,1\}^n$
3: If $\langle x|H|\phi\rangle \neq 0$, return Reject ▷ Check row x
4: **end for**
5: Return Accept

Unfortunately, Algorithm 3 does not guarantee soundness. It checks the number of zeros in $H|\phi\rangle$ rather than the corresponding energy $\langle\phi|H|\phi\rangle$. Here is a simple counter-example: Let $H = I$, let $|\phi\rangle = |0\rangle^{\otimes n}$. H is a No instance since $\lambda(H) = 1$, while Algorithm 3 accepts with probability close to 1. Although one can easily rule out this case by adding preprocessing, it is easy to construct more complex counter-examples which still make Algorithm 3 fail.

Our key idea to circumvent the above problem, is instead of uniformly sampling the rows to be checked, we use the quantum Monte Carlo method to give different weights to different rows. Specifically, we add checks in the quantum Monte Carlo algorithm which is originally designed for sampling from the ground state. We first explain how [BT10] implicitly[3] used this idea to prove that LHP w.r.t. frustration-free stoquastic Hamiltonian is **MA-complete**, then explain how we adapt this idea to general Hamiltonians. A remark is that the **MA** verification protocol below is only a simplified version of the final protocol. In the final protocol, we need more witnesses and more checks. In particular, for the witness, we not only need a classical circuit for computing the amplitude of the claimed ground state, but also need a claimed ground energy, and a "good" computational basis, which is a warm start of the Markov chain. More details are in Section 4.4.

Let us begin with the **MA** verification protocol when H is stoquastic, where stoquastic means $\langle x|H|y\rangle \leq 0$ for all $x \neq y$. When $\lambda(H) = 0$, one can show that H has a ground state $|\psi\rangle$ whose amplitudes are real and non-negative [BT10]. Given query access to the ground state $|\psi\rangle$, one can connect H to a Markov Chain [BT10], whose transition matrix P is defined as

$$P_{x\to y} = \langle y|P|x\rangle := \langle y|I - \beta H|x\rangle \frac{\langle y|\psi\rangle}{\langle x|\psi\rangle}, \tag{4.3}$$

where $\beta \leq 1/\|H\|$ is to make $\langle y|I - \beta H|x\rangle \geq 0$, thus to make $\langle y|P|x\rangle \geq 0$ since it represents a probability. Furthermore, when $\lambda(H) = 0$, every column of P sums to 1, thus P is a stochastic matrix and a *legal transition matrix*. The key connection between the Hamiltonian H and the Markov chain P, is that the stationary distribution of P is the distribution of sampling the ground state, that is, $x \sim |\langle x|\psi\rangle|^2$. Thus one can sample from the ground state by a random walk w.r.t. P, where the mixing time depends on the spectral gap of P. A modification [BT10] of this sampling algorithm can be used to decide whether $\lambda(H) = 0$ or $\lambda(H) \geq 1/poly(n)$ when H is stoquastic.

[3] [BT10] does not explain their results in the way we describe here. In their setting the succinct classical circuit for the ground state is unknown, their main focus is constructing a circuit that can compute the ratio of the amplitudes, which obscures the idea we described here.

Specifically, consider performing the random walk for time $t = poly(n)$—in the Yes instance, P is a legal transition matrix thus the random walk is always well-defined. In the No instance, for some x, the probability $P_{x\to y}$ is negative or the sum of column x of P is not 1, and thus the random walk is not well-defined. The algorithm rejects if the random walk meets that x. Furthermore, in the No instance, one can show that the accepting probability will be upper bounded by a value proportional to $(1-\beta\lambda(H))^t$, which decays exponentially fast since $\lambda(H) \geq 1/poly(n)$ and $\beta = 1/poly(n)$ in the No instance.

When H is not stoquastic, $\langle y|I-\beta H|x\rangle$ can have both positive and negative entries, and so does the entries in the ground state $|\psi\rangle$. Thus the transition matrix defined in Eq. (4.3) is not well-defined even in the Yes instance. One way to handle this problem, that is, making the transition matrix well-defined, is to use the fixed node Monte Carlo method [Ten+95] introduced in the related work section, which constructs a stoquastic Hamiltonian $F^{H,\phi}$ from a general Hamiltonian and an arbitrary state $|\phi\rangle$, with the property that $\lambda(F^{H,\phi}) \geq \lambda(H)$. Although $F^{H,\phi}$ is stoquastic and thus can be connected to a Markov chain, the main drawback of using $F^{H,\phi}$ directly is that the norm of $F^{H,\phi}$ can be exponentially large. To define a legal probability in Eq. (4.3), that is, $P_{x\to y} \geq 0$, one needs to choose the scaling factor $\beta \leq 1/\|H\|$ to be exponentially small, which might in turn make the accepting probability in the No instance, which is upper bounded by $\sim (1-\beta\lambda(H))^t$, to be close to 1. In other words, the exponentially small scaling factor β in Eq. (4.3) hides any differences between the H in the Yes and No instance, thus making it hard to efficiently distinguish the two cases.

Instead of using a discrete-time Markov chain (DTMC), we build our protocol from the continuous-time Markov Chain (CTMC) by Bravyi *et al.* [Bra+23b], which is based on the fixed node Hamiltonian $F^{H,\phi}$. In case that the readers are not familiar with CTMC, we briefly explain some key concepts here. Recall that a DTMC is described by a transition matrix P, whose entries denote the transition probability. A CTMC is described by a generator matrix G, whose entries denote the transition rate, that is, for small t and $x \neq y$, the probability of jumping from x to y is approximately $\langle y|G|x\rangle \cdot t$. Let the initial distribution be v, and the distribution after evolving DTMC w.r.t. time t and transition matrix P is $P^t v$; The distribution after evolving CTMC w.r.t. time t and generator G is $e^{Gt}v$.

Now we review the key property of the CTMC [Bra+23b], and explain our **MA** verification protocol for the non-stoquastic H. The generator of the CTMC [Bra+23b]

is a rescaled version of $F^{H,\phi}$, that is, the $\widetilde{G}^{H,\phi}$ defined as

$$\langle y|\widetilde{G}^{H,\phi}|x\rangle := \lambda(F^{H,\phi})\delta_{y,x} - \langle y|F^{H,\phi}|x\rangle \frac{\langle y|\phi\rangle}{\langle x|\phi\rangle}. \tag{4.4}$$

Note that $\widetilde{G}^{H,\phi}$ may also have an exponentially large norm. That means, if one simulates the evolution w.r.t. $\widetilde{G}^{H,\phi}$ and time $t = poly(n)$ with a *fixed* step-size, that is, $e^{Gt} = \left(e^{G\delta}\right)^{t/\delta}$, one needs to choose δ to be exponentially small, and thus the algorithm runs in time t/δ and is in-efficient. The key result in [Bra+23b] is that if $|\phi\rangle$ is the true ground state of H, then

(1) $\widetilde{G}^{H,\phi}$ is a *legal generator*.

(2) Furthermore, the CTMC w.r.t. time $t = poly(n)$ can be simulated efficiently by Gillespie's algorithm.

Here Gillespie's algorithm is another standard CTMC simulation method with *varying* step-size. In particular, [Bra+23b] proved that if the Hamiltonian has a spectral gap and some other properties, Gillespie's algorithm converges to the stationary distribution of the CTMC in $poly(n)$ time. A very rough interpretation is that by using the Gillespie's algorithm, simulating this CTMC does not require the exponentially small scaling factor β anymore, which is what we are searching for. Inspired by the verification algorithm for stoquastic Hamiltonian described previously, for general Hamiltonian, our **MA**-verification algorithm is adding consistency checks in the CTMC—in the Yes instance, since $\widetilde{G}^{H,\phi}$ is a legal generator, the CTMC is always well-defined. In the No instance, some parts of the generator are not well-defined. We reject if the random walk meets such parts.

The completeness and soundness of our protocols come from the following observations. For the completeness, a concern one may have is that, compared to [Bra+23b], our setting does not have a large spectral gap, and thus even in the Yes instance the CTMC cannot be close to the ground state sampling. We observe that the rapid mixing property is irrelevant for deciding the LHP. Instead what we need is

(1) $\widetilde{G}^{H,\phi}$ is a legal generator thus the random walk is well-defined.

(2) One can efficiently simulate the CTMC w.r.t. time $t = poly(n)$.

Note that whether a generator is legal is independent of whether the generator has a large spectral gap. We will show that both of the above conditions hold without

a spectral gap [Bra+23b]. The soundness of our protocol mainly comes from the property of the Fixed node Hamiltonians, that is, the ground energy of $F^{H,\phi}$ is always an upper bound of the ground energy of H:

$$\lambda(F^{H,\phi}) \geq \lambda(H),$$

for any adversary state $|\phi\rangle$ [Ten+95]. Thus in the No case, the accepting probability decreases exponentially fast since

$$\lambda(F^{H,\phi}) \geq \lambda(H) \geq 1/poly(n).$$

Conclusion and future work

In this manuscript, we study the underlying complexity question of existing classical algorithms for LHP, which assumes that the ground state has a succinct classical description. More specifically, we define the local Hamiltonian problem with succinct ground state, and prove this problem is **MA-complete**. A remark is that similar to stoquastic, succinct is a basis dependent property. It might be interesting to study the computational complexity of finding the basis which makes the ground state succinct (if such basis exists), where similar questions for stoquastic have been studied in [MLH19; KT19; Ioa+20]. In addition, the **MA-complete** result is established by simulating a continuous-time Markov chain using Gillespie's algorithm. An interesting avenue for future work would be to investigate whether this result could also be demonstrated without this continuous-time technique.

As we illustrate in the introduction, one of the most interesting open questions is how to relax the assumption from classical access of the ground state to classical access of certain guided states, that is, how to give a proper definition of strong guided states where the LHP with strong guided states is **MA-complete**. Based on our work, we give a candidate definition of strong guided states as Definition 84, and give the corresponding Conjecture 85. The intuitive reason that we believe Conjecture 85 is true, is because according to Eq. (4.1), the strong guided state contains two key pieces of information: (a) the strong guided state knows the sign of every amplitude, which makes the problem closer to stoquastic Hamiltonian; (b) the strong guided state roughly knows every amplitude with certain errors. Note that although in this manuscript we choose a basic Markov chain which is sensitive to errors and cannot handle (b), there exist more complicated and robust quantum Monte Carlo methods. In particular, the Projection Monte Carlo algorithm in [Bra14] can handle similar errors as in (b), where they proved if the Hamiltonian is stoquastic, then

LHP with strong guided state is **MA-complete**. The obstacle which prevents us from extending our Theorem 83 to Conjecture 85 by using the Projection Monte Carlo method [Bra14], is the step before using the Markov chain. That is, the fixed node Hamiltonian $F^{H,\phi}$ builds from a strong guided state $|\phi\rangle$ might not keep the promise gap. One way to tackle Conjecture 85 is to develop or identify variants of fixed node Hamiltonians which keep the promise gap when assisted with the strong guided states.

Our primary motivation for defining strong guided states is to dequantize quantum algorithms, based on the conjectures [Lee+23] that the guided states from existing classical algorithms are better than the standard guided states. However, on the other hand, it is also interesting to ask, with strong guided states, can we design better quantum algorithms? Here better means fewer gates, lower depth quantum circuits, or much easier to implement in near-term quantum devices.

Structure of the chapter

The structure of this chapter is as follows. In Section 4.2 we give notations and definitions which are used throughout this manuscript. In Section 4.3 we introduce the techniques used in our verification protocol. In Section 4.3 we briefly explain the argument from [Bra+23b] that w.l.o.g. we can assume that the Hamiltonian and states are real-valued. In Section 4.3 we review the continuous-time Markov chain and Gillespie's algorithm. In Section 4.3 and Section 4.3 we list the construction and key properties of the fixed node Monte Carlo method [Ten+95], and the variant of using continuous-time Markov chain [Bra+23b].

Finally, in Section 4.4 we give our **MA** verification protocol and prove Theorem 83. In particular, in Section 4.4 we assume that we have access to continuous-time randomness, that is, we assume that we can sample u from uniform distribution of $[0, 1]$ in $poly(n)$ time. In Section 4.4 we substitute this assumption by discrete randomness and prove that the error induced by the discretization is small.

4.2 Notations and Definitions

Notations. We use $\mathbb{R}, \mathbb{C}$ to represent the real field and complex field. Let $S \subseteq \{0,1\}^n$, we use $|\phi\rangle \in \mathbb{R}^{|S|}$ to denote a vector, $H, M \in \mathbb{R}^{|S|\times|S|}$ to denote matrices, and $x, y \in S$ to denote bit strings. The entries of $|\phi\rangle$ are indexed by $x \in S$, denoted by $\langle x|\phi\rangle$, similarly for H, M.

We use $\| |\phi\rangle \|$ to denote the vector L2 norm. We say $|\phi\rangle$ is normalized if $\| |\phi\rangle \| = 1$.

We say an un-normalized state $|\phi\rangle$ is regularized if $\langle x|\phi\rangle \neq 0$ for all $x \in S$. We use $\|M\|$ to denote its spectral norm, that is,

$$\|M\| := \max_{|\phi\rangle \in \mathbb{R}^{|S|}, |\phi\rangle \neq 0} \frac{\|M|\phi\rangle\|}{\||\phi\rangle\|}.$$

We use $amax(M)$ to denote $\max_{x,y \in S} |\langle x|M|y\rangle|$. For a Hermitian H, we use $\lambda(H)$ to denote its ground energy, that is, the minimum eigenvalue. We use $\lambda_{max}(H)$ to denote its maximum eigenvalue. For $d = poly(n)$, we say H is d-sparse if

(1) Each row and each column of H only have d non-zero entries.

(2) Given a row index $x \in S$ (or column index $y \in S$), there is a poly(n)-time algorithm which can list all the non-zero entries in row x (or column y), that is, list all $z \in S$ such that $\langle x|H|z\rangle \neq 0$, and similarly for column y.

We say a Hermitian matrix H or a vector $|\phi\rangle$ is real-valued if $\langle x|H|y\rangle \in \mathbb{R}, \forall x, y \in S$, or $\langle x|\phi\rangle \in \mathbb{R}, \forall x \in S$.

A matrix M is stochastic if

(1) $\langle y|M|x\rangle \geq 0, \forall x$ and y.

(2) $\sum_y \langle y|M|x\rangle = 1, \forall x$.

A Hermitian matrix M is stoquastic if

$$\langle x|M|y\rangle \leq 0, \forall x \neq y, x \in S, y \in S.$$

Given a vector $|\phi\rangle \in \mathbb{C}^{|S|}$, we use $supp(\phi)$ to denote the positions of non-zero entries, that is,

$$supp(\phi) := \{x \in S | \langle x|\phi\rangle \neq 0\}.$$

We use $\left|\phi_{supp}\right\rangle \in \mathbb{C}^{|supp(\phi)|}$ to denote the vector obtained by deleting 0 values in $|\phi\rangle$. For a vector $|\phi\rangle \in \mathbb{R}^{|S|}$, we use $Diag(\phi) \in \mathbb{R}^{|S| \times |S|}$ to denote the diagonal matrix where

$$\langle x|Diag(\phi)|x\rangle := \langle x|\phi\rangle, \text{ for } x \in S.$$

When $S = \{0,1\}^n$, we call $H, M \in \mathbb{C}^{2^n \times 2^n}$ n-qubit operators, and $|\psi\rangle, |\phi\rangle \in \mathbb{C}^{2^n}$ n-qubit vectors. When considering an n-qubit Hermitian H, we will always use $|\psi\rangle \in \mathbb{C}^{2^n}$ to denote the ground state, and $|\phi\rangle \in \mathbb{C}^{2^n}$ to denote an arbitrary n-qubit

state, which may or may not be the ground state. Note that in this manuscript $|\psi\rangle$ and $|\phi\rangle$ may be **unnormalized**.

An n-qubit k-local Hamiltonian is an n-qubit Hermitian operator $H \in \mathbb{C}^{2^n \times 2^n}$, where $H = \sum_{j=1}^{m} H_j$, and H_j acts nontrivially on at most k qubits. We always assume that k is a constant, $m = poly(n)$, and $\|H\| \leq poly(n)$.

For any real number α, we use $\lfloor \alpha \rfloor$ for the largest integer which is smaller than α, and $\lceil \alpha \rceil$ for the smallest integer which is greater than α. We use ln (or log) to represent the logarithm w.r.t. the natural exponent e (or 2).

Let w be a random variable. We use w.p. as an abbreviation for with probability. We say w is sampled from the uniform distribution $[0, 1]$, if the probability density function is $p(w) = 1, \forall w \in [0, 1]$, and $p(w) = 0$ for $w < 0$ and $w > 1$. We say w is sampled from the exponential distribution of parameter $\lambda > 0$, if the probability density function is $p(w) = \lambda e^{-\lambda w}$ for $w \geq 0$, and $p(x) = 0$ for $w < 0$.

We also consider the truncated discretized version of the exponential distribution.

Definition 86 *Let K be an integer, $\delta \in \mathbb{R}$ be a small value, and $\lambda > 0$. We define the truncated discretized exponential distribution $\mathcal{D}_{K,\delta,\lambda}$ as a distribution over $\{k\delta\}_{k=0,1,..,K}$. In particular, we say w is sampled from $\mathcal{D}_{K,\delta,\lambda}$ if the probability is*

$$Pr_{w \sim \mathcal{D}_{K,\delta,\lambda}}(w = k\delta) =$$
$$\begin{cases} \exp(-\lambda k\delta) - \exp(-\lambda(k+1)\delta), \text{ if } 0 \leq k \leq K-1 \\ \exp(-\lambda k\delta), \text{ if } k = K \\ 0 \text{ if } k < 0 \text{ or } k > K. \end{cases}$$

Note that

$$\forall k \in \{0, ..., K\}, Pr_{w \sim \mathcal{D}_{K,\delta,\lambda}}(w \geq k\delta) = \exp(-\lambda k\delta).$$

Definition 87 (Succinct quantum state) *We say an n-qubit normalized state $|\phi\rangle$ is succinct if there exists a $poly(n)$-size classical circuit C_ϕ which can compute the amplitude of $|\phi\rangle$ up to a common factor, that is, $C_\phi(x) = c_\phi(n) \cdot \langle x|\phi\rangle$, where $0 < c_\phi(n) \leq 2^{poly(n)}$ is a function independent of x.*

Note that if $|\phi\rangle$ is succinct then one can compute the ratio of amplitudes of $|\phi\rangle$, that is, $\frac{\langle x|\phi\rangle}{\langle y|\phi\rangle} = \frac{C_\phi(x)}{C_\phi(y)}$. The requirement of $c_\phi(n) \leq 2^{poly(n)}$ comes from the fact that $C_\phi(x)$ should be efficiently described by $poly(n)$-bits.

Definition 88 (LHP with succinct ground state) *Given (H, a, b) where $H = \sum_{i=1}^{m} H_i$ is an n-qubit k-local Hamiltonian, $\|H\| = poly(n)$, $m = poly(n)$, k is a constant; a, b are two parameters and $b - a \geq 1/poly(n)$. Besides, it is promised that one of the following holds:*

- *Yes instance: $\lambda(H) \leq a$, and there exists a ground state $|\psi\rangle$ which is succinct.*
- *No instance: $\lambda(H) \geq b$.*

The local Hamiltonian problem with succinct ground state is deciding whether (H, a, b) is the Yes instance or the No instance.

We implicitly assume that there is a sufficiently large polynomial $p(n) = poly(n)$, such that every value in Definition 88, that is, $\langle x|H|y\rangle, a, b, m, C_\psi(x)$ can be represented by $p(n)$-bits. More clarification on precision can be found in Appendix 4.11 Remark 6. To ease analysis, we also assume that we can use $poly(n)$-time to sample the discretized exponential distribution $\mathcal{D}_{K,\delta,\lambda}$ exactly for parameters specified in Appendix 4.11 Remark 7.

Recall our main theorem is as follows.

Theorem 89 *Under the precision assumptions (Remark 6 and Remark 7 in Appendix 4.11), LHP with succinct ground state is **MA**-complete.*

We will prove Theorem 89 in the following sections. A remark is that a slightly modified proof can generalize Theorem 89 from local Hamiltonians to sparse Hamiltonians whose spectral norm is bounded by $poly(n)$.

4.3 Preliminaries

We first give two facts on the exponential function and the spectral norm, which will be used repetitively. The proof can be found in Appendix 4.7.

Fact 90 *For any $x \in [-1, 1]$, $|e^{-x} - (1 - x)| \leq 2x^2$.*

Fact 91 *Let $S' \subseteq S \subseteq \{0,1\}^n$ be two non-empty sets. Let $M \in \mathbb{R}^{|S|\times|S|}$ be a Hermitian matrix. Let $N \in \mathbb{R}^{|S'|\times|S'|}$ be the submatrix of M by restricting rows and columns in S'. Then*

$$\lambda(M) \leq \lambda(N) \leq \lambda_{max}(N) \leq \lambda_{max}(M).$$

In particular,

$$\|N\| \leq \|M\|.$$

Reduction to real Hamiltonians

In Definition 88 the n-qubit k-local Hamiltonian H and n-qubit state $|\psi\rangle$ can have complex values. This section is to explain w.l.o.g. we can assume that H and $|\psi\rangle$ are real-valued by adding one ancilla qubit. The following argument is simplified from [Bra+23b].

Let $|\phi\rangle$ be any eigenstate of H of eigenvalue α. Write $H = H_R + iH_I$ where H_R, H_I are real-valued and k-local, write $|\phi\rangle = |\phi_R\rangle + i\,|\phi_I\rangle$ where $|\phi_R\rangle, |\phi_I\rangle$ are real-valued. Define an $(n+1)$-qubit Hamiltonian H' and $(n+1)$-qubit states $\left|\phi'^0\right\rangle, \left|\phi'^1\right\rangle$ as

$$\begin{aligned} H' &= H_R \otimes I + H_I \otimes \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix}, \\ \left|\phi'^0\right\rangle &= |\phi_R\rangle\,|0\rangle + |\phi_I\rangle\,|1\rangle\,, \\ \left|\phi'^1\right\rangle &= -\,|\phi_I\rangle\,|0\rangle + |\phi_R\rangle\,|1\rangle\,. \end{aligned}$$

One can verify that $\left|\phi'^0\right\rangle, \left|\phi'^1\right\rangle$ are orthogonal and are of eigenvalue α of H'. Besides, for any orthogonal n-qubit states $|\phi\rangle$ and $|\eta\rangle$, the set $\{\left|\phi'^0\right\rangle, \left|\phi'^1\right\rangle\}$ is orthogonal to $\{\left|\eta'^0\right\rangle, \left|\eta'^1\right\rangle\}$. Let $|\phi\rangle$ ranges over all 2^n eigenvectors of H, one will get a complete set of $2 \cdot 2^n$ eigenvectors of H'. Thus H and H' have the same spectrum. In particular H and H' have the same ground energy.

Given H, let $|\psi\rangle$ be its ground state, define the real-valued H' and $\left|\psi'^0\right\rangle$ as above. Then H' is $(n+1)$-qubit, $(k+1)$-local. Further since H is Hermitian, thus H_R is symmetric and H_I is anti-symmetric. Thus H' is symmetric and thus Hermitian. Note that one can efficiently calculate $\langle x'|H'|y'\rangle$, $\langle x'|\psi'^0\rangle$ with access to $\langle x|H|y\rangle$, $\langle x|\psi\rangle$, where x, y are the first n-bits of x', y'. Thus w.l.o.g. we assume that $H, |\psi\rangle$ are real-valued.

Review of the CTMC and Gillespie's algorithm

This section is a brief review of the continuous-time Markov chain and Gillespie's algorithm. More can be seen in Section 6 of [And] and other textbooks on Markov chain and random process [Nor98]. To ease notations, in the following we abbreviate the discrete state space, discrete-time Markov Chain as DTMC; and abbreviate the discrete state space, continuous-time Markov Chain as CTMC. In this section, we

use $S \subseteq \{0,1\}^n$ to denote the discrete state space, $P, G \in \mathbb{R}^{|S|\times|S|}$ to denote matrices, and $x, y \in S$ to denote the states. For simplicity, we abbreviate $\sum_{x\in S}$ as $\sum_x$.

To introduce CTMC, we first review the concepts related to DTMC. A DTMC with discrete state space S is a sequence of random variables $\{\xi(j)\}_{j=0,1\ldots}$, where $\xi(j) \in S$ is the state at discrete time j. A DTMC is associated with a matrix $P \in \mathbb{R}^{|S|\times|S|}$, which is called the transition matrix, where $\langle y|P|x\rangle$ denotes the *transition probability* from state x to state y. P is a *legal* transition matrix iff P is a stochastic matrix, that is,

$$\langle y|P|x\rangle \geq 0, \forall x, y; \text{ and } \sum_y \langle y|P|x\rangle = 1, \forall x. \tag{4.5}$$

The DTMC w.r.t. P is defined as the discrete-time stochastic process such that at any time j where $\xi(j) = x$, one chooses the next state $\xi(j+1)$ to be y w.p. $\langle y|P|x\rangle$.

A CTMC with discrete state space S is a set of random variables $\{\xi(t)\}_t$, where $\xi(t)$ is the state at time t, and t can take values from a continuous interval, e.g. $t \in [0,1]$[4]. A CTMC is associated with a matrix $G \in \mathbb{R}^{|S|\times|S|}$, which is called the *generator*, where $\langle y|G|x\rangle$ is called the *transition rate* from state x to state y. G is a *legal* generator iff

$$\langle y|G|x\rangle \geq 0 \text{ for } y \neq x; \text{ and } \sum_y \langle y|G|x\rangle = 0.$$

Note that when G is a legal generator, one can verify that $\exp(Gt)$ is a stochastic matrix for any $t \geq 0$. A remark is that, to make the notations consistent with previous work [Bra+23b], stochastic matrix in this manuscript (defined as in Eq. (4.5)) has a column sum of 1 instead of a row sum of 1. Also note that for a legal generator the diagonal elements are non-positive, since

$$\langle x|G|x\rangle = -\sum_{y\neq x} \langle y|G|x\rangle \leq 0.$$

We say a distribution π over the state space S is a stationary distribution of the CTMC w.r.t. G if

$$\forall y, \sum_x \langle y|G|x\rangle \pi(x) = 0,$$

which will imply

$$\sum_x \langle y|\exp(Gt)|x\rangle \pi(x) = \pi(y), \forall y.$$

[4]The main difference between CTMC and DTMC is, in DTMC, the index of time is discrete, while in CTMC, the index of time is continuous.

There are multiple ways to define a CTMC, here we define a CTMC via Gillespie's algorithm as follows: Given a legal generator $G \in \mathbb{R}^{|S|\times|S|}$, one can first define a DTMC named the embedded chain, where the transition matrix $P \in \mathbb{R}^{|S|\times|S|}$ is defined as $\forall x, y \in S$,

$$\langle y|P|x\rangle := \begin{cases} 0, \text{ if } x = y. \\ \dfrac{\langle y|G|x\rangle}{|\langle x|G|x\rangle|}, \text{ if } x \neq y. \end{cases} \tag{4.6}$$

The CTMC w.r.t. G can be defined via "Embedded chain + Waiting time". Roughly speaking, the CTMC w.r.t. G is defined as the continuous-time stochastic process that when ξ visits the state x at time τ, it will stay in the state x for a "waiting time" $\Delta\tau$, where $\Delta\tau$ is sampled from the exponential distribution with parameter $|\langle x|G|x\rangle|$. Until time $\tau + \Delta\tau$, it will move to the next state y according to the embedded chain, that is, w.p. $\langle y|P|x\rangle$. More precisely, here we define the CTMC w.r.t. a legal generator G by the Gillespie's algorithm, as shown in Algorithm 4.

Algorithm 4 Gillespie's algorithm(G, x_{in}, t)

$x \leftarrow x_{in}$,$\tau \leftarrow 0$, $\xi(0) \leftarrow x_{in}$
while $\tau < t$ **do**
  if $|\langle x|G|x\rangle| = 0$ **then**
    Set $\xi(s) = x$ for all $s \in (\tau, t]$
    $\tau \leftarrow t$
  else
    Sample $u \in [0, 1]$ from the uniform distribution $[0, 1]$
    $\Delta\tau \leftarrow \frac{\ln(1/u)}{|\langle x|G|x\rangle|}$
    Set $\xi(s) = x$ for all $s \in (\tau, \tau + \Delta\tau]$
    $\tau \leftarrow \tau + \Delta\tau$
    Sample $y \in S \setminus \{x\}$ from the probability distribution $\frac{\langle y|G|x\rangle}{|\langle x|G|x\rangle|}$
    $x \leftarrow y$
  end if
end while
Return x

Lemma 92 and Corollary 93 are properties of Gillespie's algorithm. Lemma 92 describes the infinitesimal behavior of the CTMC. Roughly speaking, Lemma 92 says that at any time t where the current state is x, during a very short time h, the probability of staying in x is approximately $1 - |\langle x|G|x\rangle|h$. The probability of jumping to $y \neq x$ is approximately $\langle y|G|x\rangle h$. What's more, the probability that ξ changes its value more than twice during the short time h is negligible. The formal statement is as follows.

Lemma 92 (Gillespie's algorithm) *Let $S \subseteq \{0,1\}^n$ be the state space, $x_{in} \in S$, and $G \in \mathbb{R}^{|S|\times|S|}$ be a legal generator. Let h be an infinitesimal value. Consider the random variable ξ in Algorithm 4 w.r.t. (G, x_{in}, t).*

For any $s < t$, let $T(x,h)$ be the number of transitions between time $[s, s+h]$ conditioned on $\xi(s) = x$, that is, the number of times that ξ changes its values. Then

$$\begin{aligned}
&Pr(T(x,h) = 0) = 1 - |\langle x|G|x\rangle|h + O(h^2),\\
&Pr(T(x,h) \geq 2) = O(h^2),\\
&Pr(T(x,h) = 1, \xi(s+h) = y, y \neq x) = \langle y|G|x\rangle h + O(h^2).
\end{aligned}$$

Using Lemma 92, one can prove the following.

Corollary 93 (Gillespie's algorithm) *Let $S \subseteq \{0,1\}^n$ be the state space, $x_{in} \in S$ and $G \in \mathbb{R}^{|S|\times|S|}$ be a legal generator. For any $s \in [0,t]$, the random variable $\xi(s) \in S$ generated by the Gillespie's algorithm w.r.t. (G, x_{in}, t) is distributed according to*

$$\pi_s(x) := \langle x| \exp(Gs)|x_{in}\rangle, \forall x.$$

In other words, Gillespie's algorithm w.r.t. (G, x_{in}, t) simulates a random process with transition matrix $\exp(Gs)$. One can find the proofs for Lemma 92 and Lemma 93 in standard textbooks on Markov chain and random process.

Fixed Node Hamiltonian

It is well-known that one can connect stoquastic Hamiltonian to Markov chain [Bra+08]. However, in Definition 88 the n-qubit k-local Hamiltonian H can be general and might not be stoquastic. The fixed node quantum Monte Carlo method [Ten+95] is a method that transforms any real-valued Hamiltonian to a stoquastic Hamiltonian.

In this section, we give a brief review of the fixed node quantum Monte Carlo method, which will be used in our protocol later. For technical reasons, in this section, we do not directly consider n-qubit Hamiltonians. Instead we consider

- A set $S \subseteq \{0,1\}^n$.
- A real-valued symmetric[5] matrix $H \in \mathbb{R}^{|S|\times|S|}$.
- A real-valued state $|\phi\rangle \in \mathbb{R}^{|S|}$.

[5] Real-valued Hermitian matrix is real-valued symmetric matrix.

The fixed node Hamiltonian $F^{H,|\phi\rangle} \in \mathbb{R}^{|S|\times|S|}$ is defined as follows: for $x, y \in S$,

$$\langle x|F^{H,\phi}|y\rangle = \begin{cases} 0 & \text{if } (x,y) \in S^+; \\ \langle x|H|y\rangle, & \text{if } (x,y) \in S^-; \\ \langle x|H|x\rangle \quad + \sum_{z:(x,z)\in S^+} \langle x|H|z\rangle \frac{\langle z|\phi\rangle}{\langle x|\phi\rangle} & \\ & \text{if } x = y, \end{cases} \tag{4.7}$$

where

$$S^+ = \{(x,y) : x \neq y \text{ and } \langle\phi|x\rangle\langle x|H|y\rangle\langle y|\phi\rangle > 0\},$$
$$S^- = \{(x,y) : x \neq y \text{ and } \langle\phi|x\rangle\langle x|H|y\rangle\langle y|\phi\rangle \leq 0\}.$$

The key properties of $F^{H,\phi}$ are

Lemma 94 ([Ten+95], also in Lemma 2 of [Bra+23b]) *Given any real-valued symmetric matrix $H \in \mathbb{R}^{|S|\times|S|}$ and any real-valued **regularized**[6] unnormalized n-qubit state $|\phi\rangle \in \mathbb{R}^{|S|}$ [7], define the fixed node Hamiltonian $F^{H,\phi} \in \mathbb{R}^{|S|\times|S|}$ as above. Then*

(1) $F^{H,\phi}$ is symmetric and real-valued thus Hermitian. Besides, $F^{H,\phi}$ is stoquastic modulo a change of basis $|x\rangle \longrightarrow sign\left(\langle x|\phi\rangle\right)|x\rangle$.

(2) $F^{H,\phi}|\phi\rangle = H|\phi\rangle$.

(3) $\lambda(F^{H,\phi}) \geq \lambda(H)$ for any $|\phi\rangle$. If further $|\phi\rangle$ is the ground state of H, then $\lambda(F^{H,\phi}) = \lambda(H)$ and $|\phi\rangle$ is the ground state of $F^{H,\phi}$, that is,

$$F^{H,\phi}|\phi\rangle = \lambda(F^{H,\phi})|\phi\rangle .$$

For completeness, we put a proof in Appendix 4.8.

CTMC related to $F^{H,\phi}$

Define $F^{H,\phi} \in \mathbb{R}^{|S|\times|S|}$ as in Section 4.3. It is worth noting that the norm of $F^{H,\phi}$, *i.e.* $\|F^{H,\phi}\|$, can be exponentially large, since $\langle x|\phi\rangle$ may be exponentially small for some x. As we discussed in the proof overview in Section 4.1, the exponentially large norm influences the mixing time in the quantum Monte Carlo methods designed

[6] Otherwise Lemma 94 (1) is not a basis change thus not well-defined. In particular if $|\phi\rangle$=0, then $F^{H,\phi} = H$ and $F^{H,\phi}$ might not be stoquastic.

[7] $|\phi\rangle$ can be arbitrary state. It is not necessary to be the ground state.

for stoquastic Hamiltonians. To handle this problem, [Bra+23b] describes a generic way of converting a stoquastic Hamiltonian into a generator matrix of a CTMC. More details can be found in Lemma 3 of [Bra+23b]. As a concrete application, [Bra+23b] applies this method to the Fixed node Hamiltonian and defines a matrix $\widetilde{G}^{H,\phi} \in \mathbb{R}^{|S|\times|S|}$ based on $F^{H,\phi}$. Specifically, for any $x, y \in S$, define

$$\langle y|\widetilde{G}^{H,\phi}|x\rangle := \lambda(F^{H,\phi})\delta_{y,x} - \langle y|F^{H,\phi}|x\rangle \frac{\langle y|\phi\rangle}{\langle x|\phi\rangle}, \tag{4.8}$$

where $\delta_{y,x} = 1$ iff $x = y$ and $\delta_{y,x} = 0$ otherwise. To avoid confusion, we emphasize here that we will define a slightly different $G^{H,\phi}$ in the following section in Eq. (4.15), which is used in the final verification protocol. For now, we temporarily focus on $\widetilde{G}^{H,\phi}$. It is worth noting that $\widetilde{G}^{H,\phi}$ is not symmetric. The properties of $\widetilde{G}^{H,\phi}$ are summarized as follows.

Corollary 95 (Corollary of Lemma 3 in [Bra+23b]) *Given any real-valued symmetric matrix $H \in \mathbb{R}^{|S|\times|S|}$, and any real-valued **regularized** state $|\phi\rangle \in \mathbb{R}^{|S|}$. Define $\widetilde{G}^{H,\phi}$ as in Eq. (4.8).*

(1) The spectrum of $\widetilde{G}^{H,\phi}$ is the same as the spectrum of $\lambda(F^{H,\phi})I - F^{H,\phi}$.

(2) $\langle y|\widetilde{G}^{H,\phi}|x\rangle \geq 0$ for $y \neq x$. Further if $|\phi\rangle$ is an unnormalized ground state of H, then for any x, we have $\sum_y \langle y|\widetilde{G}^{H,\phi}|x\rangle = 0$. Thus $\widetilde{G}^{H,\phi}$ is a legal generator of a CTMC.

(3) If $|\phi\rangle$ is an unnormalized ground state of H. Define $c = \| |\phi\rangle \|^2$ and $\pi(x) = |\langle x|\phi\rangle|^2/c$. Then for any y, $\sum_x \langle y|\widetilde{G}^{H,\phi}|x\rangle\pi(x) = 0$, thus π is a stationary distribution of the CTMC w.r.t. $\widetilde{G}^{H,\phi}$.

The key result in [Bra+23b] is proving Gillespie's algorithm, *i.e.* Algorithm 4, can efficiently simulate the CTMC w.r.t. $\widetilde{G}^{H,\phi}$. The formal statement is in Lemma 96. For completeness, we put a proof in Appendix 4.9.

Lemma 96 (Lemma 4 in [Bra+23b]) *Given any real-valued d-sparse symmetric matrix $H \in \mathbb{R}^{|S|\times|S|}$, and real-valued **regularized** unnormalized state $|\phi\rangle \in \mathbb{R}^{|S|}$, where $|\phi\rangle$ is a ground state of H. Define $\widetilde{G}^{H,\phi}$ as in Eq. (4.8).*

For any x_{in}, t, denote $\kappa(x_{in}, t)$ as the number of transitions in Algorithm 4 w.r.t. $(\widetilde{G}^{H,\phi}, x_{in}, t)$, that is, the number of times that $\xi(s)$ changes its value. Then there exists $\hat{x}_{in} \in S$ such that $\langle \hat{x}_{in}|\phi\rangle \neq 0$ and the number of transitions satisfies that

$$Pr\left(\kappa\left(\hat{x}_{in}, t\right) \leq dn^3 t\|H\|\right) \geq \frac{1}{2}. \tag{4.9}$$

Note that Lemma 96 is highly non-trivial, since the norm of $\widetilde{G}^{H,\phi}$ can still be exponentially large. In particular, in Algorithm 4 line 8, $\Delta\tau$ may be exponentially small. Since the running time of Algorithm 4 is proportional to the number of transitions, exponentially small $\Delta\tau$ may cause Gillespie's algorithm w.r.t. ($\widetilde{G}^{H,\phi}$, x_{in}, t) to take exponentially long time, even for $t = O(poly(n))$. In the CTMC literature, it means that the CTMC may transit exponentially many times in a small time interval. However, Lemma 96 proves that with high probability, the number of transitions is bounded, when H is sparse and has a small norm.

4.4 The MA verification protocol

In this section, we describe a **MA** verification protocol for deciding the local Hamiltonian problem with succinct ground state, which is defined in Definition 88. Firstly in Section 4.4, we will argue that w.l.o.g., we can assume that the instance (H, a, b) from Definition 88 satisfying

$$a = 0 \text{ and } b \geq 1/poly(n).$$

Then in Section 4.4 and Section 4.4, we will give two verification protocols based on different assumptions. In particular,

- In Section 4.4 we assume Assumption (i): we can sample u from the uniform distribution $[0, 1]$ in $poly(n)$ time.
- In Section 4.4 we substitute Assumption (i) by Assumption (ii): we assume that we can use $poly(n)$-time to sample from the truncated discretized exponential distribution $\mathcal{D}_{K,\delta,\lambda}$ with parameters specified in Remark 7 in Appendix 4.11.

In both Section 4.4 and Section 4.4, we show that there is a $poly(n)$-time algorithm which takes inputs in the form of an n-qubit local Hamiltonian H, a $poly(n)$-size circuit C_ϕ for a succinct state $|\phi\rangle$, and an n-bit string x_{in}, such that

- If (H, a, b) is a Yes instance, there exists C_ϕ and x_{in} s.t. the algorithm accepts w.p. $\geq constant$;
- If (H, a, b) is a No instance, then for any C_ϕ and x_{in}, the accepting probability is exponentially small.

The proof in Section 4.4 captures the key technical ideas. The proof in Section 4.4 further addresses the errors introduced by discretization and is our final proof for Theorem 89.

Preprocessing

Given an instance (H^*, a^*, b^*), where $H^* = \sum_{j=1}^{m} H_j^*$ is an n-qubit $(k-1)$-local Hamiltonian, m is the number of terms in H. Here k is a constant, $m = poly(n)$ and $\|H^*\| = poly(n)$.

By Section 4.3, w.l.o.g. we assume that H^* are real-valued symmetric $2^k m$-sparse matrix, and is of size $2^n \times 2^n$. Besides, the ground state of H^*, denoted as $|\phi^*\rangle \in \mathbb{R}^{2^n}$, is also real-valued.

For the **MA** protocol, the witnesses are

(i) A real value λ^*, which is supposed to be $\lambda(H^*)$[8]. W.l.o.g. we assume that $\lambda^* \leq a^*$, otherwise we reject immediately.

(ii) A $poly(n)$-size circuit C_{ϕ^*} which succinctly represents the state $|\phi^*\rangle$. $|\phi^*\rangle$ is supposed to be the ground state of H^*.

(iii) A computational basis $x_{in}^* \in \{0,1\}^n$, which is supposed to satisfy Lemma 96 w.r.t. $(G^{H_S,\phi_S}, x_{in}, t)$ which will be defined later. In particular,

- $H_S, |\phi_S\rangle, x_{in}$ are defined in the following paragraphs.
- G^{H_S,ϕ_S} is defined in Eq. (4.15).
- Let

$$\epsilon := b^* - a^* = 1/poly(n), \tag{4.10}$$

$$t := 8(n + p'(n))/\epsilon, \tag{4.11}$$

where $p'(n) = poly(n)$ is defined in Remark 8 in Appendix 4.11.

Specifically, define

$$H := H^* - \lambda^* I, \tag{4.12}$$

$$|\phi\rangle := |\phi^*\rangle, \tag{4.13}$$

$$x_{in} := x_{in}^*. \tag{4.14}$$

The following properties hold:

- In the Yes instance, since $\lambda^*, |\phi^*\rangle$ are the ground energy and the ground state of H^*, we have $\lambda(H) = 0$ and $|\phi\rangle$ is the ground state of H. That is,

$$H|\phi\rangle = 0.$$

[8] By Remark 6 in Appendix 4.11, $\lambda(H^*)$ can be represented by $poly(n)$ bits exactly.

- In the No instance, since $\lambda(H^*) \geq b^*$ and $\lambda^* \leq a^*$. We have

$$\lambda(H) \geq \epsilon.$$

Define

$$S := Supp(|\phi\rangle).$$

Define $H_S \in \mathbb{R}^{|S|\times|S|}$ to be the submatrix of H whose rows and columns are taken from the set S. Similarly we define $|\phi_S\rangle$ from $|\phi\rangle$. The main reason we define $H_S, |\phi_S\rangle$ is to make the proof rigorous. Feel free to assume that $|\phi\rangle$ is regularized, and thus $H_S = H$, $|\phi_S\rangle = |\phi\rangle$. One can check the following.

Claim 97 *Suppose $S \neq \emptyset$,*

- *H_S is symmetric real-valued, $2^k m$-sparse, where $m = poly(n)$ and $\|H_S\| = poly(n)$. Besides, $|\phi_S\rangle$ is real-valued and regularized.*
- *In the Yes instance, $H_S|\phi_S\rangle = 0$, $\lambda(H_S) = 0$[9]. $|\phi_S\rangle$ is a ground state of H_S.*
- *In the No instance,*

$$\lambda(H_S) \geq \lambda(H) \geq \epsilon = 1/poly(n).$$

- *For $x, y \in S$, $\langle x|H_S|y\rangle$ and $C_{\phi_S}(x)$ can be easily computed in $poly(n)$ time, which is obtained by directly computing $\langle x|H|y\rangle$ and $C_\phi(x)$.*

Proof: It suffices to notice that by Fact 91, when $S \neq \emptyset$, we have

$$\lambda(H) \leq \lambda(H_S) \leq \lambda_{max}(H),$$
$$\|H_S\| \leq \|H\| \leq poly(n).$$

■

Define F^{H_S,ϕ_S} as in Eq. (4.7). Note that here instead of using $\widetilde{G}^{H_S,\phi_S}$ as in Eq. (4.8), we will use G^{H_S,ϕ_S} defined below:

$$\langle y|G^{H_S,\phi_S}|x\rangle := -\langle y|F^{H_S,\phi_S}|x\rangle \frac{\langle y|\phi_S\rangle}{\langle x|\phi_S\rangle}. \tag{4.15}$$

[9]Since $\lambda(H_S) \geq \lambda(H) = 0$ by Fact 91, and 0 is achieved by $|\phi_S\rangle$.

The main reason that we use G^{H_S,ϕ_S} instead of $\widetilde{G}^{H_S,\phi_S}$ is our setting is different from in [Bra+23b]. More specifically, in [Bra+23b] $|\phi_S\rangle$ is trusted and is always the ground state, in this situation

$$\lambda(F^{H_S,\phi_S}) = \lambda(H_S).$$

By Lemma 94 (2), the value $\lambda(F^{H_S,\phi_S})$ can be computed by

$$\lambda(F^{H_S,\phi_S}) = \lambda(H_S) = \frac{\langle x|H_S|\phi_S\rangle}{\langle x|\phi_S\rangle} = \sum_y \langle x|H_S|y\rangle \frac{\langle y|\phi_S\rangle}{\langle x|\phi_S\rangle}, \tag{4.16}$$

where $x \in S$ satisfying $\langle x|\phi_S\rangle \neq 0$. However, in our setting, in the No instance, $|\phi_S\rangle$ can be any adversary state. The "$\lambda(F^{H_S,\phi_S})$" calculated by Eq. (4.16) cannot be trusted and might break the soundness of the protocol.

Here are some other remarks. Note that in the Yes instance, Eq. (4.8) and Eq. (4.15) coincide since $\lambda(F^{H_S,\phi_S}) = 0$. It is worth noting that G^{H_S,ϕ_S} is not symmetric. However, since F^{H_S,ϕ_S} is symmetric, by similar argument as Corollary 95 (1) we have the following.

Fact 98 *G^{H_S,ϕ_S} is always diagonalizable and has the same spectrum as $-F^{H_S,\phi_S}$.*

As explained in Remark 8 in Appendix 4.11, there exists a polynomial $p'(n) \gg p(n)$ such that

$$amax(G^{H_S,\phi_S}) \leq 2^{p'(n)},$$
$$\|G^{H_S,\phi_S}\|, \|F^{H_S,\phi_S}\| \leq 2^{p'(n)+2n}.$$

Besides, if $\langle x|G^{H_S,\phi_S}|y\rangle \neq 0$, then

$$\langle x|G^{H_S,\phi_S}|y\rangle \geq 1/2^{p'(n)}.$$

Protocol with continuous-time randomness

In this section, we assume that we can sample u from the uniform distribution $[0, 1]$ in $poly(n)$ time. Let H_S, $|\phi_S\rangle$ be the Hamiltonian and the state after the preprocessing procedure in Section 4.4 Claim 97.

The verification protocol is shown in Algorithm 5. The protocol is very similar to the truncated version of Gillespie's algorithm used in [Bra+23b]. The key difference is, instead of returning x, we add a checking procedure in line 11, and this algorithm returns Accept/Reject. Besides, we use G^{H_S,ϕ_S} defined in Eq. (4.15) instead of $\widetilde{G}^{H_S,\phi_S}$ in Eq. (4.8).

Compared to the setting in [Bra+23b], it is worth noting that in our setting, $|\phi_S\rangle$ can be adversarial which might not be the ground state, thus G^{H_S,ϕ_S} might not be a legal generator. The first thing to check is that Algorithm 5 is well-defined:

Claim 99 *Algorithm 5 is always well-defined and each line can be performed in poly(n) time.*

Proof: By line 4, w.l.o.g. we assume that $S \neq \emptyset$ otherwise we reject immediately. Then G^{H_S,ϕ_S}, F^{H_S,ϕ_S} are well-defined since $S \neq \emptyset$. Also note that line 4 implies a good x_{in} which is not rejected by Line 5 should satisfy

$$\langle x_{in}|\phi\rangle \neq 0.$$

Furthermore, by definition of G^{H_S,ϕ_S}, that is, Eq (4.15), we have that all the x visited by the algorithm in Line 25 satisfies $\langle x|\phi\rangle \neq 0$.

For line 11, since G^{H_S,ϕ_S} is $poly(n)$-sparse, one can use poly-time to

- list all z where $\langle z|G^{H_S,\phi_S}|x\rangle \neq 0$,
- compute $\sum_{z\in S}\langle z|G^{H_S,\phi_S}|x\rangle$ and check whether one of them is strictly negative.

Similarly since G^{H_S,ϕ_S} is $poly(n)$-sparse, in line 24, one can use poly-time to list all y where $\langle y|G^{H_S,\phi_S}|x\rangle \neq 0$, and efficiently sample y from the distribution.

Besides, when the conditions in line 11 are not satisfied, it is guaranteed that the sampling procedure in line 24 is always well-defined, that is,

$$\sum_{y\neq x}\frac{\langle y|G^{H_S,\phi_S}|x\rangle}{|\langle x|G^{H_S,\phi_S}|x\rangle|} = 1 \text{ and } \frac{\langle y|G^{H_S,\phi_S}|x\rangle}{|\langle x|G^{H_S,\phi_S}|x\rangle|} \geq 0 \text{ for } y \neq x.$$

Besides, by definition $\frac{\langle y|G^{H_S,\phi_S}|x\rangle}{|\langle x|G^{H_S,\phi_S}|x\rangle|} > 0$ implies $y \in S$. Thus all the "x" that ξ visits in Algorithm 5 are all in S. Thus all the lines querying entries of $G^{H_S,\phi_S}|x\rangle$ are well-defined.

■

The performance of Algorithm 5 is summarized as follows. The proof is in Section 4.4.

Theorem 100 *For any (H_S, ϕ_S, x_{in}), Algorithm 5 always runs in polynomial time. Besides,*

Algorithm 5 Checking(H_S, ϕ_S, x_{in})

$\kappa \leftarrow 0$ ▷ Record the number of transitions
$x \leftarrow x_{in}$, $\tau \leftarrow 0$, $\xi(0) \leftarrow x_{in}$
Set $t = 8(n + p'(n))/\epsilon$ and $M = 2^k mn^3 t\|H\|$
if $x \notin S$ **then** ▷A good x_{in} should satisfy $\langle x_{in}|\phi\rangle \neq 0$
  Return Reject
end if
while $\tau < t$ **do**
  if $C_{\phi_S}(x)$ is not represented by $p(n)$-bits **then**
    Return Reject ▷Check the format of $C_{\phi_S}(x)$
  end if
  if $\sum_{z\in S}\langle z|G^{H_S,\phi_S}|x\rangle \neq 0$ or $\exists y \neq x, y \in S$ s.t. $\langle y|G^{H_S,\phi_S}|x\rangle < 0$ **then**
    Return Reject ▷Add a Check
  end if
  if $\kappa \geq M$ **then**
    Return Reject
  end if
  if $|\langle x|G^{H_S,\phi_S}|x\rangle| = 0$ **then**
    Set $\xi(s) = x$ for all $s \in (\tau, t]$
    $\tau \leftarrow t$
  else
    Sample $u \in [0, 1]$ from the uniform distribution $[0, 1]$
    $\Delta\tau \leftarrow \frac{\ln(1/u)}{|\langle x|G^{H_S,\phi_S}|x\rangle|}$
    Set $\xi(s) = x$ for all $s \in (\tau, \tau + \Delta\tau]$
    $\tau \leftarrow \tau + \Delta\tau$
    Sample $y \in S \setminus \{x\}$ from the probability distribution $\frac{\langle y|G^{H_S,\phi_S}|x\rangle}{|\langle x|G^{H_S,\phi_S}|x\rangle|}$
    $x \leftarrow y$
    $\kappa \leftarrow \kappa + 1$
  end if
end while
Return Accept

- *For the Yes instance, there exists ϕ_S, x_{in} such that Algorithm 5 accepts w.p. $\geq 1/2$.*
- *In the No instance, $\forall\ \phi_S$, x_{in}, Algorithm 5 rejects w.p. $\geq 1 - 2^{-n}$.*

Analysis

Note that the number of iterations in Algorithm 5 is bounded by $M = poly(n)$. By Claim 99 each line can be performed in $poly(n)$-time, thus the algorithm always runs in polynomial time.

In the following, we say a string x "pass line 11" if x does **not** satisfy the conditions in line 11, thus will not be rejected immediately in line 12. In the following we give the proof of completeness and soundness.

Proof:[of Completeness of Theorem 100.] In the Yes instance, we have that

$$\lambda(H_S) = 0 \text{ and } |\phi_S\rangle \text{ is the ground state.}$$

By Lemma 94 (3),

$$\lambda(F^{H_S,\phi_S}) = \lambda(H_S) = 0.$$

Thus G^{H_S,ϕ_S} coincides with $\widetilde{G}^{H_S,\phi_S}$, see Eq. (4.8) and Eq. (4.15). By Lemma 96, we know that there exists $x_{in} \in S$ such that

$$\langle x_{in}|\phi_S\rangle \neq 0, \tag{4.17}$$

$$Pr\left(\kappa\left(x_{in}, t\right) \leq M\right) \geq \frac{1}{2}. \tag{4.18}$$

Besides, by Corollary 95 (2), we know that

$$\forall x \in S, \sum_{z\in S}\langle z|G^{H_S,\phi_S}|x\rangle = 0,$$
$$\forall y \neq x, x \in S, y \in S, \langle y|G^{H_S,\phi_S}|x\rangle \geq 0.$$

Thus the checks in line 4, 11 in Algorithm 5 are always passed. The check in line 14 in Algorithm 5 is passed with probability greater than $\frac{1}{2}$. Thus in summary, the probability of accepting is greater than $1/2$.

■

Proof:[of Soundness of Theorem 100.] W.l.o.g. assume line 4 is passed, otherwise the protocol rejects immediately. Thus we have

$$\langle x_{in}|\phi_S\rangle \neq 0 \text{ thus } |\phi_S\rangle \neq 0.$$

In the No instance, we know

$$\lambda(H_S) \geq \epsilon = 1/poly(n), \tag{4.19}$$

$|\phi_S\rangle$ can be an arbitrary adversary state, and $H_S|\phi_S\rangle \neq 0$.

In the No case G^{H_S,ϕ_S} might not be a legal generator of a CTMC. To analyze the soundness, we consider another algorithm without the check in line 14, that is,

deleting lines 14, 15 and 16. Denote this new algorithm as Algorithm 5*. The accepting probability of Algorithm 5 can only be less than Algorithm 5* [10].

Define S_{good} be the set of strings which pass line 11. That is,

$$S_{good} := \{x \in S | \sum_{y \in S} \langle y|G^{H_S,\phi_S}|x\rangle = 0, \\ \forall y \neq x, y \in S, \langle y|G^{H_S,\phi_S}|x\rangle \geq 0\}. \tag{4.20}$$

W.o.l.g. we assume that the first execution of line 11 is passed, otherwise the protocol rejects immediately. Thus we have

$$\xi(0) = x_{in} \in S_{good}.$$

Let G^{H_S,ϕ_S}_{good} and F^{H_S,ϕ_S}_{good} be the submatrix of $G^{H_S,\phi_S}, F^{H_S,\phi_S}$, where the row and column indices are in S_{good}. Let $\left|\phi_{S_{good}}\right\rangle$ be the state restricting $|\phi_S\rangle$ in S_{good}. It is worth noting that

$$G^{H_S,\phi_S}_{good} \neq G^{H_{S_{good}},\phi_{S_{good}}}, \tag{4.21}$$

since the diagonal elements are different. Instead,

$$G^{H_S,\phi_S}_{good} = Diag(\phi_{S_{good}})(-F^{H_S,\phi_S}_{good})Diag(\phi_{S_{good}})^{-1}. \tag{4.22}$$

We claim that

Claim 101

$$\lambda_{max}(-F^{H_S,\phi_S}_{good}) \leq -\epsilon, \\ \|\exp(-F^{H_S,\phi_S}_{good})\| \leq 1 - \epsilon/2.$$

where $\epsilon = 1/poly(n) \leq 1/2$,

Proof: By definition $\left|\phi_{S_{good}}\right\rangle$ is regularized. Note that by Lemma 94 (3), we have

$$\lambda(F^{H_S,\phi_S}) \geq \lambda(H_S) \geq \epsilon. \tag{4.23}$$

[10] Imagine running Algorithm 5 without any checks, that is, without line 4, 11 and 14, and we write down all the possible logs on a paper. Each log corresponds to a probability. What the checking procedure does is assigning reject to some logs. Less check, Less reject. It is possible that Algorithm 5* runs in exponential time. We do not care about the efficiency of Algorithm 5*, we only use it as a technique to bound the accepting probability of Algorithm 5.

Since $-F^{H_S,\phi_S}$ is symmetric and thus Hermitian, by Fact 91 we have

$$\begin{aligned}\lambda_{max}(-F^{H_S,\phi_S}_{good}) &\leq \lambda_{max}(-F^{H_S,\phi_S}) \\ &= -\lambda(F^{H_S,\phi_S}) \\ &\leq -\epsilon.\end{aligned}$$

Note that

$$\exp(-x) \leq 1 - x/2 \text{ for } x \leq 1/2,$$

and by definition of $\exp(\cdot)$, we know all eigenvalues[11] of $\exp(-F^{H_S,\phi_S}_{good})$ is non-negative, thus we conclude that[12]

$$\|\exp(-F^{H_S,\phi_S}_{good})\| \leq 1 - \epsilon/2 \text{ for } \epsilon \leq 1/2.$$

■

In the following, we show that, although G^{H_S,ϕ_S} does not correspond to a legal generator, Algorithm 5* w.r.t. to G^{H_S,ϕ_S} still have similar infinitesimal properties as Lemma 92. The properties are summarized in Claim 102.

To describe Claim 102, we define some notations. Consider running Algorithm 5* w.r.t.

$$(G^{H_S,\phi_S}, \phi_S, x_{in}),$$

for some x_{in}[13]. Let τ_{end} be the value of τ when Algorithm 5* returns Accept/Reject. Let $\xi(s), s \in [0, \tau_{end}]$ be the $\xi(s)$ in Algorithm 5*. We know that

$$\xi(s) \in S_{good}, \forall s \in [0, \tau_{end}).$$

To clarify, τ_{end} and ξ are random variables. For any fixed s, t, let c_1 be the first time that ξ changes its value after time s, conditioned on $\tau_{end} \geq s$ and $\xi(s) = x$, that is,

$$c_1 = \min\{\eta : \eta > s, \xi(\eta) \neq x | \tau_{end} > s, \xi(s) = x\}.$$

Similarly, define c_2 to be the second time that ξ changes its value after time s, that is,

$$c_2 = \min\{\eta : \eta > c_1, \xi(\eta) \neq \xi(c_1) | \tau_{end} > s, \xi(s) = x\}.$$

[11] F^{H_S,ϕ_S}_{good} is diagonalizable since it is Hermitian.

[12] $\exp(-F^{H_S,\phi_S}_{good})$ is Hermitian, thus its spectral norm is its maximum absolute value of eigenvalues. It is worth noting that $\exp(G^{H_S,\phi_S}_{good})$ is not Hermitian, although it has the same spectrum as $\exp(-F^{H_S,\phi_S}_{good})$, they do not have the same spectral norm.

[13] A probability distribution of x_{in} will not get a higher acceptance probability than one particular x_{in} which maximize the accepting probability.

Claim 102 *For any fixed $s < t$, any x, let h be an infinitesimal value. Use notations defined above. Conditioned on $\tau_{end} > s$ and $\xi(s) = x$, we have*[14]

- *The probability that $\tau_{end} \geq s+h$ and ξ does not change value in time $[x, x+h]$ is*

$$\begin{aligned} &Pr(c_1 \geq s+h | \tau_{end} > s, \xi(s) = x) \\ &= 1 - |\langle x|G^{H_S,\phi_S}|x\rangle|h + O(h^2). \end{aligned} \tag{4.24}$$

- *For $y \neq x, y \notin S_{good}$, the probability that Algorithm 5 ends in time between $[s, s+h]$ by hitting y is*

$$\begin{aligned} &Pr(c_1 \leq s+h, \xi(c_1) = y | \tau_{end} > s, \xi(s) = x) \\ &= \langle y|G^{H_S,\phi_S}|x\rangle h + O(h^2). \end{aligned} \tag{4.25}$$

- *For $y \neq x, y \in S_{good}$, the probability that ξ hits y in time $c_1 \in [s, s+h]$, and keeps in y in $[c_1, s+h]$ is*

$$\begin{aligned} &Pr(c_1 \leq s+h, \xi(c_1) = y, c_2 \geq s+h | \tau_{end} > s, \xi(s) = x) \\ &= \langle y|G^{H_S,\phi_S}|x\rangle h + O(h^2). \end{aligned} \tag{4.26}$$

- *The probability of other events, that is, the probability that ξ changes its value more than once*[15] *in $[s, s+h]$ is*

$$Pr(c_2 \leq s+h | \tau_{end} > s, \xi(s) = x) = O(h^2). \tag{4.27}$$

We rigorously prove Claim 102 in Appendix 4.10 using properties of the exponential distribution. On the other hand, one can intuitively imagine the correctness of Claim 102 from Lemma 92: although G^{H_S,ϕ_S} may not be a legal generator, one can consider another legal generator $G \in \mathbb{R}^{|S|\times|S|}$ which is obtained by setting column $\langle *|G^{H_S,\phi_S}|x\rangle$ to 0 for $x \notin S_{good}$. Lemma 92 holds for this legal generator G, and notice that the Algorithm 5* behaves the same w.r.t. G^{H_S,ϕ_S} as w.r.t. G conditioned on it never hits $x \notin S_{good}$.

With Claim 102 we can prove:

[14] We use c_1 to state this theorem instead of using the number of transitions as in Lemma 92, since our algorithm ends immediately when it hits a not-good string. It's a bit tricky to use the notion of number of transitions in time $[s, s+h]$ here.

[15] ξ may or may not hit a not-good string in c_2.

Claim 103 *The accepting probability of Algorithm 5* w.r.t.* $(G^{H_S,\phi_S}, \phi_S, x_{in})$ *is less than* $1/2^n$.

Proof: Consider the random process generated by running Algorithm 5* w.r.t. $(G^{H_S,\phi_S}, \phi_S, x_{in})$. w.l.o.g. assume that $x_{in} \in S$ otherwise line 4 rejects immediately.

Let $P_x(s)$ be the probability that $\tau_{end} \geq s$ and $\xi(s) = x$. Note that by definition of Algorithm 5*,

$$\xi(\tau) \in S_{good}, \forall \tau \in [0, \tau_{end}].$$

Let $\left|P_{good}(s)\right\rangle \in \mathbb{R}^{|S_{good}|}$ be the vector

$$[..., P_x(s), ...]^T \text{ for } x \in S_{good}.$$

Let h be an infinitesimal value, by Claim 102, we have for any $z \in S_{good}$,

$$\begin{aligned} P_z(s+h) = & \underbrace{P_z(s)\left(1 - |\langle z|G^{H_S,\phi_S}|z\rangle|h + O(h^2)\right)}_{\xi(s)=z, \text{ stays in } z \text{ till } s+h} \\ & + \underbrace{\sum_{x:x\neq z, x\in S_{good}} P_x(s)\left(\langle z|G^{H_S,\phi_S}|x\rangle h + O(h^2)\right)}_{\xi(s)=x, \text{ jump to } z \text{ between time } [s,s+h], \text{ stays in } z \text{ till } s+h} \\ & + \underbrace{O(h^2)}_{\substack{\xi(s)\in S_{good}, \text{ jump more than once in } [s,s+h] \\ \text{but finally stay } z \text{ in } s+h}}. \end{aligned} \tag{4.28}$$

Note that by the definition of S_{good}, we have

$$\begin{aligned} \langle x|G^{H_S,\phi_S}|x\rangle &= -\sum_{y\neq x, y\in S} \langle y|G^{H_S,\phi_S}|x\rangle \\ &\leq 0. \end{aligned}$$

Thus we have for $z \in S_{good}$,

$$-|\langle z|G^{H_S,\phi_S}|z\rangle| = \langle z|G^{H_S,\phi_S}|z\rangle.$$

Thus Eq. (4.28) is equivalent to

$$\begin{aligned} P_z(s+h) - P_z(s) &= \sum_{x\in S_{good}} \langle z|G^{H_S,\phi_S}|x\rangle P_x(s)h + O(h^2) \\ &= \sum_{x\in S_{good}} \langle z|G^{H_S,\phi_S}_{good}|x\rangle P_x(s)h + O(h^2), \end{aligned}$$

where the last equality comes from the fact that $z, x \in S_{good}$. Thus

$$\begin{aligned} P'_z(s) &= \langle z|G_{good}^{H_S,\phi_S}|P_{good}(s)\rangle, \\ \Rightarrow P'_{good}(s) &= G_{good}^{H_S,\phi_S}\left|P_{good}(s)\right\rangle, \\ \Rightarrow P_{good}(s) &= \exp(G_{good}^{H_S,\phi_S}s)\,|x_{in}\rangle\,. \end{aligned}$$

Thus

$$\begin{aligned} &Pr(\text{Algorithm } 5^* \text{ Accept}) \qquad (4.29)\\ &= Pr(\tau_{end} > t) \\ &= \sum_{z\in S_{good}} P_z(t) \\ &= \sum_{z\in S_{good}} \langle z|\exp(G_{good}^{H_S,\phi_S}t)|x_{in}\rangle \\ &= \sum_{z\in S_{good}} \langle z|Diag(\phi_{S_{good}})\exp(-F_{good}^{H_S,\phi_S}t)Diag(\phi_{S_{good}})^{-1}|x_{in}\rangle \\ &= \sum_{z\in S_{good}} \frac{\langle z|\phi_S\rangle}{\langle x_{in}|\phi_S\rangle}\langle z|\exp(-F_{good}^{H_S,\phi_S}t)|x_{in}\rangle. \end{aligned}$$

By Claim 101 and Remark 6 in Appendix 4.11 we know that

$$\begin{aligned} &\sum_{z\in S_{good}} \frac{\langle z|\phi_S\rangle}{\langle x_{in}|\phi_S\rangle}\langle z|\exp(-F_{good}^{H_S,\phi_S}t)|x_{in}\rangle \qquad (4.30)\\ &\leq \sum_{z\in S_{good}} \left|\frac{\langle z|\phi_S\rangle}{\langle x_{in}|\phi_S\rangle}\right| \cdot \left|\langle z|\exp(-F_{good}^{H_S,\phi_S}t)|x_{in}\rangle\right| \\ &\leq 2^n \cdot 2^{3p(n)} \cdot (1-\epsilon/2)^t \qquad (4.31)\\ &\leq 2^{-n}, \end{aligned}$$

for sufficiently large

$$t \geq 8(n+p'(n))/\epsilon \geq 8(n+p(n))/\epsilon,$$

where $\epsilon \leq 1/2$. ■

Thus by Claim 103 finally we conclude that the accepting probability of Algorithm 5, is smaller than the accepting probability of Algorithm 5^*, which is smaller than 2^{-n}.

■

Protocol with discrete-time randomness

Recall that we denote $S \subseteq \{0,1\}^n$ as the state space, $G \in \mathbb{R}^{|S|\times|S|}$ as a matrix, and $x, y \in S$ as the states. Additionally, $p'(n) = poly(n)$ is a precision parameter explained in Appendix 4.11 Remark 8, and M is the upper bound of the number of transitions used in Algorithm 5,

$$M = 2^k mn^3 \cdot 8(n + p'(n))/\epsilon \cdot \|H\| = poly(n).$$

In this section, we replace the assumption from Section IV B—that one can sample uniformly from $[0,1]$ in $poly(n)$ time, and thus can sample from an exponential distribution in $poly(n)$ time—with its discrete approximation. In particular, we assume that we can use $poly(n)$-time to sample from the truncated discretized exponential distribution $\mathcal{D}_{K,\delta,\lambda}$ as in Remark 7 in Appendix 4.11. The value of λ will be specified later in the algorithm, while K and δ are chosen to be

$$\delta := 2^{-2n}/M, \tag{4.32}$$

$$K := \lceil 2^{p'(n)+2n} M\,(n \ln 2 + \ln M)) \rceil. \tag{4.33}$$

Note that $\mathcal{D}_{K,\delta,\lambda}$ serves as a discrete approximation of the exponential distribution in the following sense.

Claim 104 *Sample a random variable $\Delta\tau$ according to the exponential distribution with parameter λ. Let $\Delta\tau_D$ be the rounded value of $\Delta\tau$, which is the largest value in the set $\{k\delta\}_{k=0,\ldots,K}$ that does not exceed $\Delta\tau$. Then the distribution of $\Delta\tau_D$ is $\mathcal{D}_{K,\delta,\lambda}$.*

Here, the subscript D in $\Delta\tau_D$ refers to the "discrete approximation".

The discretized **MA** verification protocol is derived by modifying Algorithm 5: First, for clarity, we rename the variables $(\tau, \Delta\tau, \xi)$ to $(\tau_D, \Delta\tau_D, \xi_D)$. Due to the discretization error, in line 3 we change the value of t from $8(n + p'(n))/\epsilon$ to $(-2^{-n}) + 8(n + p'(n))/\epsilon$. Finally, we replace the continuous-time process (lines 21, 22 and 23 in Algorithm 5) with its discrete approximation, as shown below.

Sample $\Delta\tau_D$ from $\mathcal{D}_{K,\delta,|\langle x|G^{H_S,\phi_S}|x\rangle|}$
Set $\xi_D(s) = x$ for all $s \in (\tau_D, \tau_D + \Delta\tau_D]$
$\tau_D \leftarrow \tau_D + \Delta\tau_D$

We denote the discretized **MA** verification protocol as Algorithm 5D. The performance of Algorithm 5D is summarized as follows.

Theorem 105 *For any (H_S, ϕ_S, x_{in}), Algorithm 5D always runs in polynomial time. Besides,*

- *For the Yes instance, there exists ϕ_S, x_{in} such that Algorithm 5D accepts w.p. $\geq 1/2 - 2^{-n}$.*
- *In the No instance, $\forall\ \phi_S$, x_{in}, Algorithm 5D rejects w.p. $\geq 1 - 3 \cdot 2^{-n}$.*

Connecting Algorithm 5 and Algortihm 5D

Based on Claim 104 we can interpret the discretized distribution $\mathcal{D}_{K,\delta,\lambda}$ as being derived from rounding the exponential distribution. Using this perspective[16], we will prove Theorem 105 by showing that Algorithm 5D is a good discrete approximation of Algorithm 5.

More specifically, to aid in the proof of Theorem 105, we define a new algorithm that couples the continuous and the discrete processes. The full description is provided in Algorithm 6. Algorithm 6 is derived by modifying specific lines in Algorithm 5 as follows.

① Change line 2 to contain the variables for both the continuous and discrete process:

$$x \leftarrow x_{in}, \tau \leftarrow 0, \xi(0) \leftarrow x_{in}$$
$$\tau_D \leftarrow 0, \xi_D(0) \leftarrow x_{in}$$

② Delete the sentence in line 3. To keep Algorithm 6 flexible as an analytical tool, we do not assign specific values to t and M. Accordingly, we modify line 7 to "While True" which means loop forever. Additionally, we remove the sentences in lines 14, 15 and 16, and delete line 29.

③ Modify lines 21, 22 and 23 to include the updates for both the continuous and discrete processes. In particular, the discrete process is derived by rounding the continuous process:

Sample $u \in [0, 1]$ from uniform distribution $[0, 1]$
$\Delta\tau \leftarrow \frac{\ln(1/u)}{|\langle x|G^{H_S,\phi_S}|x\rangle|}$
Set $\xi(s) = x$ for all $s \in (\tau, \tau + \Delta\tau]$

[16] We thank the anonymous reviewers for suggesting this connection, which significantly simplifies the proof.

Algorithm 6 Checking_Compare(H_S, ϕ_S, x_{in})

$\kappa \leftarrow 0$ ▷Record the number of transitions
$x \leftarrow x_{in}, \tau \leftarrow 0, \xi(0) \leftarrow x_{in}$
$\tau_D \leftarrow 0, \xi_D(0) \leftarrow x_{in}$
▷Do not specify M, t.
if $x \notin S$ **then** ▷A good x_{in} should satisfy $\langle x_{in}|\phi\rangle \neq 0$
  Return Reject
end if
while True **do**
  if $C_{\phi_S}(x)$ is not represented by $p(n)$-bits **then**
    Return Reject ▷Check the format of $C_{\phi_S}(x)$.
  end if
  if $\sum_{z\in S}\langle z|G^{H_S,\phi_S}|x\rangle \neq 0$ or $\exists y \neq x, y \in S$ s.t. $\langle y|G^{H_S,\phi_S}|x\rangle < 0$ **then**
    Return Reject ▷Add a Check
  end if
  ▷Delete the check $\kappa \geq M$

  if $|\langle x|G^{H_S,\phi_S}|x\rangle| = 0$ **then**
    Set $\xi(s) = x$ for all $s \in (\tau, +\infty]$;
    Set $\xi_D(s) = x$ for all $s \in (\tau_D, +\infty]$;
    $\tau \leftarrow +\infty$
    $\tau_D \leftarrow +\infty$
  else
    Sample $u \in [0, 1]$ from uniform distribution $[0, 1]$
    $\Delta\tau \leftarrow \frac{\ln(1/u)}{|\langle x|G^{H_S,\phi_S}|x\rangle|}$
    Set $\xi(s) = x$ for all $s \in (\tau, \tau + \Delta\tau]$
    $\tau \leftarrow \tau + \Delta\tau$
    Let $\Delta\tau_D$ be the rounded value of $\Delta\tau$, which is the largest value in the set $\{k\delta\}_{k=0,\ldots,K}$ that does not exceed $\Delta\tau$.
    Set $\xi_D(s) = x$ for all $s \in (\tau_D, \tau_D + \Delta\tau_D]$
    $\tau_D \leftarrow \tau_D + \Delta\tau_D$
    Sample $y \in S \setminus \{x\}$ from the probability distribution $\frac{\langle y|G^{H_S,\phi_S}|x\rangle}{|\langle x|G^{H_S,\phi_S}|x\rangle|}$
    $x \leftarrow y$
    $\kappa \leftarrow \kappa + 1$
  end if
end while

$\tau \leftarrow \tau + \Delta\tau$
Let $\Delta\tau_D$ be the rounded value of $\Delta\tau$, which is the largest value in the set $\{k\delta\}_{k=0,\ldots,K}$ that does not exceed $\Delta\tau$.
Set $\xi_D(s) = x$ for all $s \in (\tau_D, \tau_D + \Delta\tau_D]$
$\tau_D \leftarrow \tau_D + \Delta\tau_D$

We similarly modify lines 18 and 19.

Algorithm 6 either returns "Reject" or loops forever. By Claim 104 and the construction of Algorithm 6, we know that

- the random variables (τ, ξ) evolves in the same way as (τ, ξ) in Algorithm 5,
- the random variables (τ_D, ξ_D) evolves in the same way as (τ_D, ξ_D) in Algorithm 5D.

Here, we outline the key observations needed to prove Theorem 105. A more formal proof will follow in the next section. Roughly speaking, to establish that Algorithm 5D is a good discrete approximation of Algorithm 5, we analyze the differences between the random variables (τ_D, ξ_D) and (τ, ξ) in Algorithm 6. Specifically, we observe that:

- The sampling of $\Delta\tau$ or $\Delta\tau_D$ is independent of the other steps and can therefore be analyzed separately.
- By construction, $|\Delta\tau - \Delta\tau_D| \leq \delta$ unless $\Delta\tau \geq K\delta$, which is unlikely to happen. When we further set an upper bound $t = poly(n)$ and terminate Algorithm 6 once $\tau > t$, the accumulated error between τ_D and τ is most likely within $\delta \times$ (number of transitions in ξ).
- Note that δ is exponentially small. As long as the number of transitions is $poly(n)$, we ensure that $|\tau - \tau_D|$ is exponentially small. In the Yes instance, Lemma 96 guarantees a non-trivial probability that the number of transitions is $poly(n)$. In the NO instance, the additional check "If $\kappa \geq M$ then Return Reject" in Algorithm 5 and Algorithm 5D ensures that any adversary who successfully cheats can only have $M = poly(n)$ transitions.

Proof of Theorem 105

The formal proof of Theorem 105 is as follows. For Algorithm 6, let $\kappa(x_{in}, \tau_C)$ be the random variable which denotes the number of transitions in $\xi(s)$ for $s \in [0, \tau_C]$. Similarly, let $\kappa_D(x_{in}, \tau_D)$ denote the number of transitions in $\xi_D(s)$ for $s \in [0, \tau_D]$.

To simplify the notation, we use t_C, t_D and M to represent the values specified in Algorithm 5 and Algorithm 5D:

$$t_C := 8(n + p'(n))/\epsilon, \tag{4.34}$$

$$t_D := t_C - 2^{-n}, \tag{4.35}$$

$$M = 2^k m n^3 t_C \|H\|. \tag{4.36}$$

Note that the δ and K from Eqs. (4.32)(4.33) satisfy

$$M\delta \leq 2^{-n},$$
$$M \cdot exp(-\lambda K\delta) \leq 2^{-n}, \text{ for } \lambda \geq 2^{-p'(n)}.$$

Proof:[of Completeness of Theorem 105] Compared to Algorithm 5, its discretized version (Algorithm 5D) only modifies the sampling of the waiting time, replacing $\Delta\tau$ from a continuous process with its discrete approximation $\Delta\tau_D$. The sampling of $\Delta\tau$ or $\Delta\tau_D$ is independent of the other steps. Thus the Completeness proof of Theorem 100 also works for Theorem 105, except that we need to bound

$$Pr(\kappa_D(x_{in}, t_D) \leq M) \text{ instead of } Pr(\kappa(x_{in}, t_C) \leq M).$$

According to the randomness in Algorithm 6 we define the following two events. $Event_1$ is described w.r.t. the variables in the discrete process, and $Event_2$ is described w.r.t. the variables in the continuous process:

$$\begin{aligned} Event_1 &:= \{\kappa_D(x_{in}, t_D) \leq M\}, \\ Event_2 &:= \{\kappa(x_{in}, t_C) \leq M, \text{and for all the} \\ &\qquad \text{transitions in } \xi(s) \text{ for } s \in [0, t_C], \\ &\qquad \text{none of the } \Delta\tau \text{ is greater than } K\delta. \\ &\qquad (\text{Thus } |\Delta\tau - \Delta\tau_D| \leq \delta)\}. \end{aligned}$$

Notice that $Event_2$ implies $Event_1$ since $t_D \leq t_C - M\delta$. Thus we have

$$Pr\left(\kappa_D(x_{in}, t_D) \leq M\right) \geq Pr(Event_2). \tag{4.37}$$

To estimate $Pr(Event_2)$, firstly note that the probability that a particular $\Delta\tau$ exceeds $K\delta$ is negligible. More specifically, from line 21 in Algorithm 6, $\Delta\tau$ is sampled from the exponential distribution with parameter $\lambda_x := |\langle x|G^{H_S,\phi_S}|x\rangle|$. By line 17 we have $\lambda_x \neq 0$, thus by Remark 8 in Appendix 4.11 we have

$$\lambda_x \geq 2^{-p'(n)}.$$

Thus w.r.t. a particular x, we have

$$Pr(\Delta\tau \geq K\delta) = \exp(-\lambda_x K\delta) \leq 2^{-n}/M. \tag{4.38}$$

Thus begin with Eq. (4.37) and apply a union bound, we conclude that

$$\begin{aligned} & Pr\left(\kappa_D(x_{in}, t_D) \leq M\right) \\ & \geq Pr(Event_2) \\ & \geq Pr(\kappa(x_{in}, t_C) \leq M) - M \cdot 2^{-n}/M \\ & \geq 1/2 - 2^{-n}, \end{aligned} \tag{4.39}$$

where $Pr(\kappa(x_{in}, t_C) \leq M) \geq 1/2$ is from the completeness proof for Theorem 100.
■

Proof:[of Soundness of Theorem 105] Recall that Algorithm 6 either returns "Reject" or loops forever. We define τ_{rej} and $\tau_{rej,D}$ as the random variables which denote the value of τ and τ_D respectively, at the moment Algorithm 6 returns "Reject".

Define two events w.r.t. the variables in the discrete process:

$$\begin{aligned} Event_3 :=& \{\tau_{rej,D} \geq t_D, \text{ and } \kappa_D(x_{in}, t_D) \leq M\}, \\ Event_4 :=& \{\tau_{rej,D} \geq t_D, \text{ and } \kappa_D(x_{in}, t_D) \leq M \\ & \quad \text{and for all the transitions of } \xi_D(s) \text{ for} \\ & \quad s \in [0, t_D], \text{ none of the } \Delta\tau \text{ exceeds } K\delta. \\ & \quad (\text{Thus } |\Delta\tau - \Delta\tau_D| \leq \delta)\}. \end{aligned}$$

By the construction of Algorithm 6 we have

$$Pr(\text{Algorithm 5D Accept}) = Pr(Event_3). \tag{4.40}$$

Since the probability that a particular $\Delta\tau$ exceeds $K\delta$ is negligible, similar to the above Completeness proof of Theorem 105, using a union bound, we have

$$\begin{aligned} Pr(Event_3) &\leq Pr(Event_4) + M \cdot 2^{-n}/M \\ &= Pr(Event_4) + 2^{-n}. \end{aligned} \tag{4.41}$$

Moreover, $Event_4$ implies the event $\{\tau_{rej} \geq t_D - M\delta\}$, which is described w.r.t. the variables in the continuous process. Thus in the No instance,

$$\begin{aligned} &Pr(\text{Algorithm 5D Accept}) \\ \leq &Pr(Event_4) + 2^{-n} \qquad (4.42) \\ \leq &Pr(\tau_{rej} \geq t_D - M\delta) + 2^{-n} \\ \leq &Pr(\tau_{rej} \geq t_C - 2 \cdot 2^{-n}) + 2^{-n} \\ = &Pr(\text{Algorithm } 5^* \text{ Accept when} \qquad (4.43) \\ &t \text{ is set to } t_C - 2 \cdot 2^{-n})) + 2^{-n}. \end{aligned}$$

Recall that Algorithm 5^* is defined in Section 4.4, which is Algorithm 5 without the check "if $\kappa \geq M$, then Return Reject". Using the same analysis as in Claim 103, in particular Eqs. (4.29)(4.30)(4.31), we have that

$$\begin{aligned} &Pr(\text{Algorithm } 5^* \text{ Accept when } t \text{ is set to } t_C - 2 \cdot 2^{-n})) \\ &\leq 2^n \cdot 2^{3p(n)} \cdot (1 - \epsilon/2)^{t_C - 2 \cdot 2^{-n}} \\ &\leq 2 \cdot 2^{-n}. \end{aligned}$$

Thus we complete the proof. ∎

4.5 Appendix:Relationship to matrix verification

Our techniques for proving Theorem 83 work for a slightly general setting, that is, we can generalize from local Hamiltonians to sparse matrices with small norm. The detailed setting is as follows.

Consider a matrix $H \in \mathbb{R}^{2^n \times 2^n}$, and a vector $|\phi\rangle \in \mathbb{R}^{2^n}$. H is $poly(n)$-sparse, that is, every row and every column only have $poly(n)$ non-zero entries. Besides, H has a **small norm**, that is, $\|H\| \leq poly(n)$. Suppose we have query access to H and $|\phi\rangle$. That is,

(1) For any $x, y \in \{0,1\}^n$, there is a $poly(n)$ time algorithm which returns $\langle x|H|y\rangle$.

(2) For any row index x, there is a polynomial time algorithm which outputs the column indices of the non-zero entries in the row, that is, y s.t. $\langle x|H|y\rangle \neq 0$. There is a similar algorithm for listing all non-zero entries of a chosen column.

(3) Given x, there is a $poly(n)$-time algorithm which outputs $\langle x|\phi\rangle$ up to a common factor, that is, $c_\phi(n) \cdot \langle x|\phi\rangle$ for some unknown $c_\phi(n)$. Note that this allows one to efficiently compute the ratio $\frac{\langle x|\phi\rangle}{\langle y|\phi\rangle}$.

The problem is to design an algorithm to distinguish the following two cases, with as few queries as possible:

- Yes instance: $\langle\phi|H|\phi\rangle = 0$, $|\phi\rangle$ is promised to be the ground state of H.
- No instance: $\lambda(H) \geq 1/poly(n)$. $|\phi\rangle$ can be arbitrary.

Here $\lambda(H)$ is the ground energy of H.

Theorem 106 *Under the assumptions in Remark 6 and Remark 7 in Appendix 4.11. Given query access to $(H, |\phi\rangle)$, where $(H, |\phi\rangle)$ is promised to satisfy either the Yes or No instance. There exists an algorithm $\mathcal{A}(x)$ which takes an input $x \in \{0,1\}^n$, runs in $poly(n)$ time, and only makes $poly(n)$ queries to $H, |\phi\rangle$ such that*

- *If $(H, |\phi\rangle)$ is a Yes instance, there exists $x \in \{0,1\}^n$ such that the algorithm accepts with probability $\geq 1/2$.*
- *If it is a No instance, $\forall x$, the algorithm accepts with exponentially small probability.*

Roughly speaking $\mathcal{A}(x)$ is a random walk over $\{0,1\}^n$ which starts from the state x. We omit the proof since it is the same as the proof of Theorem 83.

4.6 Appendix:MA-hardness

In this section, we briefly explain how the proof in Section 4 of [Bra+08] implies LHP with succinct ground state is **MA-hard**. We also check the Hamiltonian in the reduction satisfies Remark 6 in Appendix 4.11. Let $p(n)$ be a sufficiently large polynomial.

Consider a problem L in **MA**, for any instance x, [Bra+08] shows that one can view the **BPP** verification circuit as a quantum circuit. Specifically,

Definition 107 ($\mathbf{MA}_{q1}$) *A promise problem $L_{yes}, L_{no} \subseteq \Sigma^*$ belongs to $\boldsymbol{MA}_{q1}$ if there exists a polynomial p and a poly-size classical reversible circuit V_x that takes input in $(\mathbb{C}^2)^{\otimes p(|x|)}$ and is followed by a single qubit measurement, such that*

$$x \in L_{yes} \Rightarrow \exists\, |\xi\rangle\,, P\left[V_x\left(|00...0\rangle, |+\rangle^{\otimes r}\,, |\xi\rangle\right) = 1\right] = 1, \tag{4.44}$$

$$x \in L_{no} \Rightarrow \forall\, |\xi\rangle\,, P\left[V_x\left(|00...0\rangle, |+\rangle^{\otimes r}\,, |\xi\rangle\right) = 1\right] \leq 1/2. \tag{4.45}$$

Note that w.l.o.g. we can assume that r is even, since adding one $|+\rangle$ state which is independent of other parts of the circuit does not influence the accepting probability.

Lemma 108 $\boldsymbol{MA} = \boldsymbol{MA}_{q1}$. *If $x \in L_{yes}$, the $|\xi\rangle$ in Eq. (4.44) can always be chosen to be a computational basis, denoted as $|w\rangle$.*

Proof: Compared to the definition of $\mathbf{MA}_q$ in Definition 7 of [Bra+08], $\mathbf{MA}_{q1}$ in Definition 107 requires perfect completeness. It is well known that the complexity class **MA** remains unchanged if we require perfect completeness. Based on this fact, one can check that the Lemma 2 in [Bra+08] for proving $\mathbf{MA} = \mathbf{MA}_q$ also works for proving $\mathbf{MA} = \mathbf{MA}_{q1}$. ∎

Begin with the **MA-complete** problem $\mathbf{MA}_{q1}$ with $L_{yes} \cup L_{no}$, suppose V_x is consisting of T classical reversible gates, denoted as $V_x := R_T...R_1$. As shown in Section 4 of [Bra+08], one can use Kitaev's circuit to Hamiltonian reduction [KSV02] to get a Hamiltonian H_x, where $x \in L_{yes}$ or $x \in L_{no}$ will be mapped to H_x with $\lambda(H_x) \leq a$ or $\lambda(H_x) \geq b$ where $b - a \geq 1/poly(n)$. Besides, one can check that each entry of the Hamiltonian is of form $\frac{N_1}{N_2}$, and greater than $1/2^{p(n)}$.

Then we check that the above reduction from x to H_x satisfies the promise of LHP with succinct ground state. That is, in the Yes case, there always *exists* a succinct ground state. Since V_x has perfect completeness, one can check that the history state below is a ground state of H_x,

$$|\psi_{hist}\rangle = \frac{1}{T+1}\sum_{t=0}^{T}\left(R_t...R_1|00...0\rangle\,|+\rangle^{\otimes r}\,|w\rangle\right) \otimes |t\rangle\,,$$

where $|w\rangle$ is the computational basis defined in Lemma 108, and the clock $|t\rangle$ uses the unary encoding, that is,

$$|t\rangle = |\underbrace{0..0}_{T-t \text{ zeros},}\ \underbrace{1..1}_{t \text{ ones.}}\rangle.$$

Given a computational basis x, denote the T bits that correspond to the clock register as x_c, and the other bits as x_o. One can check that if x_c is not of the form of unary encoding, then $\langle x|\psi_{hist}\rangle = 0$. If $x_c = |t\rangle$, then

$$\begin{aligned}\langle x|\psi_{hist}\rangle &= \frac{1}{T+1}\langle x_o|R_t...R_1|00...0\rangle\,|+\rangle^{\otimes r}\,|w\rangle \\ &= \frac{1}{T+1}\frac{1}{2^{r/2}}\sum_{y\in\{0,1\}^r}\langle (R_t...R_1)^\dagger x_o|00...0,y,w\rangle. \end{aligned} \tag{4.46}$$

Since $\{R_t\}_t$ are classical reversible gates, one can easily use the poly-size classical circuit to compute $z := (R_t...R_1)^\dagger x_o$. If in z the bits correspond to the ancillas $|00...0\rangle$ and witness w are $|00...0\rangle$ and $|w\rangle$, then Eq. (4.46) is the sum of one $\frac{1}{T+1}\frac{1}{2^{r/2}}$ plus $2^r - 1$ zeros, thus

$$\langle x|\psi_{hist}\rangle = \frac{1}{T+1}\frac{1}{2^{r/2}}. \tag{4.47}$$

Otherwise $\langle x|\psi_{hist}\rangle$ is a sum of 2^r zeros thus equal to 0.

Since r is even by Definition 107, one can check that $\langle x|\psi_{hist}\rangle$ is of form $\frac{N_1}{N_2}$, and greater than

$$\frac{1}{poly(n)2^{r/2}} \geq 2^{-p(n)}.$$

In summary, $\langle x|\psi_{hist}\rangle$ can be computed by a $poly(n)$-size classical circuit thus is succinct, and satisfies Remark 6 in Appendix 4.11. Thus LHP with succinct ground state is **MA-hard**.

4.7 Appendix:Proof of two facts

Proof:[of Fact 90] For any $x \in [-1,1]$, consider its Taylor series with Lagrange remainder term, we have

$$e^{-x} = 1 - x + R_1(x), \tag{4.48}$$

where $R_1(x) = \frac{e^{-\eta}}{2!}x^2$ for $\eta \in [-1,1]$. Thus

$$|e^{-x} - (1-x)| \leq \frac{e}{2}x^2 \leq 2x^2. \tag{4.49}$$

■

Proof:[of Fact 91] W.l.o.g., we write M as

$$M = \begin{bmatrix} N & E \\ E^\dagger & F \end{bmatrix}, \tag{4.50}$$

where $E \in \mathbb{R}^{|S'|\times(|S|-|S'|)}$, $F \in \mathbb{R}^{(|S|-|S'|)\times(|S|-|S'|)}$.

Since M, N are Hermitians, they can be diagonalized by orthogonal basis. In particular, for M, for any normalized state $|\psi\rangle$, we have

$$\lambda(M) \leq \langle\psi|M|\psi\rangle \leq \lambda_{max}(M). \tag{4.51}$$

Let $|\eta\rangle \in \mathbb{R}^{|S'|}$ be a normalized eigenvector of N with eigenvalue α. Let $|\eta 0\rangle \in \mathbb{R}^{|S|}$ to be the state that extending $|\eta\rangle$ to $\mathbb{R}^{|S|}$ by adding 0 values to entries in $S\backslash S'$. $|\eta 0\rangle$ is normalized by definition. One can verify that

$$\langle\eta 0|M|\eta 0\rangle = \langle\eta|N|\eta\rangle \tag{4.52}$$

$$= \alpha. \tag{4.53}$$

Together with inequality (4.51), we have

$$\lambda(M) \leq \alpha \leq \lambda_{max}(M). \tag{4.54}$$

Thus

$$\lambda(M) \leq \lambda(N) \leq \lambda_{max}(N) \leq \lambda_{max}(M). \tag{4.55}$$

■

4.8 Appendix:Properties of $F^{H,\phi}$.

Proof:[of Lemma 94 from [Ten+95] [Bra+23b]] Note that H is real-valued and Hermitian, thus H is symmetric. (1) is true by definition of stoquastic. For (2), one

can verify that for any x,

$$\begin{aligned}
\langle x|F^{H,\phi}|\phi\rangle &= \sum_y \langle x|F^{H,\phi}|y\rangle\langle y|\phi\rangle \\
&= \langle x|F^{H,\phi}|x\rangle\langle x|\phi\rangle + \sum_{y:(x,y)\in S^-} \langle x|H|y\rangle\langle y|\phi\rangle \\
&= \langle x|H|x\rangle\langle x|\phi\rangle + \sum_{z:(x,z)\in S^+} \langle x|H|z\rangle\langle z|\phi\rangle \\
&+ \sum_{y:(x,y)\in S^-} \langle x|H|y\rangle\langle y|\phi\rangle \\
&= \sum_y \langle x|H|y\rangle\langle y|\phi\rangle \\
&= \langle x|H|\phi\rangle.
\end{aligned}$$

Thus $F^{H,\phi}|\phi\rangle = H|\phi\rangle$. For (3), by (1) $F^{H,\phi}$ is Hermitian and thus diagonalizable, thus $\lambda(F^{H,\phi})$ is well-defined. Consider that for any complex-valued state $|\xi\rangle$, one has

$$\begin{aligned}
&\langle \xi|F^{H,\phi} - H|\xi\rangle \\
&= \sum_{x,y} \langle\xi|x\rangle\langle x|F^{H,\phi} - H|y\rangle\langle y|\xi\rangle \\
&= \sum_{(x,y)\in S^+} \langle\xi|x\rangle(-\langle x|H|y\rangle)\langle y|\xi\rangle + \sum_x \langle\xi|x\rangle\langle x|F^{H,\phi} - H|x\rangle\langle x|\xi\rangle \\
&= \sum_{(x,y)\in S^+} \langle\xi|x\rangle(-\langle x|H|y\rangle)\langle y|\xi\rangle + \sum_x \langle\xi|x\rangle \sum_{y:(x,y)\in S^+} \langle x|H|y\rangle \frac{\langle y|\phi\rangle}{\langle x|\phi\rangle}\langle x|\xi\rangle \\
&= \sum_{(x,y)\in S^+} \langle x|H|y\rangle \left[\frac{\langle y|\phi\rangle\langle\xi|x\rangle\langle x|\xi\rangle}{\langle x|\phi\rangle} - \langle\xi|x\rangle\langle y|\xi\rangle\right]. \qquad (4.56)
\end{aligned}$$

For any x, y, define

$$s(x,y) := sign(\langle x|H|y\rangle).$$

Note that H is symmetric[17], and thus

$$s(x,y) = s(y,x), \langle x|H|y\rangle = \langle y|H|x\rangle.$$

[17] Since H is real valued and Hermitian.

Using the definition of S^+, one gets Eq. (4.56) equals to

$$
\begin{aligned}
&= \sum_{(x,y)\in S^+} |\langle x|H|y\rangle| \left[\frac{|\langle y|\phi\rangle|}{|\langle x|\phi\rangle|} \langle \xi|x\rangle\langle x|\xi\rangle - s(x,y)\langle \xi|x\rangle\langle y|\xi\rangle \right] \\
&= \sum_{(x,y)\in S^+} \frac{1}{2} |\langle x|H|y\rangle| \left| \sqrt{\frac{|\langle y|\phi\rangle|}{|\langle x|\phi\rangle|}} \langle \xi|x\rangle - s(x,y) \sqrt{\frac{|\langle x|\phi\rangle|}{|\langle y|\phi\rangle|}} \langle \xi|y\rangle \right|^2 \\
&\geq 0. \qquad (4.57)
\end{aligned}
$$

In other words, H can always achieve smaller energy than $F^{H,\phi}$, thus

$$\lambda(F^{H,\phi}) \geq \lambda(H). \qquad (4.58)$$

Notice that when $|\xi\rangle = |\phi\rangle$, Eq. (4.57) equals to 0. Thus if further $|\phi\rangle$ is the ground state of H, then $F^{H,\phi}$ can achieve the energy $\lambda(H)$ w.r.t. $|\phi\rangle$, then $|\phi\rangle$ is the ground state of $F^{H,\phi}$ due to Eq. (4.58). ■

4.9 Appendix:Properties of the CTMC

Proof:[of Corollary 95.] The proof comes from [Bra+23b]. For (1), it suffices to notice that

$$\widetilde{G}^{H,\phi} = Diag(\phi)^{-1}(\lambda(F^{H,\phi})I - F^{H,\phi})Diag(\phi),$$

Note that $Diag(\phi)^{-1}$ is well-defined since ϕ is regularized and real-valued.

For (2), one can verify $\langle y|\widetilde{G}^{H,\phi}|x\rangle \geq 0$ for $y \neq x$ by definition. If $|\phi\rangle$ is a ground state of H,

$$
\begin{aligned}
\sum_y \langle y|\widetilde{G}^{H,\phi}|x\rangle &= \lambda(F^{H,\phi}) - \sum_y \langle y|F^{H,\phi}|x\rangle \frac{\langle y|\phi\rangle}{\langle x|\phi\rangle} \\
&= \lambda(F^{H,\phi}) - \langle \phi|F^{H,\phi}|x\rangle \frac{1}{\langle x|\phi\rangle} \\
&= \lambda(F^{H,\phi}) - \lambda(F^{H,\phi})\langle \phi|x\rangle \frac{1}{\langle x|\phi\rangle} \qquad (4.59) \\
&= 0,
\end{aligned}
$$

where Eq. (4.59) is from Lemma 94 (3) and the fact that F is Hermitian, $\lambda(F^{H,\phi})$ must be real-valued as eigenvalues of Hermitian $F^{H,\phi}$.

For (3), when $|\phi\rangle$ is ground state of H, by (2) we know $\widetilde{G}^{H,\phi}$ is a legal generator. One can verify that for any y,

$$\begin{aligned}
&\sum_x \langle y|\widetilde{G}^{H,\phi}|x\rangle\pi(x) \\
&= \lambda(F^{H,\phi})\pi(y) - \sum_x \langle y|F^{H,\phi}|x\rangle \frac{\langle y|\phi\rangle\langle x|\phi\rangle}{c} \\
&= \lambda(F^{H,\phi})\pi(y) - \langle y|F^{H,\phi}|\phi\rangle \frac{\langle y|\phi\rangle}{c} \\
&= \lambda(F^{H,\phi})\pi(y) - \lambda(F^{H,\phi})\langle y|\phi\rangle \frac{\langle y|\phi\rangle}{c} \\
&= 0.
\end{aligned}$$

■

Proof:[of Lemma 96] This proof is modified from [Bra+23b]. First note that since $|\phi\rangle$ is the ground state of H, by Corollary 95 (2), $\widetilde{G}^{H,\phi}$ is a legal generator of a CTMC, thus Algorithm 4 w.r.t. $\widetilde{G}^{H,\phi}$ is well-defined.

Define $c = \| |\phi\rangle \|^2$ and $\pi(x) = |\langle x|\phi\rangle|^2/c$. Let h be an infinitesimal value.

For any random process generating a random variable $\xi : [0,t] \to S$, define $T(\tau, x, h)[\xi]$ as the number of transitions occurring within the time interval $[\tau, \tau+h]$, conditioned on $\xi(\tau) = x$. Let $I_{bad}[\xi]$ be the indicator function that at least 2 transitions happen in any of the time intervals

$$[0,h], [h,2h], ..., [t-h,t] \text{ for } t = poly(n).$$

Here for simplicity we assume that t/h is an integer.

Let $\rho(\xi)$ be the distribution of $\xi : [0,t] \to S$ generated by running Algorithm 4 w.r.t. $(\widetilde{G}^{H,\phi}, x_{in}, t)$, where x_{in} is sampled from distribution π. By Lemma 92 we know

$$Pr_{\xi\sim\rho(\xi)}(T(\tau,x,h)[\xi] \geq 2) = O(h^2), \forall x, \tau.$$

By a union bound we know

$$\begin{aligned}
&Pr_{\xi\sim\rho(\xi)}(I_{bad}[\xi] = 1) \\
&\leq \sum_{j=0}^{t/h} \sum_x Pr_{\xi\sim\rho(\xi)}(\xi(jh) = x) Pr_{\xi\sim\rho(\xi)}(T(jh,x,h)[\xi] \geq 2) \\
&= \frac{t}{h} O(h^2) \\
&= O(h).
\end{aligned} \tag{4.60}$$

By Corollary 95 (3), we know that π is a stationary distribution of $\widetilde{G}^{H,\phi}$. Thus for $\xi \sim \rho(\xi)$, at any time $s \in [0,t]$, the distribution of $\xi(s)$ is given by π, which we denote as $\pi_s = \pi$. For $s \leq t$, let $\pi_{good,s}, \pi_{bad,s}$ be the distribution of $\xi(s)$ conditioned on "good", "bad", that is,

$$\begin{aligned} \pi_{good,s}(x) &= Pr_{\xi\sim\rho(\xi)}(\xi(s) = x | I_{bad}(\xi) = 0), \\ \pi_{bad,s}(x) &= Pr_{\xi\sim\rho(\xi)}(\xi(s) = x | I_{bad}(\xi) = 1), \end{aligned} \tag{4.61}$$

we have

$$\begin{aligned} \pi_s(x) =& Pr_{\xi\sim\rho(\xi)}\left(I_{bad}[\xi] = 0\right) \pi_{good,s}(x) \\ &+ Pr_{\xi\sim\rho(\xi)}\left(I_{bad}[\xi] = 1\right) \pi_{bad,s}(x). \end{aligned}$$

Thus

$$\pi_{good,s}(x) - \pi(x) = O(h). \tag{4.62}$$

In other words, $\pi_{good,s}$ is almost π.

For $s \leq t$, where s is an integer multiple of h, let $T_{good}(s)[\xi]$ be the number of

transitions in time [0,s], conditioned on $I_{bad}[\xi] = 0$. We have that[18]

$$
\begin{aligned}
&E_{\xi\sim\rho(\xi)}[T_{good}(s+h)[\xi] - T_{good}(s)[\xi]] \\
&= \sum_x \pi_{good,s}(x) \cdot 1 \cdot \left(\sum_{y\neq x} \langle y|\widetilde{G}^{H,\phi}|x\rangle h + O(h^2) \right) \qquad (4.65) \\
&= \sum_x \pi(x) \sum_{y:y\neq x} \langle y|\widetilde{G}^{H,\phi}|x\rangle h + O(h^2) \\
&= \sum_x \frac{|\langle x|\phi\rangle|^2}{c} \sum_{y:y\neq x} -\langle y|F^{H,\phi}|x\rangle \frac{\langle y|\phi\rangle}{\langle x|\phi\rangle} h + O(h^2) \\
&= \sum_x \sum_{y:y\neq x} -\langle y|F^{H,\phi}|x\rangle \frac{\langle y|\phi\rangle\langle x|\phi\rangle}{c} h + O(h^2) \\
&= \sum_{(x,y)\in S^-} -\langle x|H|y\rangle\langle y|\phi\rangle\langle x|\phi\rangle \frac{1}{c} h + O(h^2). \qquad (4.66)
\end{aligned}
$$

Thus

$$
\begin{aligned}
&E_{\xi\sim\rho(\xi)}(T_{good}(t)[\xi]) \\
&= \sum_{(x,y)\in S^-} -\langle x|H|y\rangle\langle y|\phi\rangle\langle x|\phi\rangle \frac{1}{c} t + \frac{t}{h} O(h^2).
\end{aligned}
$$

[18] In Eq. (4.65), we use

$$
\begin{aligned}
&Pr_{\xi\sim\rho(\xi)}(T(s,x,h)[\xi] = 1, \xi(s+h) = y|\xi(s) = x, I_{bad}[\xi] = 0) \\
=&Pr_{\xi\sim\rho(\xi)}(T(s,x,h)[\xi] = 1, \xi(s+h) = y|\xi(s) = x, T(s,x,h)[\xi] \leq 1) \qquad (4.63) \\
=&Pr_{\xi\sim\rho(\xi)}(T(s,x,h)[\xi] = 1, \xi(s+h) = y|\xi(s) = x) \\
&\div Pr_{\xi\sim\rho(\xi)}(T(s,x,h)[\xi] \leq 1|\xi(s) = x) \qquad (4.64) \\
=&\left(\langle y|\widetilde{G}^{H,\phi}|x\rangle h + O(h^2)\right) \div \left(1 - O(h^2)\right) \\
=&\langle y|\widetilde{G}^{H,\phi}|x\rangle h + O(h^2).
\end{aligned}
$$

Here Eq. (4.63) comes from the Markov property, that is, the events occur within the time interval $[s, s+h]$ are independent from the events occur outside this time interval. Eq. (4.64) comes from Bayesian rule.

Then

$$
\begin{aligned}
&E_{\xi\sim\rho(\xi)}[T_{good}(t)[\xi]] \\
&\leq \sum_{(x,y)\in S^-} |\langle x|H|y\rangle| \cdot |\langle y|\phi\rangle| \cdot |\langle x|\phi\rangle| \frac{1}{c} t + O(h) \\
&\leq \sum_{(x,y)} |\langle x|H|y\rangle| \cdot |\langle y|\phi\rangle| \cdot |\langle x|\phi\rangle| \frac{1}{c} t + O(h) \\
&\leq \|H\| \frac{1}{c} t \left(\sum_{(x,y):\langle x|H|y\rangle\neq 0} |\langle y|\phi\rangle| \cdot |\langle x|\phi\rangle| \right) + O(h) \\
&\leq \|H\| \frac{1}{c} t \left(\sum_{(x,y):\langle x|H|y\rangle\neq 0} |\langle y|\phi\rangle|^2 \right)^{\frac{1}{2}} \left(\sum_{(x,y):\langle x|H|y\rangle\neq 0} |\langle x|\phi\rangle|^2 \right)^{\frac{1}{2}} \\
&+ O(h) \\
&\leq \|H\| \frac{1}{c} t d \|\phi\|^2 + O(h) \qquad (4.67) \\
&= dt\|H\| + O(h),
\end{aligned}
$$

where Eq. (4.67) comes from the fact that since H is d-sparse, for every y, we know that $|\langle y|\phi\rangle|^2$ appears for at most d times in $\left(\sum_{(x,y):\langle x|H|y\rangle\neq 0} |\langle y|\phi\rangle|^2\right)$.

Let $\kappa_{good}(x,t)[\xi]$ be the number of transitions between $[0,t]$ conditioned on

$$\xi(0) = x, I_{bad}[\xi] = 0.$$

Let $A(x)$ be the distribution of ξ generated by Algorithm 4 w.r.t. $(\widetilde{G}^{H,\phi}, x, t)$. note that

$$
\begin{aligned}
&E_{\xi\sim\rho(\xi)}(T_{good}(t)[\xi]) \\
&= \sum_x \pi(x) E_{\xi\sim A(x)}(\kappa_{good}(x,t)[\xi]). \qquad (4.68)
\end{aligned}
$$

Combine Eqs. (4.67)(4.68), using an average argument, we know that there exists an $\hat{x}_{in}$ with $\pi(\hat{x}_{in}) \neq 0$, that is, $\langle \hat{x}_{in}|\phi\rangle \neq 0$, such that

$$E_{\xi\sim A(\hat{x}_{in})}(\kappa_{good}(\hat{x}_{in},t)[\xi]) \leq dt\|H\| + O(h). \qquad (4.69)$$

By a Markov bound, we know

$$
\begin{aligned}
&Pr_{\xi\sim A(\hat{x}_{in})}\left(\kappa_{good}(\hat{x}_{in},t)[\xi] \geq dn^3 t\|H\|\right) \\
&\leq 1/n^2. \qquad (4.70)
\end{aligned}
$$

Besides, we have that

$$\begin{aligned}
&Pr_{\xi\sim A(\hat{x}_{in})}(\kappa(\hat{x}_{in},t)[\xi]\geq dn^3t\|H\|)\\
&=Pr_{\xi\sim A(\hat{x}_{in})}(I_{bad}[\xi]=1,\kappa(\hat{x}_{in},t)[\xi]\geq dn^3t\|H\|)\\
&+Pr_{\xi\sim A(\hat{x}_{in})}(I_{bad}[\xi]=0,\kappa(\hat{x}_{in},t)[\xi]\geq dn^3t\|H\|).
\end{aligned}$$

Note that $Pr_{\xi\sim A(\hat{x}_{in})}(I_{bad}[\xi]=1)=O(h)$ by a similar argument as Eq. (4.60), we finally conclude that

$$\begin{aligned}
&Pr_{\xi\sim A(\hat{x}_{in})}(\kappa(\hat{x}_{in},t)[\xi]\geq dn^3t\|H\|)\\
\leq&Pr_{\xi\sim A(\hat{x}_{in})}(I_{bad}[\xi]=0,\kappa(\hat{x}_{in},t)[\xi]\geq dn^3t\|H\|)+O(h)\\
=&Pr_{\xi\sim A(\hat{x}_{in})}(I_{bad}[\xi]=0)\times\\
&Pr_{\xi\sim A(\hat{x}_{in})}\Big(\kappa_{good}(\hat{x}_{in},t)[\xi]\geq dn^3t\|H\|\mid I_{bad}[\xi]=0\Big)+O(h)\\
\leq&(1-O(h))1/n^2+O(h)\\
\leq&1/2.
\end{aligned}\tag{4.71}$$

In other words,

$$Pr_{\xi\sim A(\hat{x}_{in})}(\kappa(\hat{x}_{in},t)[\xi]\leq dn^3t\|H\|)\geq 1/2.\tag{4.72}$$

■

4.10 Appendix: Proof of Claim 102

Proof:[of Claim 102] First we define more notations for the proof. Conditioned on $\tau_{end}>s,\xi(s)=x$, let t_x be the time that ξ last arrives x before time s, that is,

$$t_x:=\max\{\eta\leq s:\xi(\eta)\neq x\}.\tag{4.73}$$

Let $\Delta\tau_x$ be the waiting time sampled by line 8 when ξ reaches x at time t_x. Note that $c_1=t_x+\Delta\tau_x$. If $y\in S_{good}$ is the state that ξ visit next, that is, $\xi(c_1)$, similarly define $\Delta\tau_y$ be the waiting time by line 8 when ξ reaches y at time c_1.

To simplify the notations, we denote

$$\begin{aligned}
u_x&:=|\langle x|G^{H_S,\phi_S}|x\rangle|,\\
P_{xy}&:=\langle y|G^{H_S,\phi_S}|x\rangle/u_x.
\end{aligned}$$

To ease notations, we abbreviate the probability density function $p(t_x = s_x|\tau_{end} > s, \xi(s) = x)\, ds_x$ as $p(t_x = s_x)ds_x$. Note that by definition

$$\int_{s_x \le s} p(t_x = s_x)\, ds_x = 1.$$

To complete the proof, notice that

- For Eq. (4.24) we can perform calculation in Appendix 4.12 Ⓐ.
- For Eq. (4.25), when $y \notin S_{good}$, we can perform calculation in Appendix 4.12 Ⓑ.
- For Eq. (4.26), when $y \in S_{good}$, we can perform calculation in Appendix 4.12 Ⓒ.
- Eq. (4.27) is one minus of the probability of Eqs. (4.24)(4.25)(4.26). Since $\sum_y \langle y|G^{H_s,\phi_s}|x\rangle = 0$, and thus we know Eq. (4.27) holds.

■

4.11 Appendix: Remarks on precision

Remark 6 (How we represent values)

We say $x \in \mathbb{C}$ is represented by $p(n)$-bits, if x is of the form $\frac{N_1}{N_2} + \frac{N_3}{N_4}i$, where $\forall i$, N_i is an integer and $|N_i| \le 2^{p(n)}$, thus can be represented by $p(n)$ binary bits.

Note that LHP with succinct ground state is a promise problem, we implicitly assume that there is a sufficiently large polynomial $p(n) = poly(n)$, such that every value in Definition 88, that is, $\langle x|H|y\rangle, a, b, m, C_\psi(x)$ can be represented by $p(n)$-bits. Note that those assumptions implicitly imply

(1) $C_\psi(x)$ can be computed **exactly**. Thus the ratio of the amplitudes, that is, $\frac{\langle x|\psi\rangle}{\langle y|\psi\rangle}$, can be computed exactly for y where $C_\psi(y) \neq 0$.

(2) If $C_\psi(x) \neq 0$, then $|C_\psi(x)| \ge 1/2^{p(n)}$. Similarly for $\langle x|H|y\rangle$.

(3) $\lambda(H)$ can be represented exactly by $poly(n)$ bits, since $\langle x|H|y\rangle, C_\psi(x)$ can be represented by $p(n)$ bits, and $\lambda(H) = \frac{\sum_y \langle x|H|y\rangle C_\psi(y)}{C_\psi(x)}$ for some x s.t. $C_\psi(x) \neq 0$.

For $S \subseteq \{0,1\}^n$, we say a matrix $G \in \mathbb{C}^{|S|\times|S|}$ can be represented by $p(n)$-bits, if all entries $\langle x|G|y\rangle$ can be represented by $p(n)$-bits.

One can check that if G can be represented by $p(n)/2$-bits, then there exists $p'(n) = poly(n)$ such that[19]

$$amax(G) \leq 2^{p'(n)},\ \|G\| \leq 2^{p'(n)+2n}.$$

Besides, if $\langle x|G|y\rangle \neq 0$, then $\langle x|G|y\rangle \geq 1/2^{p'(n)}$.

Remark 7 (Assumptions for sampling)

Let $M = poly(n)$. Set parameters

$$\delta := 2^{-2n}/M$$
$$K := \lceil 2^{2n} M\ (n \ln 2 + \ln M)) \rceil.$$

Note that for any sufficiently large $q = poly(n)$, there is a $poly(n)$-time algorithm whose output distribution approximates the truncated discretized exponential distribution $\mathcal{D}_{K,\delta,\lambda}$ within total variation distance[20] 2^{-q}. Specifically, the algorithm is sampling q^2 random bits $s_1, ..., s_q \in \{0,1\}$. Let

$$\eta = \sum_{j=1}^{q^2} s_j 2^{j-1}.$$

- If $\eta \in (\exp(-\lambda(k+1)\delta), \exp(-\lambda k\delta)]$ for $k \leq K$, output $w = k\delta$.
- If $\eta \leq \exp(-\lambda K\delta)$, output $w = K\delta$.

The total variation distance between the output distribution and $\mathcal{D}_{K,\delta,\lambda}$ is $O(\frac{1}{2^{q^2}}K) = O(2^{-q})$ for sufficiently large $q = poly(n)$.

To ease analysis, in this manuscript we assume that for the parameters defined above, we can use $poly(n)$-time to sample the discretized exponential distribution $\mathcal{D}_{K,\delta,\lambda}$ **exactly**.

[19]Given any matrix $M \in \mathbb{R}^{|S|\times|S|}$, for any normalized vector $|\phi\rangle \in \mathbb{R}^{|S|}$, since $|\langle x|\phi\rangle| \leq 1, \forall x$, we have $\langle\phi|M^\dagger M|\phi\rangle = \sum_{x,y,z}\langle\phi|x\rangle\langle x|M^\dagger|y\rangle\langle y|M|z\rangle\langle z|\phi\rangle \leq amax(M)^2 2^{3n}$. Thus $\|M\| \leq \sqrt{amax(M)^2 2^{3n}} \leq amax(M)2^{2n}$.

[20]Given two probability distribution p, q over a discrete set Ω, the total variation distance $d_{TV}(p,q)$ between p and q is defined as $d_{TV}(p,q) = \sum_{x\in\Omega}\frac{1}{2}|p(x) - q(x)|$, where $p(x)$ is the probability of $p = x$ and similarly for $q(x)$.

Remark 8 (More on how we represent values)

As in Remark 6, in the following sections, we assume values in $H, |\phi\rangle$ can be represented by $p(n)$-bits. Note that by Remark 6, if $\langle x_{in}|\phi\rangle \neq 0$, then

$$\left|\frac{\langle z|\phi\rangle}{\langle x_{in}|\phi\rangle}\right| = \left|\frac{C_\phi(z)}{C_\phi(x_{in})}\right| \leq \frac{2^{p(n)} + 2^{p(n)}}{1/2^{p(n)}} \leq 2^{3p(n)}. \tag{4.74}$$

We represent $\langle x|F^{H_S,\phi_S}|y\rangle$,$\langle x|G^{H_S,\phi_S}|y\rangle$ in a similar way as in Remark 6. Note that H_S is $2^k m$-sparse where k is a constant and $m = poly(n)$, and thus there exists another polynomial $p'(n)$, such that $\langle x|F^{H_S,\phi_S}|y\rangle$, $\langle x|G^{H_S,\phi_S}|y\rangle$ can be represented by $p'(n)/2$ bits, where

$$p'(n)/2 = O(\log[(2^{p(n)*2^k m}]) = O(poly(n)).$$

In particular, we have

(1) The followings holds:

$$amax(G^{H_S,\phi_S}) \leq 2^{p'(n)},$$
$$\|G^{H_S,\phi_S}\|, \|F^{H_S,\phi_S}\| \leq 2^{p'(n)+2n}.$$

(2) If $\langle x|G^{H_S,\phi_S}|y\rangle \neq 0$, then $\langle x|G^{H_S,\phi_S}|y\rangle \geq 1/2^{p'(n)}$.

We always assume $p(n)$, $p'(n)$ are sufficiently large polynomials, and $p(n) \ll p'(n)$.

4.12 Appendix: Calculation for Equations

The details are as follows:

Ⓐ Calculation for Claim 102 Eq. (4.24).

$$Pr(c_1 \geq s + h|\tau_{end} > s, \xi(s) = x) \tag{4.75}$$
$$= \int_{s_x \leq s} p(t_x = s_x) Pr(\Delta\tau_x \geq s + h - s_x|\Delta\tau_x \geq s - s_x) ds_x \tag{4.76}$$
$$= \int_{s_x \leq s} p(t_x = s_x) \exp(-u_x h) ds_x \tag{4.77}$$
$$= 1 - u_x h + O(h^2) \tag{4.78}$$
$$= 1 - |\langle x|G^{H_S,\phi_S}|x\rangle| h + O(h^2). \tag{4.79}$$

Ⓑ Calculation for Claim 102 Eq. (4.25).

$$Pr(c_1 \leq s+h, \xi(c_1) = y | \tau_{end} > s, \xi(s) = x) \tag{4.80}$$

$$= Pr(\Delta\tau_x \leq s+h-s_x, \xi(s_x+\Delta\tau_x) = y | \tau_{end} > s, \xi(s) = x) \tag{4.81}$$

$$= \int_{s_x \leq s} p(t_x = s_x) \int_{s-s_x \leq f \leq s+h-s_x} p(\Delta\tau_x = f | \Delta\tau_x \geq s-s_x) P_{xy} df\, ds_x \tag{4.82}$$

$$= P_{xy} \int_{s_x \leq s} p(t_x = s_x) \int_{s-s_x \leq f \leq s+h-s_x} u_x \exp\left[-u_x(f-s+s_x)\right] df\, ds_x \tag{4.83}$$

$$= P_{xy} \int_{s_x \leq s} p(t_x = s_x) \left[1 - \exp(-u_x h)\right] ds_x \tag{4.84}$$

$$= p_{xy} u_x h + O(h^2) \tag{4.85}$$

$$= \langle y | G^{H_S, \phi_S} | x \rangle h + O(h^2). \tag{4.86}$$

© Calculation for Claim 102 Eq. (4.26).

$$Pr(c_1 \le s+h, \xi(c_1) = y, c_2 \ge s+h | \tau_{end} > s, \xi(s) = x) \tag{4.87}$$

$$= Pr(\Delta\tau_x \le s+h-s_x, \xi(s_x+\Delta\tau_x) = y, s_x+\Delta\tau_x+\Delta\tau_y \ge s+h | \tau_{end} > s, \xi(s) = x) \tag{4.88}$$

$$= \int_{s_x \le s} p(t_x = s_x) \int_{s-s_x \le f \le s+h-s_x} p(\Delta\tau_x = f | \Delta\tau_x \ge s-s_x) P_{xy} Pr(\Delta\tau_y \ge h-(f-s+s_x)) df\, ds_x \tag{4.89}$$

$$= P_{xy} \int_{s_x \le s} p(t_x = s_x) \int_{s-s_x \le f \le s+h-s_x} u_x \exp\left[-u_x(f-s+s_x)\right] \exp\left[-u_y(h-(f-s+s_x))\right] df\, ds_x \tag{4.90}$$

$$= P_{xy} \int_{s_x \le s} p(t_x = s_x) \int_{s-s_x \le f \le s+h-s_x} u_x \exp\left[-u_y h\right] \exp\left[(u_y-u_x)(f-s+s_x)\right] df\, ds_x \tag{4.91}$$

$$= P_{xy} \int_{s_x \le s} p(t_x = s_x) u_x \exp\left[-u_y h\right] \frac{1}{u_y-u_x} \left[\exp\left[(u_y-u_x)h\right]-1\right] ds_x \tag{4.92}$$

$$= P_{xy} u_x \exp[-u_y h] h \tag{4.93}$$

$$= \langle y | G^{H_S, \phi_S} | x \rangle h + O(h^2). \tag{4.94}$$

Acknowledgement

We thank Jiayu Zhang, Thomas Vidick, Urmila Mahadev, Sevag Gharibian, Yupan Liu and Jielun Chen (Chris) for their helpful discussions. We thank the anonymous reviewers for their valuable suggestions for presentations and open questions. Jiaqing Jiang is supported by MURI Grant FA9550-18-1-0161 and the IQIM, an NSF Physics Frontiers Center (NSF Grant PHY-1125565).

Chapter 5

POSITIVE BIAS MAKES TENSOR-NETWORK CONTRACTION TRACTABLE

5.1 Introduction

Tensor network contraction is a powerful computational tool for studying quantum information and quantum many-body systems. It is widely used in estimating ground state properties [Whi93; Whi92; MVC07; VC21], approximating partition functions [EV15; Zha+10], simulating evolution of quantum circuits [MS08; Ped+17; Hua+20], as well as decoding for quantum error correcting codes [FP14; BSV14]. Mathematically, a tensor network $T := T(G, M)$ on a graph $G = (V, E)$ can be interpreted as an edge labeling model. Each edge can be labeled by one of d different colors, where d is called the *bond dimension*. Each vertex v is associated with a function $M^{[v]}$, called *tensor*, whose value depends on the labels of edges adjacent to v. The tensor $M^{[v]}$ can be represented as a vector by enumerating its values with respect to various edge labeling. For any edge labeling c, denote the value (entry) of the tensor $M^{[v]}$ by $M_c^{[v]}$. The *contraction value of tensor network* is defined to be

$$\chi(T) := \sum_{\text{edge labeling } c} \prod_{v \in V} M_c^{[v]}. \tag{5.1}$$

In applications of tensor networks, the contraction value represents the quantities of interest and the goal of tensor-network contraction algorithms is to compute the contraction value to high precision.

It is therefore a fundamental question to determine when $\chi(T)$ can be computed efficiently. Despite the practical and foundational importance of this question, unfortunately most rigorous results show that tensor network contraction is extremely hard, with very few tractable cases known, that is, cases for which a (quasi-)polynomial time algorithm exists. Specifically, it is well-known that computing $\chi(T)$ exactly is **#P-hard** [Sch+07] and therefore intractable in the worst case. The hardness can be further strengthened to the average case, where Haferkamp *et al.* [Haf+20] showed that even for random tensor networks on a 2D lattice, computing $\chi(T)$ exactly remains **#P-hard** for typical instances. There, the randomness is modeled by sampling the entries of the tensor network iid. from a Gaussian distribution with *zero* mean and

unit variance. Conversely, (quasi-)polynomial time algorithms are only known for restricted cases, like tensor networks on simple graphs of small tree-width [MS08], for example 1D line or tree; or for restrictive symmetric tensor network [PR17] where each entry is very close to 1, which requires that $\forall c, v, |M_c^{[v]} - 1| \leq 0.35/(\Delta + 1)$, where Δ is the maximum degree of the graph. Besides, for tensor networks with uniformly gapped parent Hamiltonians, (quasi)-polynomial time algorithm is known for computing local expectation values [SBE17].

But while efficient and provably correct tensor-network contraction algorithms are rare, for many many-body physics applications, state-of-the-art numerical algorithms achieve desired accuracy in practice [Orú19; Ban23]. To obtain a better understanding of when and why such heuristics work, it is important to identify new tractable cases in tensor network contraction. With this goal in mind, a recent line of work suggests an interesting direction, namely, that the sign structure of the tensor entries influences the entanglement and therefore affects the the complexity of tensor network contraction [GC24; Che+25]. In particular, it has been observed that there is a sharp phase transition in the entanglement and thus the complexity of approximating random tensor networks, when the mean of the entries is shifted from zero to positive [GC24; Che+25].

Main results and technical highlights

In this work, we rigorously investigate the impact of sign structure on the complexity of tensor network contraction in various regimes. We mainly focus on the contraction of the physically motivated 2D tensor networks, which are widely used as ground state ansatzes for local Hamiltonians [Cor16; Van+16] (Projected Entangled Pair States) and for the simulation of quantum circuits [Guo+19].

Recall that for random 2D tensor network whose entry has zero mean, the exact contraction is **#P-hard** [Haf+20]. We first show that a positive bias does not decrease the complexity of the exact contraction:

Theorem 109 (Informal version of Theorem 133) *The exact contraction of random 2D tensor network whose entries are iid. sampled from a Gaussian distribution with positive mean and unit variance remains* ***#P-hard****.*

While Theorem 109 indicates the exact contraction remains hard, our main result is proving that a small positive mean significantly decreases the computational complexity of multiplicative approximation, enabling a quasi-polynomial time algo-

rithm. This provides rigorous evidence that the sign structure of the tensor entries influences the contraction complexity, as observed and conjectured in previous works [GC24; Che+25]. In particular, we show the following.

Theorem 110 (Informal version of Theorem 123) *For random 2D tensor network with intermediate bond dimension $d \gtrsim n$, where the entries are iid. sampled from Gaussian distribution with mean $\mu \gtrsim 1/d$ and unit variance, there exists a quasi-polynomial time algorithm which with high probability approximates the contraction value up to arbitrary $1/poly$ multiplicative error.*

Here $a \gtrsim b$ means that a scales at least as fast as b.

While it is expected that tensor network contraction becomes easier when all entries are positive so that there is no sign problem, our result is much more fine-grained than this belief since our tensor network is only *slightly* positive, that is, a significant portion of the tensor entries are still negative. In particular, note that the mean value $\gtrsim 1/d$ is far less than the unit variance of the tensor entries. Compared to previous work [PR17] which shows that tensor-networks whose all entries are close to 1 can be contracted using Barvinok's method,[1] our result allows the entries to have significant fluctuations and to be a mixture of positive and negative values. We also note that the threshold value $\gtrsim 1/d$ matches the phase transition point predicted in [Che+25][2] with respect to the entanglement-based contraction algorithm. The fact that two different methods (our algorithm and the entanglement-based algorithm) admit the same threshold might indicate that there is a genuine phase transition in the complexity of tensor network contraction at this point. The requirement of $d \gtrsim n$ on the bond dimension in Theorem 110 is due to the fact that certain concentration effects set in at $d \sim n$. One may wonder then whether the intermediate bond dimension and the nonzero mean make the mean contraction value $\mu^n d^{2n}$ (attained when all entries in the tensor network take the mean value μ) a precise guess for the contraction value, that is $\chi(T) = \mu^n d^{2n}(1 + 1/poly(n))$. This is not the case since a simple lower bound shows that the second moment of $\chi(T)/(\mu^n d^{2n})$ is at least 2. In comparison, our algorithm can achieve an arbitrary $1/poly(n)$ multiplicative error in quasi-polynomial runtime; recall Theorem 110. Besides, although Theorem 110 is formulated for random 2D tensor networks, the proposed algorithm is well-

[1] More precisely for 2D tensor network, it requires that $\forall c, v, |M_c^{[v]} - 1| \leq 0.35/(4+1) = 0.07$.

[2] To clarify, [Che+25] draws each tensor from a Haar random distribution. If one does the same calculation for drawing each entry from Gaussian random distribution, the predicted phase transition point will also be approximately $1/d$.

defined and runs in quasi-polynomial time for an arbitrary graph G of constant degree, which may inspire new heuristic algorithms for general tensor networks.

Besides studying the average case complexity for approximating slightly positive tensor networks, we also investigate the complexity of approximating (fully) positive tensor networks, where all the entries are positive. Approximate contraction of positive tensor network is directly related to approximate counting, and we give a simple proof to show that

Theorem 111 (Informal version of Theorem 139) *$1/poly(n)$ multiplicative approximation of positive tensor network is **StoqMA-hard**. The **StoqMA-hard** remains even if we relax the multiplicative error from $1/poly(n)$ to a value exponentially close to one.*

Here, **StoqMA** is the complexity class whose canonical complete problem is to decide the ground energy for stoquastic Hamiltonians [BBT06].

In addition to multiplicative approximation, we also investigate the impact of sign in the hardness of tensor network contraction w.r.t. *certain additive* error. In particular, previously Arad and Landau [AL10] showed that approximating the contraction value w.r.t. the matrix 2-norm additive error is equivalent to quantum computation, that is **BQP-complete**. In contrast, we prove that if the tensor network is *positive*, where all entries are non-negative, then approximating the contraction value w.r.t matrix 1-norm additive error is equivalent to classical computation, that is **BPP-complete**.

Theorem 112 (Informal version of Theorem 135) *Given a positive tensor network $T \coloneqq T(G, M)$ on a constant-degree graph G. Given an arbitrary order of the vertex $\{v\}_v$, one can view each tensor $M^{[v]}$ as a matrix $O^{[v]}$ by specifying the in-edges and out-edges. It is **BPP-complete** to estimate $\chi(T)$ with additive error $\epsilon\Delta_1$, for $\Delta_1 \coloneqq \prod_v \|O^{[v]}\|_1$ and $\epsilon = 1/poly(n)$.*

Technically [AL10] simulates general matrix multiplication by quantum circuits. In Theorem 112 we simulate non-negative matrix multiplication by random walks.

Technical highlights. Our main technical contribution is to show that a small mean value dramatically decreases the complexity of approximate contraction. Our result significantly extends the regime in which efficient approximate contraction algorithms for tensor networks are known. This is formalized and proved in Theorem

110. The algorithm in Theorem 110 differs from commonly used numerical algorithms for tensor network contraction, which are based on the truncation of singular value decomposition and whose performance is determined by entanglement properties [Has07; Ara+17]. Instead, for Theorem 110 we use Barvinok's method from approximate counting. This method has previously been used for approximating the permanent, the hafnian [Bar16a; Bar16b] and partition functions [Bar14; PR17].

At a high level, Barvinok's method interprets the contraction value $\chi(T)$ as a polynomial $G(z)$ where $G(1) = \chi(T)$, and uses Taylor expansion of $\ln G(z)$ at $z = 0$ to get an additive error approximation of $\ln G(1)$, thus an multiplicative approximation of $\chi(T)$. The key technical part of applying Barvinok's method to different tasks is proving the corresponding $G(z)$ is root-free in the disk centered at 0 with radius slightly larger than 1, which ensures that $\ln G(z)$ is analytic in this disk. Denote this disk as $\mathcal{B}$. Previously Patel and Regts [PR17] had applied Barvinok's method to symmetric tensor networks where all the entries are close to 1 within error $0.35/5 = 0.07$, by proving that $G(z)$ is root-free in $\mathcal{B}$. Our setting allows entries to have significant fluctuations, and thus the root-free proof in [PR17] does not apply. We circumvent this problem using the following two ideas.

- **Root-free strip inspired from approximating random permanent**. Instead of applying Barvinok's method directly and proving $G(z)$ is root-free in the disk $\mathcal{B}$, we apply a variant of Barvinok's method used for approximating random permanents by Eldar and Mehraban [EM18]. There, the idea is to use Jensen's formula to find a root-free *strip* connecting 0 and 1. The advantage of this variant is that it allows for a constant number of zeros in the unit disk $\mathcal{B}$ as long as there is a root-free path of some width connecting 0 and 1. We notice that this method from approximating permanent can also be applied to random tensor networks. In particular, using Jensen's formula [EM18], the number of roots in $\mathcal{B}$ can be bounded by estimating the second moment $E|h(z)|^2$, where $h(z)$ is a rescaled version of $G(z)$, and E denotes the expectation value over the randomness of the tensor network. Besides, compared to [EM18], in our setting we use a different and much simpler method to find the root-free strip.

- **Mapping random instance to statistical mechanical model.** Since we are working on random tensor networks, the technique used by Eldar and Mehraban [EM18] to bound $E|h(z)|^2$ for random permanents fails entirely. To bound $E|h(z)|^2$ for random tensor networks, we adapt a technique of mapping random instances to a classical statistical mechanical model (statmech

model). This technique has been used in the physics literature to study phase transitions in random tensor networks [Yan+22; LC21; Hay+16] and random circuits [BCA20; BBA24].

Although in general such mapping and the properties of the statmech model like its partition function are hard to analyze, and heuristic approximations are needed in many related literature, we notice that in our application the statmech model is simple enough to obtain a rigorous result. In particular, we show that $E|h(z)|^2$ is proportional to the partition function of a 2D Ising model with magnetic field parameterized by z. Then we further use the finite-size variant of the Onsager solution of the 2D Ising model [Kau49; Maj66] to get a decent estimate of $E|h(z)|^2$ for relevant ranges of z, allowing the Barvinok method to be applied.

Conclusions and open problems

We investigate how the contraction complexity of tensor networks depends on the sign structure of the tensor entries. For random tensor networks in 2D, we show that there is a quasi-polynomial time approximation algorithm if the entries are drawn with a small nonzero mean and intermediate bond dimension. At the same time, exactly computing the contraction value in this setting remains **#P-hard**. Our work thus provides rigorous evidence for the observations [GC24; Che+25] that shifting the mean by a small amount away from zero dramatically decreases the contraction complexity. Compared to [PR17] which similarly uses Barvinok method but requires all entries to be close to 1, our setting allows significant fluctuations in the entries and greatly extends the known region where (quasi-)polynomial time average-case contraction algorithms exist. While it is expected that tensor network contraction becomes easier when all entries are positive, our result suggests that even for *slightly* positive tensor networks, one can still utilize the sign structure to obtain a (quasi-)efficient algorithm. Moreover, [Che+25] observed that the standard entanglement-based contraction algorithm starts working at $\mu \gtrsim 1/d$. We show that a *completely different* rigorous Barvinok-based algorithm also starts working at $\mu \gtrsim 1/d$. This might indicate that there is a genuine phase transition in the complexity of tensor-network contraction happening here.

Indeed, we also assess the worst-case complexity of approximating fully positive tensor networks. Specifically, we prove that approximating the contraction value of positive tensor networks multiplicative error close to unity is **StoqMA**-hard. But when requiring only an inverse polynomial additive error in matrix 1-norm there

exists an efficient classical algorithm.

Our work initiates the rigorous study of how the computational difficulty of contracting tensor networks depends on the sign structure of the tensor entries. If one views the hardness of contraction as a function of mean value and bond dimension, while we identify a new tractable region, there are many open questions left.

- First, while our approximation algorithm based on Barvinok's method works for typical instances, it remains an open question to what extent a positive bias can ease practical tensor network contraction. It would therefore be interesting to understand whether our algorithm or variations of it can aid in practically interesting cases.
- Moreover, our current proof works for intermediate bond dimension but not for constant bond dimension. Potentially, techniques like cluster expansion [MH21; HPR19] may be used to design new contraction algorithm for constant bond dimension, proving a correspondent of Theorem 112 for that setting. It might be worth mentioning that a direct application of cluster expansion does not work, where one can prove the expansion series is not absolutely convergent. More refined techniques are thus required.
- Finally, although current numerical algorithms have poor performance for zero-mean tensor network contraction, there is no known rigorous complexity result to establish the hardness of approximate contraction.

In the context of approximating fully positive tensor networks, it would be interesting to see whether there exists an efficient classical algorithm that can achieve the same (2-norm) precision as a quantum computer for positive tensor networks, or whether there is a room for quantum advantage even for positive tensor networks.[3]

Structure of the manuscript

The structure of this manuscript is as follows. In Section 5.2 we define notations and tensor networks. In Section 5.3 we review Barvinok's method and its variant. In section 5.4 we adapt Barvinok's method to tensor network contraction. In Section 5.5 we give a quasi-polynomial time algorithm for approximating random 2D tensor networks with small mean and intermediate bond dimension. In Section 5.6 we prove the results concerning approximating positive tensor networks.

[3] We acknowledge Zeph Landau for raising this question.

5.2 Notation and tensor networks

In this section, we introduce necessary notations and definitions for tensor networks.

Notation. We use $[k]$ for $\{0, 1, \ldots, k-1\}$. We use $\overline{z}$ to denote its complex conjugate. For $v \in \mathbb{C}$ and $\epsilon \in \mathbb{R}$, we say $\hat{v}$ approximates v with ϵ-multiplicative error if $|v - \hat{v}| \leq \epsilon |v|$. For $x \in \{\pm 1\}^n$, we use $|x|$ to denote the number of -1 in x. We use δ_{ij} for the delta function, where $\delta_{ij} = 1$ if $i = j$ and equals 0 otherwise.

For a matrix $A \in \mathbb{C}^{s \times t}$, the matrix p-norm is defined as

$$\|A\|_p \coloneqq \sup_{x \neq 0, x \in \mathbb{R}^t} \frac{\|Ax\|_p}{\|x\|_p}, \tag{5.2}$$

The 2-norm $\|A\|_2$ is known as the spectral norm. The 1-norm equals to the maximum of the absolute column sum, that is

$$\|A\|_1 = \max_{1 \leq j \leq t} \sum_{i=1}^{s} |A_{ij}|.$$

For $\mu, \sigma \in \mathbb{R}$, we use $X \sim \mathcal{N}_{\mathbb{R}}(\mu, \sigma^2)$ to denote that the random variable X is sampled from the Gaussian distribution with mean μ and standard derivation σ. For $\mu \in \mathbb{C}$, we use $\mathfrak{R}(\mu), \mathfrak{I}(\mu) \in \mathbb{R}$ to denote the real and imaginary part of μ, i.e. $\mu = \mathfrak{R}(\mu) + \mathfrak{I}(\mu) i$. We use $X \sim \mathcal{N}_{\mathbb{C}}\left(\mu, \sigma^2\right)$ if

$$\mathfrak{R}\left(X\right) \sim \mathcal{N}_{\mathbb{R}}\left(\mathfrak{R}(\mu), \frac{\sigma^2}{2}\right), \mathfrak{I}(X) \sim \mathcal{N}_{\mathbb{R}}\left(\mathfrak{I}(\mu), \frac{\sigma^2}{2}\right).$$

Tensors and tensor networks. A *tensor* N of *rank* k and bond dimension d is an array of d^k complex numbers which is indexed by $N_{i_1, i_2, \ldots, i_k}$, where i_s takes values from $[d]$ for $1 \leq s \leq k$. We call the complex numbers in the array the *entries* of the tensor N. We use $\overline{N}$ to denote the tensor obtained by complex conjugating every entry of N. For two tensors N and M with the same rank and bond dimension, the addition $N + M$ is a new tensor obtained by addition of the two arrays. For convenience, in the rest of the paper we assume that all the indices have the same dimension d.[4] We will always assume $d = O(poly(n))$.

A *tensor network* $T \coloneqq T(G, M)$ is described by an n-vertex graph $G = (V, E)$ and a set of tensors on vertices, denoted as $M = \{M^{[v]}\}_v$. More specifically, on each

[4] This is not a restriction, since we can just take d to be the maximum dimension of all indices in the tensor network, and introduce dummy dimensions elsewhere.

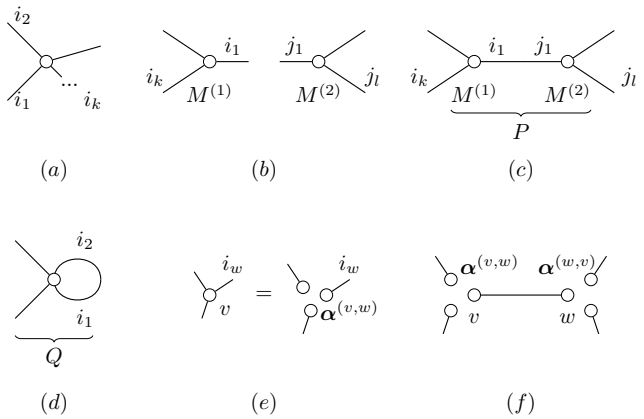


Figure 5.1: Tensor and operations on tensors. (a) A rank-k tensor. (b) The product of a rank-k and a rank-l tensor. (c) Contracting two tensors by identifying edges i_1 and j_1. (d) Contracting two free edges in the same tensor. (e) A special tensor which can be factorized into a product of rank-1 tensors. (f) If all tensors have a factorized structure, then the contraction value of tensor network can be computed by contracting the rank-1 tensors.

vertex v of degree k_v there is a tensor $M^{[v]}$ of rank k_v, where the indices $i_1, \ldots, i_{k_v}$ correspond to k_v edges. One can interpret $[d]$ as d different colors, and $i_s \in [d]$ represents that we label the corresponding edge with color i_s. Denote this edge labeling as $c : E \to [d]^{|E|}$, we write $M_c^{[v]} \coloneqq M^{[v]}_{i_1,\ldots,i_{k_v}}$. With an arbitrary ordering of edges, we can conceive of the labeling c as a vector $c \in [d]^{|E|}$. The *contraction value of tensor network* is then defined to be

$$\chi(T) \coloneqq \sum_{c \in [d]^{|E|}} \prod_{v \in V} M_c^{[v]}. \tag{5.3}$$

Product and contraction. Besides Eq. (5.3), another equivalent way of defining the contraction value of tensor network $\chi(T)$ is via a graphical representation, which is more intuitive and will be used in the proofs. As in Figure 5.1 (a), for a tensor of rank k, we represent it as a vertex with k edges. We term such edges which connect to only one vertex *free edges.*

With this graphical representation, we introduce two operations on tensors. Consider a tensor $M^{(1)}$ of rank k, with free edges indexed by $i_1, \ldots, i_k$, and another tensor $M^{(2)}$ of rank l, with free edges indexed by $j_1, \ldots, j_l$. We use Figure 5.1 (b) to represent the *product* of $M^{(1)}, M^{(2)}$, that is a new tensor $M^{(1)} \otimes M^{(2)}$ of rank $k + l$ and with free edges indexed by $i_1, \ldots, i_k; j_1, \ldots, j_l$, where

$$\left(M^{(1)} \otimes M^{(2)}\right)_{i_1,\ldots,i_k,j_1,\ldots,j_l} \coloneqq M^{(1)}_{i_1,\ldots,i_k} M^{(2)}_{j_1,\ldots,j_l}. \tag{5.4}$$

The product operation can be generalized to multiple tensors recursively,

$$M^{(1)} \otimes M^{(2)} \otimes M^{(3)} \otimes M^{(4)} \ldots := \left(\left(\left(M^{(1)} \otimes M^{(2)}\right) \otimes M^{(3)}\right) \otimes M^{(4)}\right) \ldots \quad (5.5)$$

One can check that the order of this recursion does not change the final tensor.

Another operation which defines a new tensor is *contraction*, that is, connecting different tensors by identifying a free edge of one tensor with a free edge of another tensor and summing over that index. Starting from the two tensors $M^{(1)}$ and $M^{(2)}$, contracting the indices i_1 and j_1 results in a new tensor P of rank $k+l-2$, with free edges indexed by $i_2, \ldots, i_k, j_2, \ldots, j_l$, where

$$P_{i_2,\ldots,i_k,j_2,\ldots,j_l} = \sum_{f \in [d]} M^{(1)}_{f,i_2,\ldots,i_k} M^{(2)}_{f,j_2,\ldots,j_l}. \quad (5.6)$$

Graphically, this operation is represented by joining the two contracted edges; see Figure 5.1 (c).

One can also contract two free edges in the same tensor. Consider the contraction of the indices i_1, i_2 of $M^{(1)}$. Figure 5.1 (d) represents a new tensor Q of rank $k-2$ and with free edges $i_3, \ldots, i_k$ where

$$Q_{i_3,\ldots,i_k} = \sum_{s \in [d]} M^{(1)}_{s,s,i_3,\ldots,i_k}. \quad (5.7)$$

The contraction operations can be generalized to contracting multiple pairs of edges by contracting the pairs one by one. Note that the order of contraction does not change the final tensor.

One can check that given a tensor network $T = T(G, M)$, the contraction value of tensor network defined by Eq. (5.3) is equal to the value obtained when contracting $\bigotimes_v M^{[v]}$ by identifying the free edges according to the edges of G.

For any vertex v, use $N(v)$ for the vertices adjacent to v in G.

Example 113 *Here we give an example of how the graphical representation simplifies the computation of the contraction value. Consider a case in which each $M^{[v]}$ has a factorized structure, that is, there exist vectors $\boldsymbol{\alpha}^{(v,w)} \in \mathbb{C}^d$ for $w \in N(v)$ such that*

$$M^{[v]} = \bigotimes_{w \in N(v)} \boldsymbol{\alpha}^{(v,w)},$$

equivalently the entry

$$M^{[v]}_{\ldots,i_w,\ldots} = \prod_{w \in N(v)} \boldsymbol{\alpha}^{(v,w)}_{i_w}.$$

Then $M^{[v]}$ can be represented by a product of $|N(v)|$ tensors as shown in Figure 5.1 (e). As a consequence, one can check that in this special case computing $\chi(T)$ is easy: as in Figure 5.1 (f), one can write $\chi(T)$ in a factorized way, where each edge (v, w) contributes a factor $\langle \boldsymbol{\alpha}^{(v,w)}, \boldsymbol{\alpha}^{(w,v)} \rangle := \sum_{f \in [d]} \boldsymbol{\alpha}_f^{(v,w)} \boldsymbol{\alpha}_f^{(w,v)}$ as follows:

$$\chi(T) = \sum_{\text{all edge labeling } c} \prod_{v} M_c^{[v]} \tag{5.8}$$

$$= \prod_{\text{edges } (v,w)} \langle \boldsymbol{\alpha}^{(v,w)}, \boldsymbol{\alpha}^{(w,v)} \rangle. \tag{5.9}$$

2D tensor network. We call a tensor network $T = T(G, M)$ a *2D tensor network* if the graph G is a 2D lattice. We assume the lattice has size $L_1 \times L_2$ with $n = L_1 \times L_2$, and satisfies periodic boundary conditions, that is can be mapped onto a torus. The periodic boundary condition is mainly to ease the analysis. In particular, every vertex has degree 4. For simplicity, we assume that L_2 is even.

- For $\mu > 0$, we define a *2D (μ, n, d)-Gaussian tensor network* $T(G, M)$ as an n-vertex 2D tensor network with bond dimension d, where the entries of every tensor $M^{[v]}$ are iid. sampled from the complex Gaussian distribution $\mathcal{N}_{\mathbb{C}}(\mu, 1)$, i.e.

$$(M^{[v]})_{i_1,i_2,i_3,i_4} \overset{i.i.d.}{\sim} \mathcal{N}_{\mathbb{C}}(\mu, 1). \tag{5.10}$$

- For technical reasons, for $z \in \mathbb{C}$ we also define the *2D (z, n, d)-shifted-Gaussian tensor network* $T(G, M)$, which is an n-vertex 2D tensor network with bond dimension d: For every vertex v, Let $(A^{[v]})_{i_1,i_2,i_3,i_4} \overset{i.i.d.}{\sim} \mathcal{N}_{\mathbb{C}}(0, 1)$, the entries of $M^{[v]}$ are defined to be

$$(M^{[v]})_{i_1,i_2,i_3,i_4} := 1 + z \cdot (A^{[v]})_{i_1,i_2,i_3,i_4}. \tag{5.11}$$

  We write the tensor $M^{[v]}$ as

$$M^{[v]} = J^{[v]} + zA^{[v]}, \tag{5.12}$$

  where $J^{[v]}$ is a tensor whose entries are all 1. Note that $J^{[v]}$ has a factorized structure

$$J^{[v]} = [1, \ldots, 1]^{\otimes 4}.$$

  We abbreviate the 2D (z, n, d)-shifted-Gaussian tensor network $T(G, M)$ as $T_A(z)$ where $A := \{A^{[v]}\}_v$.

5.3 Barvinok's method and its variant

In this section we review Barvinok's method, which was first developed by Barvinok [Bar16a; Bar16b], and is a general method for approximate counting. It has been applied to approximating permanents [EM18], hafnians [Bar16a; Bar16b] and partition functions [Bar14; PR17]. In particular, Barvinok's method was applied to contracting symmetric tensor networks where all entries are very close to 1 [PR17]. Our setting allows the entries having significant fluctuations where the standard Barvinok's method fails. Instead our algorithm builds from a special variant of Barvinok's method used in approximating random permanents [EM18], which we summarize below. All Lemmas and Theorems quoted here are proven in [Bar16a; Bar16b; EM18].

Roughly speaking, the idea of Barvinok's method is to approximate an analytic function via its Taylor series around 0. The performance of this approximation depends on the location of the roots of the analytic function.

Consider a polynomial $G(z)$ of degree n, where $G(z) \neq 0$ for z on a simply connected open area containing 0 in the complex plain. We choose the branch of the complex logarithm, denoted as LN, such that $\mathrm{LN}\, G(0)$ is real. Define $F(z) := \mathrm{LN}\, G(z)$. In our application, $G(1)$ will encode the contraction value of tensor network. An additive approximation of $F(1)$ will give a multiplicative approximation to $G(1)$. For $r, w > 0$, we use $\mathcal{B}(r) \subset \mathbb{C}$ to denote the the disk of radius r centered at 0, and use $\mathcal{T}(re^{i\theta}, w)$ to denote the strip of width w around the line between 0 and $re^{i\theta}$, that is

$$\mathcal{B}(r) := \{z \in \mathbb{C} \mid |z| \leq r\},$$
$$\mathcal{T}(re^{i\theta}, w) := \{z \in \mathbb{C} \mid -w \leq \Re(ze^{-i\theta}) \leq r + w, \quad |\Im(ze^{-i\theta})| \leq w\}.$$

The following lemma quantifies the approximation error incurred by approximating $F(z)$ using a root-free disk of $G(z)$.

Lemma 114 (Approximation using a root-free disk, see the proof of Lemma 1.2 in [Bar16b]) *Let $G(z)$ be a polynomial of degree n and suppose $G(z) \neq 0$ for all $|z| \leq \eta$ where $\eta > 1$. Let $F(z) := \mathrm{LN}\, G(z)$. Then $F(z)$ is analytic for $|z| \leq 1$. Moreover, consider a degree m Taylor approximation of $F(z)$,*

$$P_m(z) := F(0) + \sum_{k=1}^{m} \left.\frac{\partial^k F(z)}{\partial z^k}\right|_{z=0} \frac{z^k}{k!}. \tag{5.13}$$

Then, for all $|z| \leq 1$,

$$|F(z) - P_m(z)| \leq \frac{n}{(m+1)\eta^m(\eta - 1)}. \tag{5.14}$$

Recall that additive approximation of $F(z)$ implies multiplicative approximation of $G(z)$. To translate Lemma 114 into an efficient algorithm, one further needs to efficiently compute the first few derivatives of F. Barvinok shows that the derivatives of F can be efficiently computed using the derivatives of G.

Lemma 115 ([Bar16a]) *Let* $G(z)$ *be a polynomial of degree* n *and* $G(z) \neq 0$. *If one can compute the first* l *derivatives of* $G(z)$ *at* $z = 0$ *in time* $t(n)$, *then one can compute the first* l *derivatives of* $F(z) \coloneqq LN\, G(z)$ *at* $z = 0$ *in time* $O(l^2 t(n))$.

Lemma 114 implies that the Taylor series at $z = 0$ gives a good approximation to $F(z) = \mathrm{LN}\, G(z)$, as long as $G(z)$ is root-free in a disk centered at 0 that contains z. Lemma 114 can be generalized to the case in which $G(z)$ is allowed to have roots in the disk, but instead there exists a root-free strip from 0 to z. The main idea in this generalization is to construct a new polynomial $\phi(z)$ which embeds the disk into a strip. Given such ϕ, we can then approximate $G(\phi(z))$ using the approximation via a root-free disk, since $G(\phi(z))$ is guaranteed to be root free in a disk of some radius. Furthermore, we can still use this approximation to estimate $G(1)$, which will encode our quantity of interest.

Lemma 116 (Embedding a disk into a strip, Lemma 8.1 in [Bar16a]) *For* $0 < \rho < 1$, *define*

$$\alpha = \alpha(\rho) = 1 - e^{-\frac{1}{\rho}}, \quad \eta = \eta(\rho) = \frac{1 - e^{-1-\frac{1}{\rho}}}{1 - e^{-\frac{1}{\rho}}} > 1, \tag{5.15}$$

$$K = K(\rho) = \left\lfloor \left(1 + \frac{1}{\rho}\right) e^{1+\frac{1}{\rho}} \right\rfloor \geq 14, \quad \sigma = \sigma(\rho) = \sum_{k=1}^{K} \frac{\alpha^k}{k} \text{ and} \tag{5.16}$$

$$\phi(z) = \phi_\rho(z) = \frac{1}{\sigma} \sum_{k=1}^{K} \frac{(\alpha z)^k}{k}. \tag{5.17}$$

Then $\phi(z)$ *is a polynomial of degree* K *such that* $\phi(0) = 0, \phi(1) = 1$, *and embeds the disk of radius* η *into the strip of width* 2ρ, *i.e.,*

$$-\rho \leq \Re(\phi(z)) \leq 1 + 2\rho \text{ and } |\Im(\phi(z))| \leq 2\rho \text{ provided } |z| \leq \eta. \tag{5.18}$$

Corollary 117 (Approximation using a root-free strip) *Let $G(z)$ be a polynomial of degree n and suppose there exists a constant $\rho \in (0,1)$ such that $G(z) \neq 0$ for all z in the strip $\mathcal{T}(1, 2\rho)$. Define $\eta, K, \phi(z)$ as in Lemma 116 where $\eta > 1$. Let*

$$F(z) \coloneqq LN\, G(\phi(z)).$$

Then $F(z)$ is analytic for $|z| \leq 1$. Moreover, consider a degree m Taylor approximation of $F(z)$,

$$P_m(z) \coloneqq F(0) + \sum_{k=1}^{m} \frac{\partial^k F(z)}{\partial z^k}|_{z=0} \frac{z^k}{k!}. \tag{5.19}$$

Then, for all $|z| \leq 1$,

$$|F(z) - P_m(z)| \leq \frac{nK}{(m+1)\eta^m(\eta - 1)}. \tag{5.20}$$

Proof: Recall that $\eta > 1$. By Lemma 116 and the assumption that $G(z) \neq 0$ for all z in $\mathcal{T}(1, 2\rho)$, we have for any $|z| \leq \eta$, $G(\phi(z)) \neq 0$. Note that $G(\phi(z))$ is a polynomial in z of degree nK. Then use Lemma 114 w.r.t $G(\phi(z))$ and $F(z) \coloneqq \mathrm{LN}\, G(\phi(z))$ we prove the Corollary. ∎

In the above Lemmas, we have assumed $G(z)$ is a fixed polynomial, and the performance of the Taylor expansion of $F(z)$ depends on the location of roots of $G(z)$. When $G_A(z)$ are random polynomials indexed by randomness A, [EM18] illustrates a way of using Jensen's formula to estimate the expectation of the number of roots.

For convenience of later usage, in the following we use the notation $h_A(z)$ for polynomials instead of $G_A(z)$. In later applications $h_A(z)$ will be a rescaled version of $G_A(z)$. By Lemma 119 [EM18] connects the expected number of roots in a disk to the second moment of h_A.

Definition 118 (Average Sensitivity [EM18]) *Let $h_A(z)$ be a random polynomial where A is sampled from some random ensembles and $h_A(0) \neq 0$. For any real number $r > 0$, the stability of $h_A(z)$ at point r is defined as*

$$k_h(r) \coloneqq E_\theta E_A \left[\frac{|h_A(re^{i\theta})|^2}{|h_A(0)|^2} \right], \tag{5.21}$$

where $E_\theta[\cdot] = \int_{\theta=0}^{2\pi}[\cdot]\frac{d\theta}{2\pi}$ is the expectation over θ from a uniform distribution over $[0, 2\pi)$, and E_A is the expectation over the randomness of A.

Lemma 119 (Proposition 8 [EM18]) *Let $h_A(z)$ be a random polynomial where A is sampled from some random ensemble and $h_A(0) \neq 0$. Let $N_A(r)$ be the number of roots of $h_A(z)$ inside $\mathcal{B}(r)$, and $0 < \lambda < 1/2$. Then,*

$$E_A\left[N_A(r - r\lambda)\right] \leq \frac{1}{2\lambda} \ln k_h(r). \tag{5.22}$$

Lemma 119 bounds the expectation value of the number of roots in $\mathcal{B}(r)$. In later sections, we will apply Lemma 119 to show that our polynomial of interest has very few roots in the disk. This will allow us to find a root-free strip with high probability. For completeness, we provide a proof of Lemma 119 in Appendix 5.8.

5.4 Tensor network contraction algorithm from Barvinok's method

We are now ready to present our algorithm for approximate tensor network contraction. The algorithm is based on Barvinok's method and takes the following inputs,

- A tensor network $T = T(G, M)$, where $G = (V, E)$ is a graph comprising $n = |V|$ vertices and has constant degree κ.
- A precision parameter $\epsilon \in (0, 1]$.

The goal is to approximate $\chi(T)$ with ϵ-multiplicative error. In order to achieve this, we will choose the following parameters that will enter the algorithm appropriately.

- A set of non-zero complex values $\{\mu_v\}_{v \in G}$. We will choose μ_v to be the mean value of the entries of the tensor $M^{[v]}$ at vertex $v \in V$.
- A complex value $z_{end} \neq 0$.
- A real value $0 < \rho < 1$. This value will determine the width of the strip $\mathcal{T}(1, 2\rho)$ in the complex plane.

The algorithm we describe in this section is well-defined for an arbitrary tensor network. In Section 5.5 we will apply this algorithm to random 2D tensor network whose entries have a small positive bias and show that it succeeds with high probability.

The polynomial

To apply Barvinok's method in Corollary 117, we map the contraction value of tensor network to a polynomial as follows. For each vertex v, with some abuse of notations, we use $J^{[v]}$ to represent the tensor by substituting all entries in $M^{[v]}$ by 1. We define

$$A^{[v]} \coloneqq (\mu_v^{-1} M^{[v]} - J^{[v]}) \cdot z_{end}^{-1}. \tag{5.23}$$

In other words,

$$\mu_v^{-1} M^{[v]} = J^{[v]} + z_{end} \cdot A^{[v]}. \tag{5.24}$$

Eq. (5.24) states that we intepret the normalized version of $M^{[v]}$ as the all-one tensor $J^{[v]}$ interpolated by $A^{[v]}$. We note that we allow z_{end} to be much larger than 1.

Since the contraction value of $T(G, \{M^{[v]}\}_v)$ equals $\prod_v \mu_v$ times the contraction value of the normalized tensor network $T(G, \{\mu_v^{-1} M^{[v]}\}_v)$, without loss of generality, from now on we assume that the tensor network has been normalized and

$$M^{[v]} = J^{[v]} + z_{end} \cdot A^{[v]}. \tag{5.25}$$

If we substitute z_{end} with a variable z in Eq. (5.25) for each tensor $M^{[v]}$, we will obtain a family of new tensor networks, denoted by $T_A(z)$. The contraction value $\chi(T_A(z))$ is a degree-n polynomial in z. Denoting this polynomial as $g_A(z)$, we have

$$g_A(z) = \chi(T_A(z)), \quad 0 \le z \le z_{end} \tag{5.26}$$

$$g_A(z_{end}) = \chi(T_A(z_{end})) = \chi(T). \tag{5.27}$$

Recall that $0 < \rho < 1$. Define the polynomial $\phi(z)$ as in Lemma 116. For convenience of applying Barvinok's method, we also define $G_A(z)$ by rescaling $g(z)$,

$$G_A(z) \coloneqq g_A(z \cdot z_{end}), \quad 0 \le z \le 1. \tag{5.28}$$

$$F_A(z) \coloneqq \mathrm{LN}\, G_A(\phi(z)). \tag{5.29}$$

$F_A(z)$ will be analytic in the disk $\mathcal{B}(1)$ if $G_A(z)$ is root-free in the strip $\mathcal{T}(1, 2\rho)$.

Computing the derivatives of g_A

We first note that the first few derivatives of $g_A(z)$ can be computed efficiently.

Lemma 120 *For any integer m, the first m derivatives $\{g_A^{(k)}(0)\}_{k=0,\ldots,m}$ can be computed in time $O(m^2 d^{\kappa m} n^{m+1})$.*

Proof: For any subset $S \subseteq V$, we denote by $\frac{\partial T_A(z)}{\partial S}$ the tensor network which is obtained by substituting the tensor $M^{[v]}$ with $A^{[v]}$ at every vertex $v \in S$. Using the product rule of derivatives and induction on k, one can check that

$$\frac{\partial^k g_A(z)}{\partial z^k} = \sum_{S \subseteq V, |S|=k} \chi\left(\frac{\partial T_A(z)}{\partial S}\right) \cdot k! \tag{5.30}$$

In particular, note that when $z = 0$, by the definition of $\frac{\partial T_A(0)}{\partial S}$, for any vertex $w \notin S$, the corresponding tensor at w in $\frac{\partial T_A(0)}{\partial S}$ is

$$M^{[v]} = J^{[v]} + 0 \cdot A^{[v]} = J^{[v]}.$$

As in Example 113 and Figure 5.1 (f), we can decompose each tensor $J^{[v]} = [1, \ldots, 1]^{\otimes \kappa}$ as a product of κ all-one vectors $[1, \ldots, 1]$. Then the graphical representation of $\frac{\partial T_A(0)}{\partial S}$ consists of many disconnected sub-graphs, where each sub-graph has at most $|S| \leq m$ vertices. The contraction value $\chi(\frac{\partial T_A(0)}{\partial S})$ is the product of the contraction value of each subgraph, and can be computed in time $n \cdot O(d^{\kappa m} m)$. Here n is an upper bound of the number of sub-graphs, and $O(d^{\kappa m} m)$ is the cost of directly contracting an m vertices subgraph of a tensor network on degree-κ graph. Thus by Eq. (5.30) for any $k \leq m$, $g_A^{(k)}(0)$ can be computed in time $O(n^m d^{\kappa m} nm)$. We conclude that the first m derivatives $\{g^{(k)}(0)\}_{k=0,\ldots,m}$ can be computed in time $O(m^2 d^{\kappa m} n^{m+1})$. ∎

In later proofs we will set $m = O(\ln(n/\epsilon))$. When $d = poly(n)$ and $\epsilon = O(1/poly(n))$, the cost $O(m^2 d^{4m} n^{m+1})$ of computing the m-th derivative is then quasi-polynomial.

Using Lemma 120 one can efficiently compute the first few derivatives of $F_A(z)$.

Lemma 121 *Assume that ρ is a constant. Then for any integer m, the first m derivatives of $\{F_A^{(k)}(0)\}_{k=1}^m$ can be computed in time*

$$O\left(m^4 d^{\kappa m} n^{m+1} + m^6\right).$$

Proof: By Lemma 120 and the definition of $G_A(z)$, the first m derivatives $\{G_A^{(k)}(0)\}_{k=0}^m$ can be computed in time $O(m^2 d^{\kappa m} n^{m+1})$. Besides, from Lemma 116 and the assumption that ρ in the definition of ϕ is a constant, $\phi(z)$ is a polynomial of degree K where $K = K(\rho)$ is a constant, thus the first m derivatives $\{\phi^{(k)}(0)\}_{k=0}^m$ can be computed in time $O(m)$. Thus by Lemma 143 in Appendix 5.8, one can compute the the first m derivatives of the composite function $G_A(\phi(z))$ at $z = 0$ in time

$$O(m^2 d^{\kappa m} n^{m+1} + m^4).$$

Note that K is a constant and $G_A(\phi(z))$ is a polynomial of degree nK. Then by Lemma 115, we can compute the first m derivatives $F_A^{(k)}(0)$ in time

$$O\left(m^4 d^{\kappa m} n^{m+1} + m^6\right).$$

■

The algorithm and its performance

Our goal is to approximate $\mathrm{LN}\, G_A(1) = \mathrm{LN}\, G_A(\phi(1))$ with respect to an additive error, which will give a multiplicative approximation to $G_A(1) = \chi(T)$. The algorithm is just computing the derivatives and $P_m(z)$ in Corollary 117, that is Algorithm 7.

Algorithm 7 Barvinok(G_A, m, ρ)

1: Let $F_A(z) \coloneqq \ln G_A(\phi(z))$ with $\phi = \phi_\rho$ as defined in Eq. (5.17).
2: Compute the first m derivatives $\{F_A^{(k)}(0)\}_{k=0}^m$ of F_A using Lemma 121.
3: Compute $P_m(1) \coloneqq F(0) + \sum_{k=1}^m F_A^{(k)}(0)\frac{1}{k!}$.
4: Return $\hat{\chi}(T) := e^{P_m(1)}$.

Algorithm 7 returns a good approximation of $\chi(T)$ with multiplicative error ϵ if $G_A(z)$ is root-free in $\mathcal{T}(1, 2\rho)$ and we set $m = O(\ln(n/\epsilon))$.

Theorem 122 *Let $0 < \rho < 1$ be a constant. If $G_A(z) \neq 0$ for any z in strip $\mathcal{T}(1, 2\rho)$, then for any ϵ, Algorithm 7 runs in time*

$$O\left(m^4 d^{\kappa m} n^{m+1} + m^6\right).$$

Choosing $m = O(\ln(n/\epsilon))$, Algorithm 7 outputs a value $\hat{\chi}(T)$ that approximates $\chi(T)$ with ϵ-multiplicative error. That is

$$|\hat{\chi}(T) - \chi(T)| \leq \epsilon |\chi(T)|.$$

Proof: As in Corollary 117 and Lemma 116, we define two constants η, K from ρ. Set

$$m \coloneqq \frac{\ln(enK/\epsilon) - \ln(\eta - 1)}{\ln \eta} = O(\ln(n/\epsilon)). \tag{5.31}$$

Define the polynomial $P_m(z)$ as in Corollary 117. Applying Corollary 117 to the functions $G_A(z), F_A(z)$, we have for $|z| \leq 1$,

$$|F_A(z) - P_m(z)| \leq \epsilon/e. \tag{5.32}$$

One can check that $|e^x - 1| \leq e|x|$ for complex x where $|x| \leq 1$. Thus for any $|z| \leq 1$,

$$\begin{aligned} |e^{F_A(z)} - e^{P_m(z)}| &= |e^{F_A(z)}| \cdot |1 - e^{P_m(z) - F_A(z)}| && (5.33)\\ &\leq |e^{F_A(z)}| \cdot e \cdot \epsilon/e. && (5.34)\\ &= \epsilon \cdot |e^{F_A(z)}|. && (5.35) \end{aligned}$$

Note that

$$\chi(T) = \chi(T_A(z_{end})) = e^{F_A(1)}. \tag{5.36}$$

Since

$$\hat{\chi}(T) \coloneqq e^{P_m(1)}. \tag{5.37}$$

Eq. (5.35) implies

$$|\chi(T) - \hat{\chi}(T)| \leq \epsilon \cdot |\chi(T)|. \tag{5.38}$$

The runtime of the algorithm is the time for computing the polynomial $P_m(1)$, which is dominated by the time for computing $F_A^{(k)}(0)$ for $k = 1, \ldots, m$. By Lemma 121 we can compute the the first m derivatives $F_A^{(k)}(0)$ in time

$$O\left(m^4 d^{\kappa m} n^{m+1} + m^6\right).$$

■

5.5 Approximating random PEPS with positive mean

In this section, we apply Algorithm 7 to the task of approximating the contraction value of 2D tensor networks. We show that the algorithm succeeds with high probability if the tensors are drawn randomly with vanishing positive mean and intermediate bond dimension $d \geq nc^{-1}$ for constant c. The formal statement of the result is as follows.

Theorem 123 *Suppose $d \geq nc^{-1}$ for some constant c. Let λ be an arbitrary small constant satisfying $0 \leq \lambda \leq e^{-3c}/80$. Let $\epsilon \in (0, 1]$ be a precision parameter. Suppose*

$$\mu \geq \frac{1}{d}\frac{1}{(1-2\lambda)}.$$

Then there is an algorithm $\mathcal{A}$ which runs in time

$$O\left(m^4 d^{4m} n^{m+1} + m^6\right) \text{ for } m = O(\ln(n/\epsilon)),$$

such that with probability at least $\frac{3}{4} + \frac{1}{25}$ over the randomness of the 2D (μ, n, d)-Gaussian tensor network T, it outputs a value $\hat{\chi}(T)$ that approximates $\chi(T)$ with ϵ-multiplicative error. That is

$$|\hat{\chi}(T) - \chi(T)| \leq \epsilon|\chi(T)|.$$

Note that if a random variable $X \sim \mathcal{N}_{\mathbb{C}}(\mu, 1)$ with $\mu > 0$, then $\frac{1}{\mu}X \sim \mathcal{N}_{\mathbb{C}}(1, \frac{1}{\mu^2})$. That is

$$\frac{1}{\mu}X = 1 + \frac{1}{\mu}Y \text{ for } Y \sim \mathcal{N}_{\mathbb{C}}(0, 1).$$

Thus approximating 2D (μ, n, d)-Gaussian tensor networks with multiplicative error can be reduced to approximating 2D (z, n, d)-shifted-Gaussian tensor networks for $z = \frac{1}{\mu}$.

In other words, to prove Theorem 123 it suffices to show that one can approximate 2D (z, n, d)-shifted-Gaussian tensor networks $T_A(z)$ for $0 \leq z \leq d \cdot (1 - 2\lambda)$, where

$$z_* = \frac{1}{\mu} \leq d \cdot (1 - 2\lambda)$$

gives the contraction value, that is $\chi(T) = \mu^n \cdot \chi(T_A(z_*))$.

Specify parameters in the algorithm

We use the algorithm 7 in Section 5.4 to prove Theorem 123. Recall that λ is a parameter in Theorem 123. We specify the parameters in the input of the algorithm as:

- $\kappa = 4$ since the degree of a 2D lattice of periodic boundary condition is 4.
- $\mu_v = 1$ for all $v \in G$.
- $z_{end} = d \cdot (1 - 2\lambda)$.

- $\rho = \frac{\pi\lambda^4}{4(1-2\lambda)}$ for the strip $\mathcal{T}(1, 2\rho)$.

The key Lemma for proving Theorem 123 is showing that $G_A(z)$ has no roots in $\mathcal{T}(1, 2\rho)$ with high probability. We use the same notations $g_A(z), G_A(z)$ as defined in Section 5.4.

Lemma 124 (Root-free strip) *With probability at least $\frac{3}{4} + \frac{1}{25}$ over the randomness of A, $G_A(z)$ has no roots in $\mathcal{T}(1, 2\rho)$.*

Before proving Lemma 124 we first prove that Theorem 122 and Lemma 124 together imply Theorem 123.

Proof:[Proof of Theorem 123] By Lemma 124 with probability at least $\frac{3}{4} + \frac{1}{25}$ over the randomness of A, $G_A(z) \neq 0$ for $z \in \mathcal{T}(1, 2\rho)$. Then we prove Theorem 123 by Theorem 122. ∎

Rescaling polynomials. Recall that

$$g_A(z) = \chi(T_A(z)), \text{ for } 0 \leq z \leq d \cdot (1 - 2\lambda) \tag{5.39}$$

$$G_A(z) = \chi(T_A(z \cdot d \cdot (1 - 2\lambda))), \text{ for } 0 \leq z \leq 1. \tag{5.40}$$

For convenience, we also define

$$h_A(z) = \chi(T_A(zd)), \text{ for } 0 \leq z \leq 1 - 2\lambda. \tag{5.41}$$

Then Lemma 124 is equivalent to the following.

Lemma 125 (Root-free strip) *With probability at least $\frac{3}{4} + \frac{1}{25}$ over the randomness of A, $h_A(z)$ has no roots in $\mathcal{T}(1 - 2\lambda, w)$ for $w = \pi\lambda^4/2$.*

The following Sections are used to prove Lemma 125. In Section 5.5 we review the 2D Ising model. In Section 5.5 we map the random 2D tensor network to the 2D Ising model. In Section 5.5 we analyze the partition function of the 2D Ising model and use it to find the root-free strip. Finally in Section 5.5 we prove the exact contraction of random 2D tensor network with positive mean remains **#P-hard**.

2D Ising model

This section is a review of the 2D Ising model. Let L_1 and L_2 be two integers where

$$n = L_1 \times L_2.$$

For simplicity we assume L_2 is even. Consider an $L_1 \times L_2$ 2D lattice with periodic boundary conditions, meaning that the lattice can be embedded onto a torus. We assume the periodic boundary condition to simplify the analysis.

Denote the lattice as $G = (V, E)$ and let $n \coloneqq |V|$. At each vertex v, there is a spin which takes a value $s_v \in \{-1, +1\}$. The Hamiltonian (or energy function) of the *2D Ising model* is defined to be the function H mapping a spin configuration $s \coloneqq \{s_v\}_{v \in V}$ to its energy

$$H(s, \mathcal{J}, \varrho) = -\mathcal{J} \sum_{(v,w) \in E} s_v s_w - \varrho \sum_{v \in V} s_v, \tag{5.42}$$

where $\mathcal{J} \in \mathbb{R}$ is the pair-wise interaction strength and $\varrho \in \mathbb{R}$ quantifies the strength of an external magnetic field. The partition function at inverse temperature β is defined as

$$\mathcal{Z}(\beta, \mathcal{J}, \varrho) = \sum_{s \in \{\pm 1\}^n} \exp(-\beta H(s, \mathcal{J}, \varrho)) \tag{5.43}$$

$$= \sum_{s \in \{\pm 1\}^n} \prod_{(v,w) \in E} \exp(\beta \mathcal{J} s_v s_w) \cdot \prod_{v \in V} \exp(\beta \varrho s_v). \tag{5.44}$$

It is well known that when there is no external magnetic field, that is $\varrho = 0$, the partition function of the 2D Ising model with periodic boundary has a closed form. In the thermodynamic limit, this formula is known as Onsager's solution [Ons44]. For a finite-size lattice, a refined formula has been given by Kaufman [Kau49], which is summarized in Lemma 126. There is no closed form formula for the partition function $\mathcal{Z}(\beta, \mathcal{J}, \varrho)$ when $\varrho \neq 0$.

Lemma 126 ([Kau49]) *The partition function of the 2D Ising model on an $L_1 \times L_2$ lattice with periodic boundary conditions and zero magnetic fields is given by*

$$\mathcal{Z}(\beta, \mathcal{J}, 0) = \frac{1}{2} (2 \sinh 2\beta\mathcal{J})^{L_1 L_2/2} \times \left\{ \prod_{r=1}^{L_2} \left(2 \cosh \frac{L_1}{2} \gamma_{2r} \right) + \prod_{r=1}^{L_2} \left(2 \sinh \frac{L_1}{2} \gamma_{2r} \right) \right. \tag{5.45}$$

$$\left. + \prod_{r=1}^{L_2} \left(2 \cosh \frac{L_1}{2} \gamma_{2r-1} \right) + \prod_{r=1}^{L_2} \left(2 \sinh \frac{L_1}{2} \gamma_{2r-1} \right) \right\}, \tag{5.46}$$

where for $j = 1, \ldots, 2L_2$, we define

$$\cosh \gamma_j \coloneqq \cosh 2H^* \cdot \cosh 2\beta\mathcal{J} - \sinh 2H^* \cdot \sinh 2\beta\mathcal{J} \cdot \cos(j\pi/L_2), \tag{5.47}$$

$$\tanh H^* \coloneqq \exp(-2\beta\mathcal{J}). \tag{5.48}$$

Notice that Eq. (5.47) does not specify the sign of γ_j. Since here we are only interested in an upper bound of $\mathcal{Z}(\beta, \mathcal{J}, 0)$, we can just assume that γ_j has a positive sign.[5]

Mapping random tensor networks to the Ising model

In this section, we estimate $E_A|h_A(z)|^2$ by mapping it to the partition function of a 2D Ising model. To this end, observe that choosing $\beta, \mathcal{J}$ such that $\exp(\beta\mathcal{J}) = d^2$ and $\exp(-\beta\mathcal{J}) = d\sqrt{d}$, we can write

$$\mathcal{Z}(\beta, \mathcal{J}, 0) = \sum_{s\in\{\pm 1\}^n} R(s) \tag{5.49}$$

in terms of a function

$$R(s) = \prod_{(v,w)\in E} r_{vw}(s), \tag{5.50}$$

with weights

$$r_{vw}(s) = \begin{cases} d\sqrt{d}, & \text{if } s_u \neq s_v, \\ d^2, & \text{if } s_u = s_v. \end{cases} \tag{5.51}$$

We then show the following lemma.

Lemma 127 *For $z \in \mathbb{C}$, $|z| > 0$, set $\beta, \mathcal{J}, \varrho$ in the 2D Ising model to satisfy*

$$\beta\mathcal{J} = \frac{\ln d}{4}, \quad \beta\varrho = \ln|z|. \tag{5.52}$$

Then we have that over the randomness of A, we have that

$$E_A|h_A(z)|^2 = d^{7n/2}|z|^n \mathcal{Z}(\beta, \mathcal{J}, \varrho) \tag{5.53}$$

$$= \sum_{s\in\{\pm 1\}^n} R(s)|z|^{2|s|}. \tag{5.54}$$

Note that in Lemma 127, the 2D Ising model has a non-zero magnetic field $\varrho \neq 0$, and thus the closed form formula for 2D Ising model without magnetic fields (Lemma 126) does not directly apply.

[5] For readers who are interested in numerically verifying Lemma 126, the sign of γ_j influences the value of $\sinh \frac{L_1}{2}\gamma_j$ and thus the value of $\mathcal{Z}(\beta, \mathcal{J}, 0)$. The sign of γ_j is explained in Remark 15 and Figure 3 of [Kau49]: $\gamma_j \geq 0$ for all $j \neq 2n$; but γ_{2n} is negative if $\beta\mathcal{J} < H_c$, and is non-negative if $\beta\mathcal{J} \geq H_c$, where H_c is the critical point which is approximately 0.4407. A remark is [Kau49] denotes our γ_{2n} as γ_0.

In the remainder of this section, we prove Lemma 127. Here we use the techniques of mapping random instances to classical statistical mechanical models, which are widely used in the physics literature for studying phase transitions [BCA20; SRN19; Yan+22; LC21]. This section will heavily use the graphical representations of tensor networks, which was explained in Section 5.2.

Recall that in the 2D tensor network $T_A(zd)$, for each vertex v, the tensor $M^{[v]}$ can be written as

$$M^{[v]} = J^{[v]} + zd \cdot A^{[v]}. \tag{5.55}$$

In the following Lemma 128, we first compute the expectation of the product of tensor $M^{[v]}$ and its conjugate, that is $M^{[v]} \otimes \overline{M^{[v]}}$. Evaluating this average will allow us to compute $E_A|h_A(z)|^2$, since $h_A(z) \cdot \overline{h_A(z)}$ is the product of two 2D tensor networks where, for each vertex we can pair the tensors as $M^{[v]} \otimes \overline{M^{[v]}}$, as we will explain in detail in the proof of Lemma 127.

Lemma 128 *Define the delta tensor T^δ to be a tensor of rank 2 with free edges i, i' and bond dimension d, where $(T^\delta)_{ii'} = \delta_{ii'}$. As in Figure 5.2 , we have*

$$E_A\left[A^{[v]} \otimes \overline{A^{[v]}}\right] = T^\delta \otimes T^\delta \otimes T^\delta \otimes T^\delta := \left(T^\delta\right)^{\otimes 4}, \tag{5.56}$$

$$E_A\left[M^{[v]} \otimes \overline{M^{[v]}}\right] = J^{[v]} \otimes J^{[v]} + |z|^2 (d^{1/2} \cdot T^\delta)^{\otimes 4}, \tag{5.57}$$

where in Figure 5.2 (a) we use $\square$ to represent the vertex for a delta tensor T^δ, and in Figure 5.2 (b) we use $\bullet$ to represent the vertex for the tensor $[1, 1, 1 \ldots, 1]$.

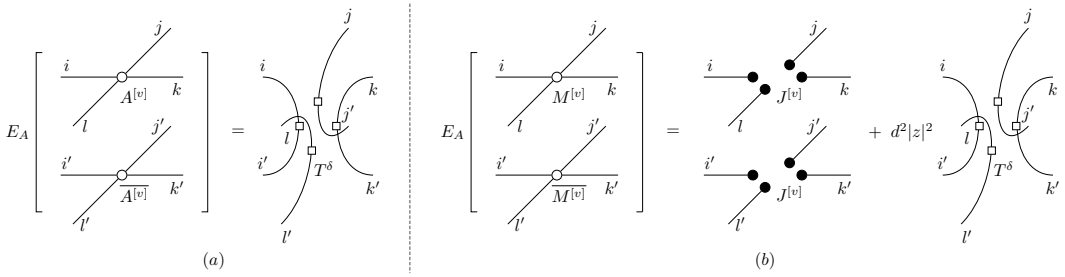


Figure 5.2: Graphical representation of the expectation of tensor products. (a) The expectation of the product of $A^{[v]}$ and its conjugate is a product of delta tensors. (b) The expectation of the product $M^{[v]} \otimes \overline{M^{[v]}}$ decomposes into a linear combination of delta tensors and a product of rank-1 tensors.

Proof: As in Figure 5.2 (a), we label the free edges of the first copy of $A^{[v]}$ by i, j, k, l, and the free edges of the second copy by i', j', k', l'. By definition, all

entries of $A^{[v]}$ are sampled independently from $\mathcal{N}_{\mathbb{C}}(0,1)$. Thus, we have that

$$\left(E_A\left[A^{[v]} \otimes \overline{A^{[v]}}\right]\right)_{ijkl,i'j'k'l'} = \delta_{ii'} \cdot \delta_{jj'} \cdot \delta_{kk'} \cdot \delta_{ll'}, \tag{5.58}$$

which proves Eq. (5.56). To prove Eq. (5.57), it suffices to notice that

$$M^{[v]} = J^{[v]} + zd \cdot A^{[v]}. \tag{5.59}$$

$$E_A\left[J^{[v]} \otimes \overline{A^{[v]}}\right] = E_A\left[A^{[v]} \otimes \overline{J^{[v]}}\right] = 0. \tag{5.60}$$

■

Proof:[Proof of Lemma 127]

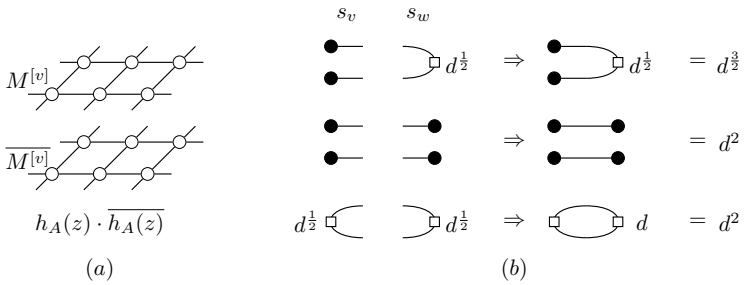


Figure 5.3: (a) Pair the tensors $M^{[v]}$ and its conjugate in $E_A|g_A(z)|^2$. (b) The tensors with respect to edge (v,w) in $T(s)$. From top to button, the value of (s_v, s_w) are $(1,-1), (1,1), (-1,-1)$ respectively.

As in Figure 5.3 (a), $h_A(z) \cdot \overline{h_A(z)}$ is the product of two 2D tensor network, where for each vertex v, we can pair the tensors as $M^{[v]} \otimes \overline{M^{[v]}}$. By Lemma 128 we know that

$$N^{[v]} \coloneqq E_A\left[M^{[v]} \otimes \overline{M^{[v]}}\right] = J^{[v]} \otimes J^{[v]} + |z|^2 (d^{1/2} \cdot T^{\delta})^{\otimes 4}. \tag{5.61}$$

Define a new 2D tensor network $T(N)$ where at each vertex v the tensor is $N^{[v]}$. Notice that $E_A|h_A(z)|^2 = \chi(T(N))$ since

$$E_A|h_A(z)|^2 = E_A\left(\sum_{c\in[d]^{|E|}} \prod_{v\in V} M_c^{[v]}\right)\left(\sum_{c\in[d]^{|E|}} \prod_{v\in V} \overline{M_c^{[v]}}\right) \tag{5.62}$$

$$= \sum_{c,c'\in[d]^{|E|}} \prod_{v\in V} E_A\left(M^{[v]} \otimes \overline{M^{[v]}}\right)_{c,c'} \tag{5.63}$$

$$= \chi(T(N)). \tag{5.64}$$

We map $\chi(T(N))$ to the partition function of the 2D Ising model as follows. For any configuration $s \coloneqq \{s_v\}_{v\in V}$, $s_v \in \{\pm 1\}$, construct a new 2D tensor network $T(s)$ as follows:

- If $s_v = 1$ we set the tensor on v to be $J^{[v]} \otimes J^{[v]}$;
- If $s_v = -1$ we set the tensor on v to be $(d^{1/2} \cdot T^{\delta})^{\otimes 4}$.

As in the top figure in Figure 5.3 (b), for an edge (v, w), if $(s_v, s_w) = (1, -1)$, then the edge (v, w) contributes a scalar factor as $d^{\frac{3}{2}}$ to $T(s)$, which is the contraction value of the tensor $J^{[v]} \otimes J^{[v]}$ and the tensor $(d^{1/2} \cdot T^{\delta})^{\otimes 4}$. Similarly, as in Figure 5.3 (b), if $(s_v, s_w) = (1, 1)$ or $(-1, -1)$, the edge (v, w) contributes a scalar factor as d^2. Thus one can check that the contraction value of $T(s)$ is given by $R(s)$.

With an arbitrary ordering of the n vertices, we write the configuration $s = \{s_v\}_{v \in V}$ as a vector $s \in \{\pm 1\}^n$. Based on Eq. (5.61), one can compute $\chi(T(N))$ by expanding $N^{[v]}$, that is for any configuration s,

- We use $s_v = 1$ to represent choosing $J^{[v]} \otimes J^{[v]}$,
- We use $s_v = -1$ for $|z|^2 \cdot (d^{1/2} \cdot T^{\delta})^{\otimes 4}$.

Then define $|s|$ to be the number of -1 in s, we have

$$\chi(T(N)) = \sum_{s \in \{\pm 1\}^n} \chi(T(s)) |z|^{2|s|} = \sum_{s \in \{\pm 1\}^n} R(s) |z|^{2|s|}. \tag{5.65}$$

One can check that setting $\beta \mathcal{J} = \frac{\ln d}{4}$ and $\beta \varrho = \ln |z|$, for any $s \in \{\pm 1\}^n$, we have

$$d^{7n/2} |z|^n \exp(-\beta H(s, \mathcal{J}, \varrho)) = R(s) |z|^{2|s|}. \tag{5.66}$$

■

Finding a root-free strip

In this section, we show that one can efficiently find a root-free strip with high probability. In particular, we will bound $E_A[|h_A(z)|^2]$ and use Lemma 119.

Recall that we consider a 2D lattice with periodic boundary conditions, where the 2D lattice has size $n = L_1 \times L_2$ and L_2 is even. The 2D Ising model and $R(s)$ are defined in Section 5.5.

The exact formula for the partition function $\mathcal{Z}(\beta, \mathcal{J}, 0)$ in Lemma 126 is intimidating. We upper bound $\mathcal{Z}(\beta, \mathcal{J}, 0)$ by a simpler formula. Then we will use this formula to bound $E_A[|h_A(z)|^2]$.

Lemma 129 (Bound on the partition function with no magnetic field) *If* $\beta\mathcal{J} = 1/4 \cdot \ln d$ *and* $d \geq 3$*, we have that*

$$2d^{\frac{n}{2}} \leq \mathcal{Z}(\beta, \mathcal{J}, 0) \leq 2d^{\frac{n}{2}} \left(1 + \frac{3}{d}\right)^n . \tag{5.67}$$

Proof: *Here we use the same notation as in Lemma 126. By the definition of the partition function, and the fact that there are* $2n$ *edges in the 2D square lattice with periodic boundary conditions, we have*

$$\mathcal{Z}(\beta, \mathcal{J}, 0) \geq \sum_{s=00..0 \text{ or } 11...1} \prod_{(v,w)\in E} \exp(\beta\mathcal{J} s_v s_w) = 2d^{n/2}. \tag{5.68}$$

Now, note that we can rewrite the definitions in Lemma 126, for any j *as*

$$2\cosh\gamma_j = \cosh 2H^* \cdot 2\cosh 2\beta\mathcal{J} \cdot (1 - \alpha\cos(j\pi/L_2)) , \tag{5.69}$$

where $\alpha \coloneqq \tanh 2H^* \cdot \tanh 2\beta\mathcal{J} \in (0, 1)$*. Since* L_2 *is even, we have*

$$\prod_{r=1}^{L_2}(1 - \alpha\cos(2r\pi/L_2)) = \prod_{r=1}^{L_2/2} (1 - \alpha\cos(2r\pi/L_2)) \left(1 - \alpha\cos(2(r + \frac{L_2}{2})\pi/L_2)\right) \tag{5.70}$$

$$= \prod_{r=1}^{L_2/2} (1 - \alpha\cos(2r\pi/L_2)) (1 + \alpha\cos(2r\pi/L_2)) \tag{5.71}$$

$$= \prod_{r=1}^{L_2/2} \left(1 - \alpha^2\cos^2(2r\pi/L_2)\right) \tag{5.72}$$

$$\leq 1. \tag{5.73}$$

By the definition of cosh, $2\cosh kx \leq (2\cosh x)^k$ *for any* x *and any integer* $k \geq 0$*, we have that*

$$\prod_{r=1}^{L_2}\left(2\cosh\frac{L_1}{2}\gamma_{2r}\right) \leq \left(\prod_{r=1}^{L_2} 2\cosh\gamma_{2r}\right)^{L_1/2} \tag{5.74}$$

$$= (\cosh 2H^* \cdot 2\cosh 2\beta\mathcal{J})^{L_1L_2/2} \left(\prod_{r=1}^{L_2}(1 - \alpha\cos(2r\pi/L_2))\right)^{L_1/2} \tag{5.75}$$

$$\leq (\cosh 2H^* \cdot 2\cosh 2\beta\mathcal{J})^{L_1L_2/2} , \tag{5.76}$$

where the second equality comes from Eq. (5.69) and the last inequality comes from Eq. (5.73). Similarly we can get the same upper bound for

$$\prod_{r=1}^{L_2}\left(2\cosh\frac{L_1}{2}\gamma_{2r-1}\right), \text{ and } \prod_{r=1}^{L_2}\left(2\sinh\frac{L_1}{2}\gamma_{2r}\right), \prod_{r=1}^{L_2}\left(2\sinh\frac{L_1}{2}\gamma_{2r-1}\right),$$

where we get the bound for the last two terms by $|\sinh x| \leq \cosh x$. *Besides, from* $\tanh H^* = \exp(-2\beta\mathcal{J})$ *we have*

$$\exp(2H^*) = \frac{\exp(2\beta\mathcal{J})+1}{\exp(2\beta\mathcal{J})-1} = \frac{\sqrt{d}+1}{\sqrt{d}-1} \tag{5.77}$$

$$\cosh 2H^* = \frac{1}{2}\left(\frac{\sqrt{d}+1}{\sqrt{d}-1} + \frac{\sqrt{d}-1}{\sqrt{d}+1}\right) \leq 1+\frac{3}{d}, \text{ for } d \geq 3. \tag{5.78}$$

We estimate

$$2\sinh 2\beta\mathcal{J} = \sqrt{d} - \frac{1}{\sqrt{d}} \leq \sqrt{d}, \tag{5.79}$$

$$2\cosh 2\beta\mathcal{J} = \sqrt{d} + \frac{1}{\sqrt{d}}. \tag{5.80}$$

Thus by Lemma 126, we finally conclude that

$$\mathcal{Z}(\beta,\mathcal{J},0) \leq \frac{1}{2}\cdot d^{\frac{n}{4}}\cdot 4\cdot\left(1+\frac{3}{d}\right)^{\frac{n}{2}} d^{\frac{n}{4}}\left(1+\frac{1}{d}\right)^{\frac{n}{2}} \tag{5.81}$$

$$\leq 2d^{\frac{n}{2}}\left(1+\frac{3}{d}\right)^{n}. \tag{5.82}$$

■

Using Lemma 129 we can now estimate $\sum_s R(s)$.

Lemma 130

$$\sum_{s\in\{\pm1\}^n} R(s) \leq 2d^{4n}(1+\frac{3}{d})^n \tag{5.83}$$

Proof: Since the 2D lattice we consider has n vertices and satisfies the periodic boundary condition, there are in total $2n$ edges. Set $z = 1, \beta\mathcal{J} = 1/4\cdot\ln d$, $\beta\varrho = \ln|z| = 0$, (that is $\varrho = 0$). By Lemma 127 and Lemma 129 we have that

$$\sum_{s\in\{\pm1\}} R(s) = d^{7n/2}\mathcal{Z}(\beta,\mathcal{J},0) \leq 2d^{4n}\left(1+\frac{3}{d}\right)^{n}. \tag{5.84}$$

■

Then we estimate $E_A|h_A(z)|^2$ for small z and for $z \leq 1$. For completeness we also give a lower bound on $E_A|h_A(z)|^2$ in Lemma 131 (c). (c) will not be used in other proofs.

Lemma 131 *Let c and ρ be two constants where $0 < \rho < 1$. Assume $d \geq nc^{-1}$.We have*

(a) For $|z| \leq \rho$, $E_A|h_A(z)|^2 \leq d^{4n}(1 + 2\rho^2 e^{3c})$.

(b) For $|z| \leq 1$, $E_A|h_A(z)|^2 \leq d^{4n} \cdot 2e^{3c}$.

(c) For any z, $E_A|h_A(z)|^2 \geq d^{4n}(1 + \frac{|z|^2}{d^4})^n$.

Proof: Note that when $s = 0 \ldots 0$, $R(s) = d^{4n}$. For (a), since $|z| \leq \rho$, by Lemma 127 we have

$$E_A|h_A(z)|^2 = d^{4n} \cdot |z|^0 + \sum_{s\in\{\pm 1\}^n : |s|\geq 1} R(s)|z|^{2s} \tag{5.85}$$

$$\leq d^{4n} + \rho^2 \sum_{s\in\{\pm 1\}^n} R(s) \tag{5.86}$$

$$\leq d^{4n} + \rho^2 \cdot 2d^{4n}\left(1 + \frac{3}{d}\right)^n \tag{5.87}$$

$$\leq d^{4n}(1 + 2\rho^2 e^{3c}), \tag{5.88}$$

where the second inequality comes from $|z| \leq \rho < 1$ and $R(s) \geq 0, \forall s$; the third inequality comes from Lemma 130; and the last inequality comes from $d \geq n \cdot c^{-1}$.

For (b), by Lemma 127 and Lemma 130 we have for $|z| \leq 1$,

$$E_A|h_A(z)|^2 \leq \sum_{s\in\{\pm 1\}^n} R(s) \cdot 1 \tag{5.89}$$

$$\leq d^{4n} 2\left(1 + \frac{3}{d}\right)^n \tag{5.90}$$

$$\leq d^{4n} \cdot 2e^{3c}. \tag{5.91}$$

For (c), note that for $|s| = k$, since there are at most $4k$ edges (v, w) in $R(s)$ which take values $r_{vw}(s) = d\sqrt{d}$, we must have $R(s) \geq d^{4n}/\sqrt{d}^{4k}$. Thus

$$E_A|h_A(z)|^2 = \sum_{k=0}^{n} \sum_{s\in\{\pm 1\}^n : |s|=k} R(s)|z|^{2k}. \tag{5.92}$$

$$\geq \sum_{k=0}^{n} \binom{n}{k} d^{4n} \left(\frac{|z|^2}{d^2}\right)^k \tag{5.93}$$

$$= d^{4n}\left(1 + \frac{|z|^2}{d^2}\right)^n. \tag{5.94}$$

■

Recall that $N_A(r)$ is the number of roots of $h_A(z)$ inside the disk $\mathcal{B}(r)$. With Lemma 131, we can estimate $N_A(r)$ by using Lemma 119.

Corollary 132 *Suppose $d \geq nc^{-1}$ for some constant c. Let λ be an arbitrary small constant satisfying $0 \leq \lambda \leq e^{-3c}/80$. We have*

$$Pr_A\left[N_A(\lambda) = 0 \quad \& \quad N_A(1-\lambda) \leq \frac{1}{\lambda^2}\right] \geq 4/5. \tag{5.95}$$

Proof: Note that

$$h_A(0) = d^{2n}.$$

We have

$$\begin{aligned} E_A[N_A(\lambda)] &\leq E_A\left[N_A\left(2\lambda - 2\lambda\cdot\lambda/2\right)\right] && (5.96)\\ &\leq \frac{1}{\lambda}\ln E_\theta E_A \frac{|h_A(2\lambda\cdot e^{i\theta})|^2}{|h_A(0)|^2} && (5.97)\\ &\leq \frac{1}{\lambda}\cdot\ln(1+8\lambda^2 e^{3c}) && (5.98)\\ &\leq 8\lambda\cdot e^{3c}, && (5.99) \end{aligned}$$

where the first inequality comes from the fact that the disk of radius $2\lambda - 2\lambda\cdot\lambda/2$ contains the disk of radius λ; the second inequality comes from Lemma 119; and the third inequality comes from Lemma 131 (a) by setting $\rho = 2\lambda$.

Similarly from Lemma 119 and Lemma 131 (b) we get

$$\begin{aligned} E_A\left[N_A\left(1-\lambda\right)\right] &\leq \frac{1}{2\lambda}\ln E_\theta E_A \frac{|h_A(e^{i\theta})|^2}{|h_A(0)|^2} && (5.100)\\ &\leq \frac{1}{2\lambda}\cdot\ln(2e^{3c}). && (5.101) \end{aligned}$$

Then by Markov inequality, we have

$$\begin{aligned} Pr_A\left[N_A(\lambda) \geq 1\right] &\leq E_A\left[N_A(\lambda)\right] && (5.102)\\ &\leq 8\lambda\cdot e^{3c} && (5.103)\\ &\leq \frac{1}{10}. && (5.104) \end{aligned}$$

$$\begin{aligned} Pr_A\left[N_A\left(1-\lambda\right) \geq \frac{1}{\lambda^2}\right] &\leq \lambda^2\cdot E_A\left[N_A(1-\lambda)\right] && (5.105)\\ &\leq \lambda/2\cdot\ln(2e^{3c}) && (5.106)\\ &\leq \frac{1}{10}. && (5.107) \end{aligned}$$

Then we get the desired result by a union bound. ■

Finally we prove Lemma 125.

Proof:[Proof of Lemma 125] Let

$$M \coloneqq 1/\lambda^3, \quad \theta \coloneqq 2\pi/M, \quad w \coloneqq \pi\lambda^4/2. \tag{5.108}$$

For simplicity we assume that $1/\lambda$ is an integer. Consider the disk $\mathcal{B}(1-\lambda)$, that is the disk centered at 0 and of radius $1-\lambda$. As in Figure 5.4, we divide $\mathcal{B}(1-\lambda)$ into M disjoint circular sectors, where for each sector the central angle is θ. Inside each sector which is indexed by $k \in [M]$, we consider a strip of width w, that is

$$\mathcal{T}_k \coloneqq \mathcal{T}((1-2\lambda)e^{ik\theta}, w).$$

Note that since $\sin x \geq x/2$ for $0 \leq x \leq \pi/3$, we have

$$\lambda \sin(\theta/2) \geq \lambda\theta/4 = \pi\lambda^4/2 = w. \tag{5.109}$$

Thus all the M strips $\{\mathcal{T}_k\}_{k=1}^M$ are disjoint outside $\mathcal{B}(\lambda)$. Besides, one can check that the end part of the strip $\mathcal{T}_k$ is inside the k-th sector by noticing

$$0 \leq \lambda \leq 1/80 \Rightarrow (1-2\lambda)\tan\frac{\theta}{2} \geq w.$$

Denote S_{good} as the set of tensors A which have no roots in the disk of radius λ and few roots in the disk of radius $1-\lambda$:

$$S_{good} \coloneqq \{A : \quad N_A(\lambda) = 0 \ \&\ N_A(1-\lambda) \leq \frac{1}{\lambda^2}\}.$$

By Corollary 132, we know that

$$Pr_A\left[A \in S_{good}\right] \geq 4/5. \tag{5.110}$$

In the following we argue that S_{good} can be further partitioned into disjoint subsets, where in each subset, with probability at least $(1-\lambda)$ over the randomness of A there are no roots in $\mathcal{T}(1-2\lambda, w)$.

To this end, first we observe that for any $k \in [M]$, by the definition of $h_A(\cdot)$, we have

$$h_{e^{ik\theta}A}(z) = h_A(e^{ik\theta}z). \tag{5.111}$$

Note that $e^{ik\theta}z$ is just a rotation of z in the complex plane. By the rotational symmetry of disks, the roots of $h_{e^{ik\theta}A}(z) = h_A(e^{ik\theta}z)$ are simply rotated compared to the roots of $h_A(z)$. Thus if $A \in S_{good}$, then so is $e^{ik\theta}A$.

Next, we partition S_{good} into disjoint subsets in the way that A, A' are in the same subset iff there exists $k \in [M]$ such that $A' = e^{ik\theta}A$. For convenience, for each subset we fix an arbitrary A as the representative and write the subset as

$$S_{good}(A) = \{e^{ik\theta}A | \, k \in [M]\}.$$

By the definition of S_{good}, for any $A \in S_{good}$, there are no roots in $\mathcal{B}(\lambda)$ and there are at most $1/\lambda^2$ roots in $\mathcal{B}(1-\lambda)$. Since the M tubes $\{\mathcal{T}_k\}_{k=1}^M$ are disjoint outside $\mathcal{B}(\lambda)$, there is at most a $\frac{1/\lambda^2}{M} = \lambda$ fraction of the M tubes which contains roots of $h_A(z)$. Further, recall that

$$h_{e^{ik\theta}A}(z) = h_A(e^{ik\theta}z),$$

and thus the tube $\mathcal{T}_0$ with respect to to $e^{ik\theta}A$ corresponds to the tube $\mathcal{T}_k$ with respect to A. Hence, there is at most a λ fraction of $A' \in S_{good}(A)$ such that the corresponding strip $\mathcal{T}_0$ contains roots of $h_A(z)$.

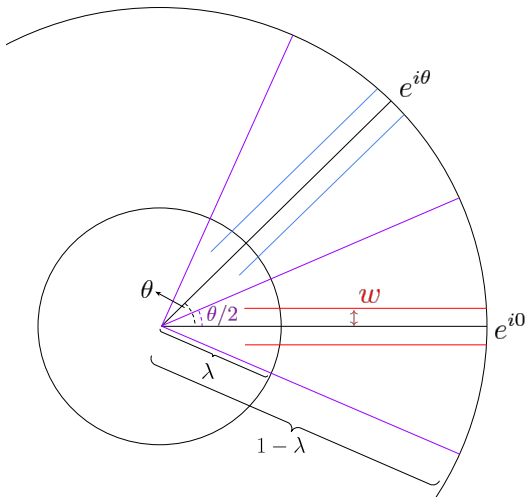


Figure 5.4: Illustration of dividing the circle into M disjoint circular sectors. The radius of the small disk and the big disk is λ and $1-\lambda$ respectively. We divide the big disk $\mathcal{B}(1-\lambda)$ into M circular sectors. In each sector we choose a strip of width w. The first strip $\mathcal{T}_0$ starts from $-w$ and ends at $1-2\lambda+w$. Other strips are rotations of $\mathcal{T}_0$. All the strips are disjoint outside the small disk $\mathcal{B}(\lambda)$.

In summary, we conclude that the fraction of A such that there are no roots in $\mathcal{T}(1-2\lambda, w)$ is greater than

$$\frac{4}{5} \cdot (1-\lambda) \geq \frac{3}{4} + \frac{1}{25}.$$

■

#P-hardness of exact contraction

Finally we prove that the exact contraction of the random 2D tensor network with a positive mean remains **#P-hard**. The proof is a simple adaption of Theorem 1 and Theorem 3 in [Haf+20]. For completeness, we put a proof in Appendix 5.7.

To make the statement rigorous, here we consider the finite precision approximation of the Gaussian distribution, denoted as $\overline{\mathcal{N}}_{\mathbb{C}}(\mu, \sigma^2)$, where each sample can be represented by finite bits instead of being an arbitrary real or complex number. For example here we set the $\overline{\mathcal{N}}_{\mathbb{C}}(\mu, \sigma^2)$ to be the distribution where each sample $z \sim \overline{\mathcal{N}}_{\mathbb{C}}(\mu, \sigma^2)$ is obtained by firstly sampling y according to Gaussian distribution $\mathcal{N}_{\mathbb{C}}(\mu, \sigma^2)$, and then setting z to be the value by rounding y to n^2 bits. $\overline{\mathcal{N}}_{\mathbb{C}}(\mu, \sigma^2)$ behaves similarly as $\mathcal{N}_{\mathbb{C}}(\mu, \sigma^2)$ but makes the statements of exact contraction and proofs more rigorous.

Accordingly, we consider finite precision 2D (μ, n, d)-Gaussian tensor network instead of 2D (μ, n, d)-Gaussian tensor network, where we substitute $\mathcal{N}_{\mathbb{C}}(\mu, \sigma^2)$ by $\overline{\mathcal{N}}_{\mathbb{C}}(\mu, \sigma^2)$.

Theorem 133 (#P-hard) *For any $\mu \in [0, poly(n)]$, $n \geq 25$ and $d = O(poly(n))$, if there exists an algorithm $\mathcal{A}$ which runs in $poly(n)$ time and with probability at least $\frac{3}{4} + \frac{1}{n}$ over the randomness of the finite precision 2D (μ, n, d)-Gaussian tensor network T, it outputs the exact value of $\chi(T)$, then there exists an algorithm which runs in randomized $poly(n)$ time and solves **#P-complete** problems.*

5.6 Approximating arbitrary positive tensor networks

In previous sections we have considered approximating random tensor networks. In this section we move to the task of contracting a fixed tensor network.

For a general tensor network $T = T(G, M)$, computing the contraction value $\chi(T)$ *exactly* is known to be **#P-hard** [Sch+07]. On the other hand, Arad and Landau [AL10] proved that *approximating* $\chi(T)$ up to an inverse polynomial additive error in the matrix 2-*norm* is **BQP-complete**.

In this section, we focus on *positive tensor network* $T = T(G, M)$. These are defined by tensors $\{M^{[v]}\}_v$ the entries of which are all non-negative. The main part of this section will establish that when T is a positive tensor network, approximating $\chi(T)$ up to an inverse polynomial additive error in the matrix 1-*norm* is **BPP-complete**.

Then, in Section 5.6 we give a short proof showing that approximating positive tensor network with inverse-poly multiplicative error is at least **StoqMA-hard**. Section 5.6 is self-contained and can be read independently. We first review the swallowing algorithm for tensor network contraction. Then we explain Arad and Landau's **BQP-c**ompleteness result and our **BPP-c**ompleteness result, which are both based on the swallowing algorithm.

A swallowing algorithm and notations

Recall that in Section 5.2 we have introduced two operations on tensor networks, taking their product and contraction. Given a tensor network $T = T(G, M)$, the swallowing algorithm (Algorithm 8) is a standard method to exactly compute the contraction value $\chi(T)$, by contracting edges of tensors $\{M^{[v]}\}_v$ according to the graph $G = (V, E)$.

Algorithm 8 The swallowing algorithm

1: Given an ordering of vertex $v_1, \ldots, v_n$ of G.
2: Set $i \leftarrow 1$. Set the current tensor $A[i]$ to be the tensor $M^{[v_1]}$, which can be pictured as one vertex and some free edges as in Figure 5.1 (a). $i \leq n$ {Adding tensor $M^{[v_{i+1}]}$ to $A[i]$} $M^{[v_{i+1}]}$ and $A[i]$ share edges in G
3: Construct a new tensor $A[i+1]$ by contraction, i.e. identifying the shared edges and summing over the corresponding indices.
4: $M^{[v_{i+1}]}$ and $A[i]$ has no common edge.
5: Then $A[i+1]$ is defined as the product $M^{[v_{i+1}]} \otimes A[i]$.
6: $i \leftarrow i+1$
7: $\chi(T) \leftarrow A[n]$

High level ideas for Arad and Landau's result and our result. Let us first describe Arad and Landau's result at a high level before writing down formal statements with heavy notations. As in Figure 5.5, given an arbitrary ordering to the vertices, for every vertex v_i, we implicitly partition the free edges of $M^{[v_i]}$ into input and output edges. With respect to this partition of in-edges and out-edges, one can write $M^{[v_i]}$ as a matrix denoted as $M^{[v_i]in,out}$. As in Algorithm 8, the contraction value of the tensor network is then given by sequentially mapping the in-edges to out-edges, which can be represented by the matrix multiplication $\prod_i M^{[v_i]in,out} \otimes I_{else}$, where I_{else} denotes the free edges other than the input edges in $A[i-1]$. Arad and Landau's result shows that this matrix multiplication can be simulated by a quantum circuit through embedding each matrix $M^{[v_i]in,out}$ into a unitary, where the embedding is done by adding an ancillary qubit. Our result is, when every $M^{[v_i]in,out}$ is a positive

matrix, instead of embedding it into a unitary, we embed the positive matrix into a stochastic matrix and simulate positive matrix multiplication with a random walk.

To explain Arad and Landau's result formally, we define more notation which is used in the swallowing algorithm. This notation is adapted from [AL10].

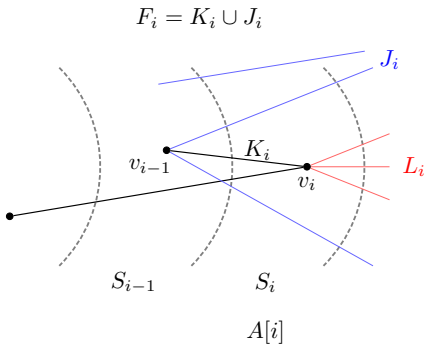


Figure 5.5: Illustration of the notations. F_i is the edges connecting $\{v_1, ..., v_{i-1}\}$ and $\{v_i, ..., v_n\}$. The edges attached to v_i are partitioned into the in-edges K_i and out-edges L_i. When contracting the tensor $M^{[v_i]}$ we map the in-edges to out-edges. The edges in F_i but not in K_i are called J_i.

As in Figure 5.5, define:

- $S_i = \{v_1, \ldots, v_i\}$.
- F_i is the set of edges which connect S_{i-1} and V/S_{i-1}. F_i are the free edges in tensor $A[i-1]$.
- K_i is the set of edges which connects S_{i-1} and v_i. K_i are the edges being contracted when contracting $A[i-1]$ and $M^{[v_i]}$. Note that $K_1 = \emptyset$.
- $J_i \coloneqq F_i/K_i$, which are the free edges in both $A[i-1]$ and $A[i]$. $J_1 = \emptyset$.
- L_i is the set of edges of v_i which are not in K_i. In other words, L_i are the new free edges introduced by adding tensor $M^{[v_i]}$ to $A[i-1]$.

Denote edges in K_i as $\{e_1^{K_i}, \ldots, e_{|K_i|}^{K_i}\}$. Denote edges in J_i and L_i similarly. With some abuse of notations, we use $e_s^{K_i}$ to denote both the name of the edge and the colors in $[d]$ that the edge $e_s^{K_i}$ takes.

In the following, we explain that the update from tensors $A[i-1]$ to $A[i]$ can be written as matrix multiplication. More specifically:

- First note that $A[i]$ can be viewed as a column vector consisting of $d^{|F_{i+1}|}$ entries, where the entries are indexed by the free edges of $A[i]$, that is

$$F_{i+1} = K_{i+1} \cup J_{i+1} = L_i \cup J_i.$$

More specifically, write this vector as $|A[i]\rangle \in \mathbb{C}^{d^{|F_{i+1}|}}$, then for edges in J_i, L_i taking colors as $e_1^{J_i}, \ldots, e_{|J_i|}^{J_i}, e_1^{L_i}, \ldots, e_{|L_i|}^{L_i}$ where $e_s^{J_i}, e_t^{L_i} \in [d]$, we define

$$\left\langle e_1^{J_i}, \ldots, e_{|J_i|}^{J_i}, e_1^{L_i}, \ldots, e_{|L_i|}^{L_i} \middle| A[i] \right\rangle \coloneqq A[i]_{e_1^{J_i}, \ldots, e_{|J_i|}^{J_i}, e_1^{L_i}, \ldots, e_{|L_i|}^{L_i}}. \tag{5.112}$$

- Note that $M^{[v_i]}$ can be viewed as a matrix: when adding $M^{[v_i]}$ to $A[i-1]$ in Line 3 of Algorithm 8, we contract the free edges in K_i and introducing new free edges L_i. One can view $M^{[v_i]}$ as a mapping from $\mathbb{C}^{d^{|K_i|}}$ to $\mathbb{C}^{d^{|L_i|}}$, denoted as $M^{[v_i]K_iL_i}$, which can be written as a matrix of size $d^{|L_i|} \times d^{|K_i|}$ where

$$\left\langle e_1^{L_i}, \ldots, e_{|L_i|}^{L_i} \middle| M^{[v_i]} \middle| e_1^{K_i}, \ldots, e_{|K_i|}^{K_i} \right\rangle = M^{[v_i]}_{e_1^{L_i}, \ldots, e_{|L_i|}^{L_i}, e_1^{K_i}, \ldots, e_{|K_i|}^{K_i}} \tag{5.113}$$

In particular, since $K_1 = \emptyset$, $M^{[v_i]K_iL_i}$ is a column vector in $\mathbb{C}^{d^{|L_1|}}$. For convenience, define $|A[0]\rangle$ to be a scalar,

$$|A[0]\rangle \coloneqq 1.$$

Denote I_{J_i} as the identity operator on the indices with respect to edges in J_i.

- One can check that the updates from $A[i-1]$ to $A[i]$ can be written as matrix multiplication, that is for $i = 1, \ldots, n$,

$$|A[i]\rangle = I_{J_i} \otimes M^{[v_i]K_iL_i} |A[i-1]\rangle, \tag{5.114}$$

in the sense that

$$\begin{aligned} &\left\langle e_1^{J_i}, \ldots, e_{|J_i|}^{J_i}, e_1^{L_i}, \ldots, e_{|L_i|}^{L_i} \middle| A[i] \right\rangle \\ &= \sum_{e_1^{K_i}, \ldots, e_{|K_i|}^{K_i}} \left\langle e_1^{L_i}, \ldots, e_{|L_i|}^{L_i} \middle| M^{[v_i]} \middle| e_1^{K_i}, \ldots, e_{|K_i|}^{K_i} \right\rangle \left\langle e_1^{K_i}, \ldots, e_{|K_i|}^{K_i}, e_1^{J_i}, \ldots, e_{|J_i|}^{J_i} \middle| A[i-1] \right\rangle. \end{aligned} \tag{5.115}$$

In particular, $|A[n]\rangle$ is a scalar which equals to the contraction value. Thus we have

$$|A[n]\rangle = \prod_{i=1}^{n} I_{J_i} \otimes M^{[v_i]K_iL_i} = \chi(T). \tag{5.116}$$

To ease notations we define the *swallowing operator* as

$$O^{[v_i]} \coloneqq I_{J_i} \otimes M^{[v_i]K_iL_i}. \tag{5.117}$$

BPP-completeness of additive-error approximation

According to the discussion in the previous section, one can compute $\chi(T)$ exactly by updating $|A[i]\rangle$ according to Eq. (5.114). It is well known that computing $\chi(T)$ exactly is **#P-hard** even for a constant degree graph G, thus one cannot efficiently perform the exact version of the update Eq. (5.114). However, interestingly [AL10] showed that one can approximately perform the update efficiently using a quantum computer, where the approximation refers to an inverse polynomial additive error in the 2-norm of the tensors.

Theorem 134 (Additive 2-norm approximation of tensor networks [AL10]) *Let $G = (V, E)$ be an n-vertex graph of constant degree. Let $T(G, M)$ be a tensor network on G with bond dimension $d = O(poly(n))$. The following approximation problem is **BQP-complete**: given as input*

- *a tensor network $T(G, M)$, and a precision parameter $\epsilon = 1/poly(n)$, and*
- *an ordering of the vertices $v_1, \ldots, v_n$, and the corresponding swallowing operators $O^{[v_i]}$ defined in Eq. (5.117),*

output a complex number $\hat{\chi}(T)$ such that

$$Pr\left(|\chi(T) - \hat{\chi}(T)| \leq \epsilon \Delta_2\right) \geq 3/4, \tag{5.118}$$

where

$$\Delta_2 \coloneqq \prod_{i=1}^{n} \|O^{[v_i]}\|_2. \tag{5.119}$$

Note that by Eq. (5.117), both the 2-norm $\|\cdot\|_2$ and 1-norm of $\|\cdot\|_1$ of $O^{[v_i]}$ equal to the corresponding norm of $M^{[v_i]K_iL_i}$.

We prove that if the tensor network T is a positive tensor network, then instead of using a quantum computer, we can approximate the update Eq. (5.114) efficiently using a classical computer, where the approximation refers to an inverse polynomial additive error in the matrix 1-norm of the matrices $O^{[v_i]}$.

Theorem 135 (Additive 1-norm approximation of positive tensor networks) *Let $G = (V, E)$ be an n-vertex graph of constant degree. Let $T(G, M)$ be a positive tensor network on G with bond dimension $d = O(poly(n))$. The following approximation problem is **BPP-complete**: given as input*

- *a positive tensor network* $T(G, M)$*, and a precision parameter* $\epsilon = 1/poly(n)$*, and*
- *an ordering of vertices* $v_1, \ldots, v_n$*, and the corresponding swallowing operators* $O^{[v_i]}$ *defined in Eq.* (5.117)*,*

output a complex number $\hat{\chi}(T)$ *such that*

$$Pr\left(|\chi(T) - \hat{\chi}(T)| \leq \epsilon\Delta_1\right) \geq 3/4, \tag{5.120}$$

where

$$\Delta_1 \coloneqq \prod_{i=1}^{n} \|O^{[v_i]}\|_1. \tag{5.121}$$

Proof of Theorem 135

The proof of the **BPP-hardness** part for Theorem 135 is similar to Section 4.2 in [AL10]. For completeness we give a proof sketch in Appendix 5.9. In the following we prove the "inside **BPP**" part of Theorem 135, that is, we provide an efficient classical algorithm that achieves Eq. (5.120). The main idea of the algorithm is to simulate non-negative matrix multiplication via a stochastic process.

We say that a matrix is *non-negative* if all of its entries are non-negative. We first give a lemma which extends a non-negative matrix to a stochastic matrix.

Lemma 136 *Let* $M \neq 0$ *be a non-negative matrix, that is* $M \in \mathbb{R}_{\geq 0}^{m\times n}$*. Then there exists* $N \in \mathbb{R}_{\geq 0}^{2m\times n}$ *such that* N *is a stochastic matrix*[6]*, and* $\frac{M}{\|M\|_1}$ *is the first* m *rows of* N*.*

Proof: Since $\|M\|_1 = \max_{1\leq j\leq n} \sum_{i=1}^{m} |M_{ij}|$, thus for each column, the column sum of $\frac{M}{\|M\|_1}$ lies in $[0, 1]$, thus one can embed $\frac{M}{\|M\|_1}$ as the first m rows in a stochastic matrix N. ∎

Before we state the **BPP** algorithm, we recall some notations. Recall that we have defined $F_i, J_i, K_i, L_i, A[i], M^{[v_i]K_iL_i}$ in Section 5.6 and Figure 5.5. To ease notation, we abbreviate $M^{[v_i]K_iL_i}$ as $M^{[v_i]}$. Since we are working with positive tensor network, $M^{[v_i]}$ is a non-negative matrix and the entries are indexed by K_i, L_i.

Lemma 136 says that we can embed $M^{[v_i]}/\|M^{[v_i]}\|_1$ in a stochastic matrix $N^{[v_i]}$ by adding one ancillary bit, that is $N^{[v_i]}$ is indexed by $L_i \cup \{w_i\}$ and K_i, where

[6] N is a stochastic matrix iff N is a non-negative matrix, and each column sums to 1.

$w_i \in \{0,1\}$ is an index such that $w_i = 0$ (or 1) refers to the first (or second) $d^{|L_i|}$ rows of $N^{[v_i]}$. Define

$$W_i \coloneqq \{w_1, \ldots, w_{i-1}\}. \tag{5.122}$$

We first explain the high level idea of the **BPP** algorithm, and then give the pseudo code. The idea of the **BPP** algorithm in Theorem 135 is to mimic non-negative matrix multiplication by stochastic methods. From a high level idea, we will embed the vector $|A[i]\rangle$ in a probability distribution, and embed the matrix $M^{[v_i]}$ in a stochastic matrix $N^{[v_i]}$ by adding an ancillary bit. Then the update rule

$$|A[i+1]\rangle = I_{J_i} \otimes M^{[v_i]K_iL_i} |A[i]\rangle, \tag{5.123}$$

is embedded in applying the stochastic matrix $N^{[v_i]}$ to the distribution $A[i]$, which can be simulated by a random walk.

Before writing down the pseudo codes we explain some notations.

- We use $|k\rangle_K$ to represent the (ordered) coloring $k \in [d]^{|K|}$ of the edges in K. When $K = \emptyset$ we view $|k\rangle_K$ as empty, that is writing down nothing. Similarly for $|j\rangle_J, |w\rangle_W$.
- We will use s_i to denote a computational basis whose distribution embed the vector $A[i-1]$.
- Recall that the free edges of $A[i-1]$ are $F_i = K_i \cup J_i$. Also recall that $F_{i+1} = J_i \cup L_i$.

The pseudo codes are stated as Algorithm 9 and Algorithm 10. Their performance is given in Corollary 138.

Lemma 137 *The probability of the Trial algorithm (Algorithm 9) to return "Success" is $\chi(T)/\Delta_1$ where $\Delta_1 \coloneqq \prod_{i=1}^{n} \|O^{[v_i]}\|_1$. Its runtime is $poly(n)$.*

Proof: To ease notation, for a set $Q \cup W$ where Q is a set of edges of G, and W is a set of ancillary indices $\{\ldots, w_i, \ldots\}$, we use

$$String(Q \cup W) \coloneqq [d]^{|Q|} \times \{0,1\}^{|W|},$$

where edges/indices in Q take values in $[d]$, and ancillary indices in W takes values in $\{0,1\}$.

Algorithm 9 Trial(T=T(G,M))

$i = 1$, $s_1 = \emptyset$. $i \le n$
Interpret $s_i = |k\rangle_{K_i} |j\rangle_{J_i} |w\rangle_{W_i}$. $\{K_1 = J_1 = W_1 = \emptyset.\}$
Recall that $N^{[v_i]}$ has rows indexed by $L_i \cup \{w_i\}$, column indexed by K_i.
Since $N^{[v_i]}$ is a stochastic matrix, each column of $N^{[v_i]}$ corresponds to a distribution over the row index. i=1
$N^{[v_1]}$ is a column vector. Denote the distribution according to this column as $\mathcal{D}$.
Denote the distribution according to the k-th column of $N^{[v_i]}$ as $\mathcal{D}$.
Sample a row index according to $\mathcal{D}$.
Denote this row index as $|lw'\rangle_{L_i \cup \{w_i\}}$, where $l \in [d]^{|L_i|}$, $w' \in \{0,1\}$.
Set $s_{i+1} \leftarrow |l\rangle_{L_i} |j\rangle_{J_i} |ww'\rangle_{W_{i+1}}$. $\{ N^{[v_i]}$ maps register K_i to $L_i \cup \{w_i\}.\}$
$i \leftarrow i + 1$.
Interpret $s_{n+1} = |w\rangle_{W_{n+1}}$ {We know $L_n = J_n = \emptyset$}
Return Success if $w = 00\ldots00$, that is the all zero state; otherwise return Fail.

Algorithm 10 Approximating positive tensor network

Set $K = 10\epsilon^{-2}$
Run the $Trial(T)$, that is Algorithm 9, for K times.
Count the number of Success as $\#Success$.
Return $\hat{\chi}(T) \leftarrow \frac{\#Success}{K} \cdot \Delta_1$ as the approximation of $\chi(T)$.

Define

$$S_i \coloneqq String(L_i \cup J_i \cup W_{i+1}),$$

define $|\pi_{i+1}\rangle$ as the probability distribution of s_{i+1}, which is a probability distribution over S_i. Note that

$$|\pi_2\rangle = I_{J_1} \otimes N^{[v_1]}, \text{ where } I_{J_1} = I_\emptyset = 1 \tag{5.124}$$

$$\begin{aligned} |\pi_{i+1}\rangle &= I_{J_i} \otimes N^{[v_i]} |\pi_i\rangle \\ &= \prod_{h=1}^{i} I_{J_h} \otimes N^{[v_h]} \qquad (5.125) \\ &= \prod_{h=1}^{i} \frac{I_{J_h} \otimes M^{[v_h]}}{\|M^{[v_h]}\|_1} |0\ldots0\rangle_{W_{i+1}} + \text{additional terms} \qquad (5.126) \\ &= \prod_{h=1}^{i} \frac{O^{[v_h]}}{\|O^{[v_h]}\|_1} |0\ldots0\rangle_{W_{i+1}} + \text{additional terms} \qquad (5.127) \end{aligned}$$

where Eq. (5.125) is from Algorithm 9; Eq. (5.126) is from the definition of $N^{[v_h]}$

that $N^{[v_h]}$ embeds $M^{[v_h]}$; and the last equality comes from definition of $O^{[v_h]}$,

$$O^{[v_h]} = I_{J_h} \otimes M^{[v_h]}, \tag{5.128}$$

$$\|O^{[v_h]}\|_1 = \|M^{[v]}\|_1. \tag{5.129}$$

From Eq. (5.116) we conclude

$$|\pi_{n+1}\rangle = \frac{\chi(T)}{\Delta_1} |0\dots0\rangle_{W_{n+1}} + \text{additional terms} \tag{5.130}$$

To prove the runtime is $poly(n)$, notice that in Theorem 135 we assume that G is a graph of constant degree, thus $|L_i|, |K_i|$ are constants, thus $d^{|L_i|}, d^{|K_i|}$ are $poly(n)$ whenever $d = poly(n)$ and $N^{[v_i]}$ is a matrix of size $poly(n) \times poly(n)$. Thus Line 7 in Algorithm 9 can be done efficiently. ∎

Corollary 138 *For $t = poly(n)$ sufficiently large, the output $\hat{\chi}(T)$ in Algorithm 10 satisfies*

$$Pr\left(|\chi(T) - \hat{\chi}(T)| \leq \epsilon\Delta_1\right) \geq 3/4, \tag{5.131}$$

where $\Delta_1 \coloneqq \prod_{i=1}^{n} \|O^{[v_i]}\|_1$.

Proof: The proof directly follows from Eq. (5.130) and Chebyshev's inequality. Write X_i as the result of the i-th trial, where $X_i = 1$ if the Trial algorithm (Algorithm 9) returns Success and = 0 otherwise. By Lemma 137, we have

$$E[X_i] = \chi(T)/\Delta_1.$$

Besides, note that $|\chi(T)/\Delta_1| \leq 1$ since it corresponds to a probability, we have

$$|Var(X_i)| \leq E|X_i|^2 + |E[X_i]|^2 \leq E|X_i|^2 + |\chi(T)/\Delta_1|^2 \leq 2. \tag{5.132}$$

Define $X \coloneqq (X_1 + \ldots + X_K)/K$, we have

$$E(X) = \frac{\chi(T)}{\Delta_1}, \tag{5.133}$$

$$Var(X) = \frac{1}{K} Var(X_1) \leq \frac{2}{K}. \tag{5.134}$$

By definition of $\hat{\chi}(T)$ in Algorithm 10, use Chebyshev's inequality we have

$$Pr\left(|\chi(T) - \hat{\chi}(T)| \geq \epsilon\Delta_1\right) = Pr\left(|\frac{\chi(T)}{\Delta_1} - X| \geq \epsilon\right) \tag{5.135}$$

$$\leq \frac{Var(X)}{\epsilon^2} \tag{5.136}$$

$$\leq \frac{2}{K\epsilon^2} \tag{5.137}$$

$$= 1/5. \tag{5.138}$$

■

StoqMA-hardness of multiplicative-error approximation

In this section, we consider the task of approximating a positive tensor network up to a multiplicative error. We show that this approximation is **StoqMA-hard** up to exponentially close to 100% error.

Theorem 139 *Let $G = (V, E)$ be an n-vertex graph of constant degree. Let $T(G, M)$ be a positive tensor network on G with bond dimension $d = poly(n)$. Consider the following approximation problem: given as inputs a tensor network $T(G, M)$ and a precision parameter $\epsilon \leq 1 - \exp(-n)$, output a complex number $\hat{\chi}(T)$ such that*

$$Pr\left(|\chi(T) - \hat{\chi}(T)| \leq \epsilon|\chi(T)|\right) \geq 3/4. \tag{5.139}$$

*If there exists a $poly(n)$-time randomized algorithm for solving the above approximation problem, then there exists a $poly(n)$-time randomized algorithm for solving **StoqMA** with probability greater than $3/4$.*

Note that $\epsilon \leq 1 - \exp(-n)$ means we allow very large (close to 100%) multiplicative error. Recall that **StoqMA** is a subclass of **QMA** which is related to deciding ground energy for stoquastic Hamiltonians [BBT06]. For our purpose, we use an equivalent definition of **StoqMA** that makes use of the notion of a stoquastic verifier.

Definition 140 (StoqMA, from [BBT06]) *A stoquastic verifier is a tuple $V = (n, n_w, n_0, n_+, U)$, where*

- *n is the number of input bits, n_w is the number of input witness qubits.*
- *n_0 is the number of input ancillas $|0\rangle$, n_+ is the number of input ancillas $|+\rangle$.*
- *U is a quantum circuit on $n + n_w + n_0 + n_+$ qubits with X, CNOT, and Toffoli gates.*

The acceptance probability of a stoquastic verifier V on input string $x \in \Sigma^n$ and witness state $|\psi\rangle \in (\mathbb{C}^2)^{n_w}$ is defined as

$$Pr(V; x, \psi) = \langle\psi_{in}|U^\dagger \Pi_{out} U|\psi_{in}\rangle, \tag{5.140}$$

$$where\ |\psi_{in}\rangle = |x\rangle \otimes |\psi\rangle \otimes |0\rangle^{\otimes n_0} \otimes |+\rangle^{\otimes n_+}, \tag{5.141}$$

$$\Pi_{out} = |+\rangle\langle+|_1 \otimes I_{else}. \tag{5.142}$$

*A promise problem $L = L_{yes} \cup L_{no} \subseteq \Sigma^*$ belongs to **StoqMA** iff there exists a uniform family of stoquastic verifier V which uses at most $poly(n)$ qubits and gates, and obeys the following:*

- *Completeness. If $x \in L_{yes}$, then there exists $|\psi\rangle$ such that $Pr(V; x, \psi) \geq b$,*
- *Soundness. If $x \in L_{no}$, then for any $|\psi\rangle$ we have $Pr(V; x, \psi) \leq a$,*

where $0 \leq a < b \leq 1$ and $b - a \geq 1/poly(n)$.

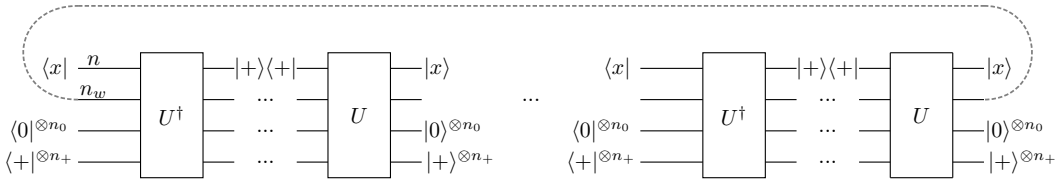


Figure 5.6: Represent $tr(M_x^2)$ as a tensor network. The above figure contains two copies of M_x. When connecting the right side of the first M_x and the left side of the second M_x, we get the operator M_x^2. Further connecting the left side of the first M_x and the right side of the second M_x by the dashed line, we get $tr(M_x^2)$.

The proof of Theorem 139 is adapted from the folklore proof of **QMA** $\subseteq$ **PP**, where the adaption is mainly translating matrix operations (multiplication, trace, etc) to tensor network operations. We only give a proof sketch here.

Proof:[Proof of Theorem 139] In this proof we use the notions in Definition 140. Consider a language $L = L_{yes} \cup L_{no}$ in **StoqMA** with stoquastic verifier V in Definition 140. For input x, as pictured in Figure 5.6 we define a positive semi-definite Hermitian operator acting on n_w qubits as

$$M_x := \langle\phi|U^\dagger \Pi_{out} U|\phi\rangle, \tag{5.143}$$

$$\text{where } |\phi\rangle = |x\rangle \otimes |0\rangle^{\otimes n_0} \otimes |+\rangle^{\otimes n_+}. \tag{5.144}$$

Denote the maximum eigenvalue of M_x as $\lambda_{\max}(x)$. By assumption we have that

- If $x \in L_{yes}$, then $\lambda_{\max}(x) \geq b$.
- If $x \in L_{No}$, then $\lambda_{\max}(x) \leq a$.

Since M_x is positive semi-definite, then for any $k \in \mathbb{N}$ we have

- If $x \in L_{yes}$, then $tr(M_x^k) \geq b^k$.

- If $x \in L_{No}$, then $tr(M_x^k) \leq 2^{n_w} a^k$.

Recall that $\epsilon \leq 1 - \exp(-n)$ is the precision parameter. By the assumption in Theorem 139, there is a $poly(n)$-time randomized algorithm $\mathcal{A}$ such that with probability at least $3/4$, the algorithm returns

$$(1-\epsilon)|\chi(T)| \leq |\hat{\chi}(T)| \leq (1+\epsilon)|\chi(T)|. \tag{5.145}$$

To distinguish the yes and no cases, set

$$k \geq \left(n_w \ln 2 + \ln \frac{1+\epsilon}{1-\epsilon} \right) / \ln \frac{b}{a} = poly(n).$$

We have

$$2^{n_w} a^k (1+\epsilon) < b^k (1-\epsilon). \tag{5.146}$$

Thus one can distinguish whether $x \in L_{yes}$ or $x \in L_{no}$ by approximating $tr(M_x^k)$ using the algorithm $\mathcal{A}$.

It remains to explain that $tr(M_x^k)$ can be represented by a positive tensor network $T = T(G, M)$ with $poly(n)$ bond dimension, where G is a $poly(n)$-vertex graph of constant degree.

First notice that similarly as Section 5.9 or Section 4.2 in [AL10], one can naturally represent M_x as a tensor network $T = T(G, M)$ with $poly(n)$ bond dimension. Since the gates in U are X, CNOT, and Toffoli, and the ancillas are computational basis or $|+\rangle$, one can check that $T(G, M)$ is a positive tensor network. Further, since U has $poly(n)$ gates and each gate has constant number of input qubits and output qubits, we have that G is a $poly(n)$-vertex graph of constant degree.

To represent $tr(M_x^k)$ as a tensor network, as in Figure 5.6, it suffices to additionally notice that

- The tensor network for the operator M_x^2 can be represented by putting 2 copies of M_x in a line, then connecting the right side of the first M_x and the left side of the second M_x, that is contracting the free edges w.r.t register n_w for the first and second copy.
- Similarly as explained in Eq. (5.7) in Section 5.2, if we further connect the left side of the first M_x and the right side of the second M_x by the dashed line, we get $tr(M_x^2)$.

- The tensor network for the operator $tr(M_x^k)$ can be represented similarly, that is, putting k copies of M_x in a line, and then contracting the free edges w.r.t register n_w sequentially.

■

Acknowledgements

We thank Garnet Chan and Zeph Landau for helpful discussions. Part of this work was conducted while the authors were visiting the Simons Institute for the Theory of Computing during summer 2023 and spring 2024, supported by DOE QSA grant #FP00010905. D.H. acknowledges financial support from the US DoD through a QuICS Hartree fellowship. N.S. acknowledges financial support by the Austrian Science Fund FWF (Grant DOIs 10.55776/COE1 and 10.55776/F71) and the European Union's Horizon 2020 research and innovation programme through Grant No. 863476 (ERC-CoG SEQUAM). Jiaqing Jiang is supported by MURI Grant FA9550-18-1-0161 and the IQIM, an NSF Physics Frontiers Center (NSF Grant PHY-1125565). Jielun Chen is supported by the US National Science Foundation under grant CHE-2102505.

5.7 Appendix:#P-hardness of exactly contracting random 2D tensor networks

Here we prove that the exact contraction of the random 2D tensor network with a positive mean remains #**P**-**hard**.

Firstly we prove some properties of standard Gaussian distribution, while the finite precision Gaussian distribution behaves similarly up to $O(\exp(-n))$ derivation in the error bounds. Recall that we use $X \sim \mathcal{N}_{\mathbb{C}}(\mu, \sigma^2)$ (or $X \sim \mathcal{N}_{\mathbb{R}}(\mu, \sigma^2)$) to denote that the random variable X is sampled from the complex (or real) Gaussian distribution with mean μ and standard derivation σ. We use $\vec{X} = (X_1, X_2, \ldots, X_m) \sim \prod_{i=1}^m \mathcal{N}_{\mathbb{C}}(\mu_i, \sigma^2)$ to denote the random variable $\vec{X}$ where each X_i is independently sampled from $\mathcal{N}_{\mathbb{C}}(\mu_i, \sigma^2)$. When $\mu_i = \mu, \forall i$, we abbreviate the notation as $\vec{X} \sim \mathcal{N}_{\mathbb{C}}(\mu, \sigma^2)^m$. For two distribution $\mathcal{D}_1, \mathcal{D}_2$, we use $\|\mathcal{D}_1 - \mathcal{D}_2\|$ to denote the total variation distance.

Lemma 141 (Analogy of Lemma 5 in [Haf+20]) [7] *For $\mu_i \in \mathbb{C}$. It holds that*

$$\|\mathcal{N}_{\mathbb{C}}(\mu, (1-\epsilon)^2\sigma^2)^m - \mathcal{N}_{\mathbb{C}}(\mu, \sigma^2)^m\| \leq 4m\epsilon, \tag{5.147}$$

$$\|\prod_{i=1}^{m} \mathcal{N}_{\mathbb{C}}(\mu_i, \sigma^2) - \mathcal{N}_{\mathbb{C}}(\mu, \sigma^2)^m\| \leq \frac{2}{\sigma}\left(|\mu_1 - \mu| + \ldots + |\mu_m - \mu|\right). \tag{5.148}$$

Proof: Recall that for $\mu \in \mathbb{C}$, we use $\Re(\mu), \Im(\mu) \in \mathbb{R}$ for the real and imaginary part of μ, that is $\mu = \Re(\mu) + \Im(\mu)i$. Besides, $X \sim \mathcal{N}_{\mathbb{C}}(\mu, \sigma^2)$ iff $\Re(X) \sim \mathcal{N}_{\mathbb{R}}(\Re(\mu), \frac{\sigma^2}{2})$, $\Im(X) \sim \mathcal{N}_{\mathbb{R}}(\Im(\mu), \frac{\sigma^2}{2})$. It suffices to notice that

$$\|\prod_{i=1}^{m} \mathcal{N}_{\mathbb{C}}(\mu_i, \sigma_i^2)^m - \mathcal{N}_{\mathbb{C}}(\mu, \sigma^2)^m\| \tag{5.149}$$

$$= \frac{1}{2}\int_{(x_1,\ldots,x_m)} \left| \prod_{i=1}^{m} \frac{1}{2\pi(\sigma_i/\sqrt{2})^2} \exp^{-\frac{1}{2}|\frac{x_i-\mu_i}{\sigma_i/\sqrt{2}}|^2} - \prod_{i=1}^{m} \frac{1}{2\pi(\sigma/2)^2} \exp^{-\frac{1}{2}|\frac{x_i-\mu}{\sigma/2}|^2} \right| dx_1 \ldots dx_m \tag{5.150}$$

$$= \frac{1}{2}\int_{(x_1,\ldots,x_m)} \left| \prod_{i=1}^{m} \frac{1}{2\pi(\sigma_i/\sqrt{2})^2} \exp^{-\frac{1}{2}|\frac{x_i-\mu_i+\mu}{\sigma_i/\sqrt{2}}|^2} - \prod_{i=1}^{m} \frac{1}{2\pi(\sigma/2)^2} \exp^{-\frac{1}{2}|\frac{x_i}{\sigma/2}|^2} \right| dx_1 \ldots dx_m \tag{5.151}$$

$$= \|\prod_{i=1}^{m} \mathcal{N}_{\mathbb{C}}(\mu_i - \mu, \sigma_i^2)^m - \mathcal{N}_{\mathbb{C}}(0, \sigma^2)^m\|. \tag{5.152}$$

Thus by Lemma 5 in [Haf+20], we have

$$\|\mathcal{N}_{\mathbb{C}}(\mu, (1-\epsilon)^2\sigma^2)^m - \mathcal{N}_{\mathbb{C}}(\mu, \sigma^2)^m\| = \|\mathcal{N}_{\mathbb{C}}(0, (1-\epsilon)^2\sigma^2)^m - \mathcal{N}_{\mathbb{C}}(0, \sigma^2)^m\| \tag{5.153}$$

$$\leq 2 \times 2m\epsilon \tag{5.154}$$

$$\|\prod_{i=1}^{m} \mathcal{N}_{\mathbb{C}}(\mu_i, \sigma^2) - \mathcal{N}_{\mathbb{C}}(\mu, \sigma^2)^m\| = \|\prod_{i=1}^{m} \mathcal{N}_{\mathbb{C}}(\mu_i - \mu, \sigma^2) - \mathcal{N}_{\mathbb{C}}(0, \sigma^2)^m\| \tag{5.155}$$

$$\leq 2 \times \frac{1}{\sigma}\left(|\mu_1 - \mu| + \ldots + |\mu_m - \mu|\right), \tag{5.156}$$

where we add a $2\times$ since we are working with $\mathcal{N}_{\mathbb{C}}$ while Lemma 5 in [Haf+20] is with $\mathcal{N}_{\mathbb{R}}$. ∎

[7] There is a remark on the notation difference. [Haf+20] uses $\mathcal{N}_{\mathbb{C}}(\mu, \sigma)$ to denote Gaussian distribution with mean value μ and standard derivation σ. In this manuscript we denote this distribution as $\mathcal{N}_{\mathbb{C}}(\mu, \sigma^2)$ which is the more standard notation.

Proof:[Proof of Theorem 133] The proof follows directly from the proof idea of Theorem 2 in [Haf+20]. Here we only give a proof sketch.

Firstly [Haf+20; Sch+07] showed that one can encode any n-variable boolean function $f(x_1, \ldots, x_n)$ into a projected entangled-pair states (PEPS) of $poly(n)$ vertices, which describes an un-normalized state $|\psi\rangle$, such that computing $\langle\psi|\psi\rangle$ exactly is equivalent to computing the value

$$s(f) = |\{x \in \{0,1\}^n : f(x) = 1\}|,$$

which is **#P-complete**. One can check that in this case $\langle\psi|\psi\rangle$ equals to the contraction value of a 2D[8] tensor network of $poly(n)$ vertices, where the 2D tensor network has bond dimension $d = O(poly(n))$, and every entry of the tensor network is bounded by a constant. one can further make the underlying 2D lattice for the 2D tensor network to have periodic boundary condition, by adding edges connecting boundaries and slightly modify the tensors near the boundary to make sure the contraction value remains invariant. Denote the final 2D lattice with periodic boundary condition as G. Denote the final 2D tensor network which encodes the fixed boolean function f as

$$T\left(G, (P^{[v]})_v\right),$$

where $P^{[v]}$ is the tensor on vertex v. Note that $P^{[v]}$ has d^4 entries and thus $(P^{[v]})_v$ are described by in total $d^4 \times n$ entries. For convenience, we assign an arbitrary order to those entries and denoted them as $\{p_i\}_{i=1}^{d^4 n}$. Recall that by construction we have $|p_i| \leq c$ for constant c.

Theorem 133 is proved by an argument of average to worse case reduction via interpolation. Here we define the polynomial for the interpolation. Set

$$\epsilon := \min\left\{\frac{1}{4(c+\mu+1)d^4 n^3}, \frac{1}{2}\right\}. \tag{5.157}$$

$$k = poly(n) \text{ be sufficiently large.} \tag{5.158}$$

$$\text{Let } S = \{t_i\}_{i\in[k]} \text{ be the set of } k \text{ equidistant points in } [0, \epsilon]. \tag{5.159}$$

Recall that $0 \leq u \leq poly(n)$ thus $\epsilon = 1/poly(n)$.

We randomly sample a 2D (μ, n, d)-Gaussian tensor network $T\left(G, (Q^{[v]})_v\right)$. Let $t \in S$, define a new 2D tensor network $T(t)$, where for vertex v the tensor $R(t)^{[v]}$ is

[8] $\langle\psi|\psi\rangle$ is a stack of two PEPS. One can transform it into a 2D tensor network by contracting the stack of two PEPS via free boundaries of the two PEPS.

defined as

$$R(t)^{[v]} \coloneqq tP^{[v]} + (1-t)Q^{[v]}. \tag{5.160}$$

Denote the exact contraction value of $T(t)$ as $q(t)$. Note that $q(t)$ is a degree-n polynomial of t. Besides, from construction we know that computing $q(1)$ will solve **#P-complete** problem.

In the following, we show that if one can compute the exact contraction value of finite precision 2D Gaussian tensor network with high probability, then we can compute $q(1)$ with high probability by interpolation. More specifically, for input $T(t)$, denote the value returned by the algorithm $\mathcal{A}$ in Theorem 133 as $\mathcal{A}(t)$.

(i) First we prove that since t is small, $\mathcal{A}(t)$ is a good approximation of $q(t)$. Specifically, define

$$\mu_i = tp_i + (1-t)u. \tag{5.161}$$

By Eq. (5.160), we know that the entries of $\left(R(t)^{[v]}\right)_v$ are sampled from distribution

$$\mathcal{D} \coloneqq \prod_{i=1}^{d^4n} \overline{\mathcal{N}}_{\mathbb{C}}(\mu_i, (1-t)^2). \tag{5.162}$$

Since $\overline{\mathcal{N}}_{\mathbb{C}}$ approximates $\mathcal{N}_{\mathbb{C}}$ within exponential precision, we know that

$$\|\mathcal{D} - \overline{\mathcal{N}}_{\mathbb{C}}(\mu, 1)^{d^4n}\| \leq O(\exp(-n)) + \|\prod_{i=1}^{d^4n} \mathcal{N}_{\mathbb{C}}(\mu_i, (1-t)^2) - \mathcal{N}_{\mathbb{C}}(\mu, 1)^{d^4n}\|, \tag{5.163}$$

where by Lemma 141 we have

$$\begin{aligned}
\|\prod_{i=1}^{d^4n} \mathcal{N}_{\mathbb{C}}(\mu_i, (1-t)^2) - \mathcal{N}_{\mathbb{C}}(\mu, 1)^{d^4n}\| &\leq \|\prod_{i=1}^{d^4n} \mathcal{N}_{\mathbb{C}}(\mu_i, (1-t)^2) - \mathcal{N}_{\mathbb{C}}(\mu, (1-t)^2)^{d^4n}\| \\
&\quad + \|\mathcal{N}_{\mathbb{C}}(\mu, (1-t)^2)^{d^4n} - \mathcal{N}_{\mathbb{C}}(\mu, 1)^{d^4n}\| \\
&\leq \frac{2}{(1-t)}\left(|\mu_1 - \mu| + \ldots |\mu_{d^4n} - \mu|\right) + 4 \cdot d^4n \cdot t, \\
&\leq 4 \cdot d^4n \cdot (c+\mu)\epsilon + 4 \cdot d^4n \cdot \epsilon, \qquad (5.164) \\
&\leq 4 \cdot d^4n \cdot (c+\mu+1)\epsilon \\
&\leq \frac{1}{n^2}, \qquad (5.165)
\end{aligned}$$

where Eq. (5.164) comes from the fact that

$$t \leq \epsilon \leq 1/2, \tag{5.166}$$

$$|\mu_i - \mu| = |t(p_i - \mu)| \leq (c + \mu)\epsilon. \tag{5.167}$$

Eqs. (5.163)(5.165) together imply

$$\|\mathcal{D} - \overline{\mathcal{N}}_{\mathbb{C}}(\mu, 1)^{d^4 n}\| \leq O(\exp(-n)) + \frac{1}{n^2}. \tag{5.168}$$

In other words, for any $t_i \in S$, the distribution of $\left(R(t)^{[v]}\right)_v$ is almost the same as the finite precision 2D (μ, n, d)-Gaussian tensor network. Let n and $k = poly(n)$ be sufficiently large. By Eq. (5.168) and the assumption of the performance of $\mathcal{A}$ we have

$$Pr\left(\mathcal{A}(t_i) = q(t_i)\right) \geq \frac{3}{4} + \frac{1}{n} - O(\exp(-n)) - \frac{1}{n^2} \geq \frac{3}{4} + \frac{1}{n^2}, \tag{5.169}$$

$$E|\{i : \mathcal{A}(t_i) = q(t_i)\}| \geq \left(\frac{3}{4} + \frac{1}{n^2}\right) k, \tag{5.170}$$

where in the second inequality E refers to expectation. By Chernouff bound we know that for sufficiently large $k = poly(n)$,

$$Pr\left(|\{i : \mathcal{A}(t_i) = q(t_i)\}| \geq \frac{k+n}{2}\right) \geq 1 - \exp(-n). \tag{5.171}$$

(*ii*) We then use the following theorem to recover the degree n polynomial $q(t)$

Theorem 142 (Berlekamp-Welch [Mov18]) *Let q be a degree-n polynomial over any field $\mathbb{F}$. Suppose we are given k pairs of elements $\{(x_i, y_i)\}_{i=1}^k$ with all x_i distinct, and with the promise that $y_i = q(x_i)$ for at least $\max(n+1, (k+n)/2)$ points. Then, one can recover q exactly in $poly(k, n)$ deterministic time.*

In Theorem 142 let

$$x_i \coloneqq t_i, \tag{5.172}$$

$$y_i \coloneqq \mathcal{A}(t_i). \tag{5.173}$$

Then by Eq. (5.171) we can recover $q(t)$ with probability $1 - \exp(-n)$ in $poly(n)$-time.

(*iii*) Finally, we have $q(t)$ in hand, which is correct with probability $1 - \exp(-n)$. Since $q(t)$ is a degree n polynomial, we can easierly compute $q(1)$, which solves a **#P-complete** problem.

■

5.8 Appendix:More on Barvinok's method

Proof:[Proof of Lemma 119] The proof uses Jensen's formula and follows the idea from [EM18]. Let $z_1, \ldots, z_j, \ldots$ be the roots of $h_A(z)$, Jensen's formula establishes the connection between the roots in the disk $\mathcal{B}(r)$, and the average of $\ln |h_A(z)|$ on the boundary of $\mathcal{B}(r)$:

$$\sum_{|z_j|\le r} \ln \frac{r}{|z_j|} + \ln |h_A(0)| = E_\theta \ln |h_A(re^{i\theta})|. \tag{5.174}$$

First notice that

$$\sum_{|z_j|\le r} \ln \frac{r}{|z_j|} \ge \sum_{|z_j|\le r(1-\lambda)} \ln \frac{r}{|z_j|} \tag{5.175}$$

$$\ge \sum_{|z_j|\le r(1-\lambda)} \ln \frac{r}{r(1-\lambda)} \tag{5.176}$$

$$\ge \lambda \cdot N_A(r(1-\lambda)), \tag{5.177}$$

where in the last inequality we use $\ln \frac{1}{(1-\lambda)} \ge \lambda$ for $\lambda \le 1/2$. Thus by Eqs. (5.177)(5.174), we have

$$E_A\left[N_A(r-r\lambda)\right] \le \frac{1}{\lambda} E_\theta E_A \ln \frac{|h_A(re^{i\theta})|}{|h_A(0)|} \tag{5.178}$$

$$= \frac{1}{2\lambda} E_\theta E_A \ln \frac{|h_A(re^{i\theta})|^2}{|h_A(0)|^2} \tag{5.179}$$

$$\le \frac{1}{2\lambda} \ln E_\theta E_A \frac{|h_A(re^{i\theta})|^2}{|h_A(0)|^2}. \tag{5.180}$$

where the last inequality holds since ln is a concave function. ∎

Lemma 143 (Derivatives of composite function) *Let $G(z)$ and $\phi(z)$ be two functions satisfying $\phi(0) = 0$. Let m be an integer. Suppose the first m derivatives $\{G^{(k)}(0)\}_{k=0}^m$ and $\{\phi^{(k)}(0)\}_{k=0}^m$ can be computed in time $t(n)$ where n is a parameter. Then the first m derivatives of the composite function $G(\phi(z))$ at $z = 0$, denoted as*

$$\left\{ \frac{\partial^k}{\partial z^k} G(\phi(z)) \Big|_{z=0} \right\}_{k=1}^m,$$

can be computed in time $t(n) + O(m^4)$.

Proof: For integer k and r, define the Bell polynomial to be

$$B_{k,r}(\phi^{(1)}(z), \phi^{(2)}(z), \ldots, \phi^{(k-r+1)}(z))$$

$$= \sum \frac{k!}{j_1! j_2! \ldots j_{k-r+1}!} \left(\frac{\phi^{(1)}(z)}{1!}\right)^{j_1} \left(\frac{\phi^{(2)}(z)}{2!}\right)^{j_2} \cdots \left(\frac{\phi^{(k)}(z)}{k!}\right)^{j_{k-r+1}} \tag{5.181}$$

where the summation is

$$j_1 + j_2 + \ldots + j_{k-r+1} = r \text{ and } j_1 + 2j_2 + \ldots + (k-r+1)j_{k-r+1} = k. \quad (5.182)$$

To compute the derivative of the composite function $G(\phi(z))$, we will use the Faa di Bruno's formula which states that

$$\frac{\partial^k}{\partial z^k} G(\phi(z))\Big|_{z=0} = \sum_{r=1}^{k} G^{(r)}(\phi(0)) \cdot B_{k,r}(\phi^{(1)}(0), \phi^{(2)}(0), \ldots, \phi^{(k-r+1)}(0)) \quad (5.183)$$

$$= \sum_{r=1}^{k} G^{(r)}(0) \cdot B_{k,r}(\phi^{(1)}(0), \phi^{(2)}(0), \ldots, \phi^{(k-r+1)}(0)), \quad (5.184)$$

where the notation $G^{(r)}(\phi(0))$ refers to $G^{(r)}(z)|_{z=\phi(0)}$ and the last equality comes from $\phi(0) = 0$. For each $B_{k,r}$, we define the corresponding partial ordinary Bell polynomials as

$$\hat{B}_{k,r}(y_1, \ldots, y_{k-r+1}) = \frac{r!}{k!} B_{k,r}(x_1, \ldots, x_{k-r+1}), \quad (5.185)$$

where $y_i = \frac{x_i}{i!}$. They satisfy the recurrence formula

$$\hat{B}_{k,r}(y_1, \ldots, y_{k-r+1}) = \sum_{i=1}^{k-r+1} y_i \hat{B}_{k-i,r-1}(y_1, \ldots, y_{k-r+1-i}). \quad (5.186)$$

Then after computing the first m derivatives $\{G^{(k)}(0)\}_{k=0}^{m}$ and $\{\phi^{(k)}(0)\}_{k=0}^{m}$ in time $t(n)$, Algorithm 11 computes $B_{k,r}$ in time $O(k^2 r)$. It suffices to compute $B_{m,r}$ for $r = 1, \ldots, m$ since all lower orders can be computed along the way, which takes total time $O(m^4)$. Therefore, the first m derivatives of $G(\phi(z))$ at $z = 0$ can be computed in time $t(n) + O(m^4)$. ■

Algorithm 11 Compute Bell polynomials [Tag23]

1: Set $y_i \leftarrow x_i/i!$, $i = 1, \ldots, k-r+1$
2: Set $\hat{B}_{0,0} = 1$, $\hat{B}_{i,0} = 0$, $i = 1, \ldots, k-r$ $l = 1, \ldots, r$ $i = l, \ldots, k-r+l$
3: $\hat{B}_{i,l}(y_1, \ldots, y_{k-r+1}) \leftarrow \sum_{j=1}^{i-l+1} y_j \hat{B}_{i-j,l-1}(y_1, \ldots, y_{k-r+1})$
4: Set $B_{k,r}(x_1, \ldots, x_{k-r+1}) \leftarrow \frac{k!}{r!} \hat{B}_{k,r}(y_1, \ldots, y_{k-r+1})$

5.9 Appendix:BPP-hardness of additive-error approximation (Theorem 135)

In this section, we prove the approximation problem in Theorem 135 is **BPP-hard**. This proof is similar to Section 4.2 in [AL10] and here we give a proof sketch.

First we embed classical randomized computations into quantum circuits. Given a parameter n, consider a quantum circuit of following form:

- Takes input as $|0\rangle^p |+\rangle^q$ for $p, q = poly(n)$.
- Applies a sequence of gates $Q = Q_L \dots Q_1$, where $L = poly(n)$ and $\{Q_i\}_i$ are reversible gates on constant qubits.
- Measure the first qubit in computational basis.

Denote p_0 as the probability of getting measurement outcome 0 in the first qubit. Suppose it is promised that either one of the following holds:

- Yes case: $p_0 \geq 2/3$,
- No case: $p_0 \leq 1/3$.

One can check that the problem of given such a circuit, output Yes/No correctly with probability greater than $2/3$ is **BPP-hard**.[9] In other word

Claim 144 *An algorithm for estimating p_0 with high probability is **BPP-hard**.*

Similarly as [AL10], to write p_0 as a tensor network, we first define a related circuit U on $p+q+1$ qubits: As shown in Figure 5.7, U firstly applies Q to $|0^{\otimes(p+1)}, +^{\otimes q}\rangle$, then copies the first qubit of Q to the additional qubit by CNOT, and then applies Q^{-1}.

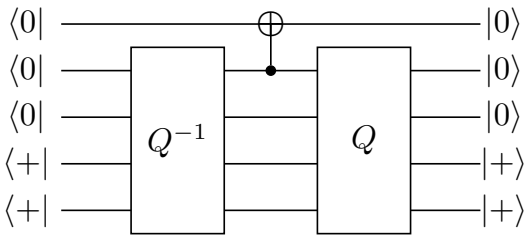


Figure 5.7: Illustration of Circuit for p_0.

One can check that

$$\langle 0^{\otimes(p+1)}, +^{\otimes q} \big| U \big| 0^{\otimes(p+1)}, +^{\otimes q} \rangle = p_0.$$

One can transform $\langle 0^{\otimes(p+1)}, +^{\otimes q} \big| U \big| 0^{\otimes(p+1)}, +^{\otimes q} \rangle$ to a tensor network T similarly as [AL10], then

$$\chi(T) = \langle 0^{\otimes(p+1)}, +^{\otimes q} \big| U \big| 0^{\otimes(p+1)}, +^{\otimes q} \rangle = p_0, \tag{5.187}$$

where

[9] Readers who are not familiar with randomized reduction may read Definition 7.19 in [AB09].

- Each reversible gate Q_i on constant qubits is translated to a tensor $M^{[Q_i]}$, which is of constant rank (constant degree) and bond dimension 2. Note that since Q_i is a reversible gate, which is a permutation, thus we have $\|Q_i\|_1 = 1$.
- We pair the input qubits on the left and right in Figure 5.7. $|0\rangle\langle 0|$ is translated into a tensor $M^{[0]} = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}$, $|+\rangle\langle +|$ is translated into a tensor $M^{[+]} = \frac{1}{2}\begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix}$. Note that $\|M^{[0]}\|_1 = \|M^{[+]}\|_1 = 1$.

Thus the approximation scale Δ_1 in Eq. (5.121) is equal to 1. Thus for $\epsilon = 1/poly(n)$, the approximation problem in Theorem 135, that is Eq. (5.120), requires approximating $\chi(T) = p_0$ within precision ϵ with high probability, thus is **BPP-hard** by Claim 144.

Chapter 6

COMMUTING LOCAL HAMILTONIAN PROBLEM ON 2D BEYOND QUBITS

6.1 Introduction

Understanding the properties of ground states of local Hamiltonians is a central problem in condensed matter physics. Kitaev famously formulated this problem as a decision problem, amenable to analysis from the perspective of computational complexity, by defining the local Hamiltonian problem (LHP), which asks whether the ground energy of a local Hamiltonian is below one threshold or greater than another. The LHP can be interpreted as the quantum generalization of the Boolean Satisfiability problem (SAT), where for SAT, all the terms $\{h_i\}_i$ are diagonal in the computational basis. Kitaev showed that the LHP is **QMA-complete** [KKR06], a quantum analog of the Cook–Levin theorem showing that SAT is **NP**-complete. Formally, a k-local Hamiltonian $H = \sum_i h_i$ is a Hermitian operator on n qudits, where each term h_i only acts on k qudits. Given two parameters a, b with $b - a \geq \frac{1}{poly(n)}$, the LHP is to determine whether the ground energy of H, namely the minimum eigenvalue, is smaller than a or greater than b.

It is widely believed that **QMA** $\neq$ **NP**, which would imply that the LHP is strictly harder than the SAT. A natural question then to ask is what properties make quantum SAT (LHP) harder than classical SAT? Alternatively, what additional constraints can make LHP easier than **QMA-complete**? An intermediate model that sits in between classical and quantum Hamiltonians is the Commuting Local Hamiltonian problem (CLHP) [BV03], in which the terms of the local Hamiltonian pairwise commute. Compared to the general LHP, the idea that CLHs should be more classical in nature stems from the intuition [FS97] in quantum physics, which suggests that it is the non-commutativity that makes the quantum world different from classical (The Heisenberg's uncertainty principle). Moreover, since all the terms of a CLH can be simultaneously diagonalized by a single basis, every eigenstate of the full Hamiltonian can be specified up to degeneracies by the corresponding eigenvalue for each term. The fact that the eigenstates have a classical specification suggests that the structure of the eigenbasis is more classical in nature. Based on this reasoning, it might be natural to conjecture that CLHP is equivalent to SAT. In fact, initial results

showing that special cases of CLHP are in **NP** did so by showing that the ground states of such CLHs are of limited entanglement, i.e. can be prepared by constant depth quantum circuits. If this held for all CLHs, then the general CLHP would be in **NP**, since one can take the constant depth circuit as a witness and check the ground energy classically by a light-cone argument. Unfortunately, it is not the case that ground states of all CLHs exhibit limited-entanglement. Indeed, the eigenstates of CLHs can potentially be highly entangled, as is true for the famous example of Kitaev's toric code [Kit03b]. Therefore, more sophisticated techniques need to be developed to prove certain subclasses of CLHP are in **NP**, as will be described in more detail later.

Commuting Hamiltonians provide a lens to study many fundamental aspects of quantum computing and many-body systems. For example, the stabilizer framework [Got97] is the basis for most error-correcting codes, and stabilizer codes can be seen as the ground states of commuting Hamiltonians. Commuting Hamiltonians are commonly used as a test ground for attacking difficult problems such as the quantum-PCP conjecture [AE11; Has13], NLTS conjecture [ABN23], Gibbs states preparation [KB16] and fast thermalization [Bar+23; Bar+23]. In particular, studying the ground state structure of commuting Hamiltonians could potentially provide insight into the area law and its connection to the efficient expressibility of ground states. The Area Law states that for ground states of gapped Hamiltonians on a finite-dimensional lattice, the entanglement entropy between two regions of ground states scales with the boundary between the regions as opposed to their volume. The area law is known to hold for 1D Hamiltonians and is known for 2D only under the assumption that the Hamiltonian is *frustration-free* and *uniformly gapped* [AAG22b]. It is widely hoped that proving the area law will lead to insight into whether such ground states can be efficiently expressed or constructed by quantum circuits. As a subclass of LHP, it is well known that the area law holds for CLHs, and yet, even in 2D, we do not know whether the ground states of CLHs can be efficiently represented or constructed.

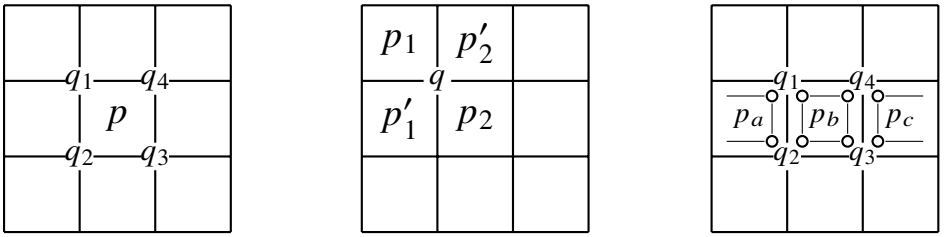
Previous Results

Despite two decades of study, the complexity of CLHP still remains open, with a few special cases known to be in **NP** ([BV03; AE13; AE11; Sch11; AKV18; Has12]). Bravyi and Vyalyi initiated this line of work, showing that qudit 2-local CLHP is in **NP** [BV03]. Their proof uses a decomposition lemma based on the theory of finite-dimensional C^*-algebra representations. All subsequent work on the CLHP

has made use of this framework.

Aharonov and Eldar [AE11] extended the results to the 3-local case for qubits and qutrits. Specifically, they proved that 3-local qubit-CLHP is in **NP**. They also proved that 3-local qutrit CLHP is in **NP** on the Nearly Euclidean interaction graphs. All the above results are proved by showing that there is a *trivial ground state*, i.e. the ground state can be prepared by a constant depth quantum circuit. This constant depth circuit is the **NP** witness and the energy of this trivial ground state can be checked in classical polynomial time by a light-cone argument. However, for 4-local CLHP, even if the interaction graph is a 2D square lattice, there are systems like the Toric code which have no trivial ground states.

In this work, we mainly focus on 4-local CLHP on a 2D square lattice with qudits (abbreviated as qudit-CLHP-2D). Specifically, consider a 2D square lattice as in Figure 6.1(a) with a qudit q on each vertex and on each plaquette p, there is a Hermitian term acting on the qudits on its four vertices. With some abuse of notations, we also use p to denote the Hermitian term on the plaquette p. The qudit-CLHP-2D is given an n-qudit Hamiltonian $H = \sum_p p$ where $\{p\}_p$ are commuting, two parameters a, b where $b - a \geq 1/poly(n)$, decide whether the minimum eigenvalue of H is smaller than a or greater than b. There is an alternate 2D setting in which qudits are placed on the edges and there are Hermitian terms on "plaquettes" and "stars". The two settings, i.e. qudits on vertices or qudits on edges, are equivalent when the underlying graph is a 2D square lattice. (See Appendix 6.7.)



(a) (b) (c)

Figure 6.1: Illustration of qudit-CLHP-2D. (a) Definition of the qudit-CLHP-2D. (b) The four terms which involve q are p_1, p_2, p_1', p_2'. (c) An example of 1D structure in Schuch's [Sch11] proof. In the figure the symbol $\circ$ represents the plaquette term acts non-trivially on the qudit. Specifically here among the 9 plaquettes in (c), only terms p_a, p_b, p_c acts non-trivially on at least one of q_1, q_2, q_3, q_4.

Along this line, Schuch [Sch11] first proved that the qubit-CLHP-2D is in **NP**. Schuch's proof provides a witness showing that a low-energy state exists without

giving an explicit description of the state. Indeed his proof leaves open the question of whether an explicit description of the state even exists. Aharonov, Kenneth, and Vigdorovich [AKV18] later gave a constructive proof, showing that after some transformation, the qubit-CLHP-2D is equivalent to the Toric code permitting boundaries. We say a proof is *constructive* if the prover shows that the ground energy is below a by providing a circuit for preparing the ground state. A *boundary* roughly means a qubit q where one of the four terms involving this qubit, i.e. p_1, p_2, p_1', p_2' in Figure 6.1(b), acts trivially on q. A Hamiltonian is equivalent to the Toric code permitting boundaries if after choosing an appropriate basis for each qubit, terms $\{p\}_p$ in the Hamiltonian act similarly as X or Z on the non-boundary qubits. Both proofs [Sch11; AKV18] for the qubit-CLHP-2D heavily depend on the restrictions to the qubits, since the induced algebra of a term p on a qubit q can only have a very limited structure. The limited structure for the qubits case does not hold for higher dimensional particles, not even for qutrits.

Hastings [Has12] proved a subclass of the qudit-CLHP-2D is in **NP** with some restrictive assumptions. Roughly speaking, he grouped all qudits on the same vertical line as a supersite, then viewed the 2D lattice as a 1D line, which reduced the qudit-CLHP-2D back to the "qudit" 2-local case [BV03]. However, the main problem is that those supersites are not really qudit of constant dimension. In fact, the dimension of the supersite is sub-exponential in the number of qudits. Thus Hastings further assumes that several technical conditions hold, like certain operators in the proof can be efficiently represented by matrix product operators of bounded dimension.

Besides the 2-local qudit-CLHP-2D, Bravyi and Vyalyi [BV03] also addressed the special case of the commuting Hamiltonian problem where all the terms are *factorized* (CHP-factorized). That is $H = \sum_i h_i$, and each term h_i is a tensor product of single-qudit Hermitian operators. For example, the Toric code is an instance of CHP-factorized, since each term is $X^{\otimes 4}$ or $Z^{\otimes 4}$. In general, in CHP-factorized each h_i is not necessarily local, and there are no constraints on the underlying interaction graph. Bravyi and Vyalyi give a non-constructive proof to show that the qubit-CHP-factorized is in **NP**. It is an open question whether their proof can be generalized to higher dimensional particles (qudit-CHP-factorized) or whether their proof for qubits can be made constructive.

All of the above results prove that subclasses of CLHP are in **NP**. On the other hand, Gosset, Mehta, and Vidick [GMV17] give a result that indicates CLHP might

be harder than **NP**, or even as hard as the general LHP. They show that the ground space connectivity problem of commuting local Hamiltonian is **QCMA**-complete, which is as hard as the ground space connectivity problem for the general local Hamiltonian.

Main results and proof overview

Main results

In this work, we give two new results on the qudit-CLHP-2D. The family of Hamiltonians considered in both results contains the Toric code as a special case. In this section we give an overview of those results and the correponding proof techniques. The formal proofs are in Section 6.4 and Section 6.5.

Theorem 145 extends the results for the qubit-CLHP-2D [Sch11; AKV18] to qutrits.

Theorem 145 *The qutrit-CLHP-2D is in **NP**.*

As far as we know, Theorem 145 is the first result for CLHP on 2D lattice beyond qubits. As noted, the results for the qubit-CLHP-2D [Sch11; AKV18] heavily depend on the limited dimension of qubits — the induced algebras on qubits have a very limited structure — which does not hold for qutrits. Our key idea to circumvent this problem introduces a technique to decrease the dimension of qudits. Specifically, denote the qudit-CLHP-2D instance as $H = \sum_p p$ and the Hilbert space of a qudit q as $\mathcal{H}^q$. Under certain conditions, we observe that it suffices to consider a new instance of qudit-CLHP by (1) Restricting $\mathcal{H}^q$ to a subspace of smaller dimension and (2) Constructing a new Hamiltonian by projecting all p to the smaller subspace, and then rounding all non-1 eigenvalues of the projected terms to 0. Moreover, the new instance preserves the correct answer in that "no" instances are converted to "no" instances, and "yes" instances are converted to "yes" instances. Thus, we show that our "decrease dimension and rounding" method (Lemma 167) can be interpreted as a non-constructive self-reduction for the qudit-CLHP. Here self-reduction means we reduce the original qudit-CLHP to a new qudit-CLHP where some qudits have strictly smaller dimensions. We emphasize that the key lemma, Lemma 167, works for the qudit-CLHP rather than only for qutrits, even without 2D geometry. This lemma might be of independent interest and bring new insights to tackle the general CLHP. We will explain this in more detail in the proof overview.

We next consider the special case of qudit-CLHP-2D where all the terms are factorized (qudit-CLHP-2D-factorized). Our second result, namely Theorem 146, is a

constructive proof showing that qudit-CLHP-2D-factorized is in **NP**. Although we do not give the details here, our proof could be considerably simplified to provide a non-constructive version based on the ideas of [BV03] and [Sch11]. Here we give a stronger constructive proof that characterizes the structure of the ground space.

Theorem 146 (Informal version of Theorem 189) *The Hamiltonian in the qudit-CLHP-2D-factorized is equivalent to a direct sum of qubit stabilizer Hamiltonian. In particular, a factorized 2D commuting local Hamiltonian always has a ground state which is equivalent to qubit stabilizer state. This implies qudit-CLHP-2D-factorized is in **NP**.*

We first briefly explain terminologies in Theorem 146. We say a commuting Hamiltonian $H = \sum_i h_i$ on space $\mathcal{H}_* := \otimes_q \mathcal{H}_*^q$ is equivalent to a qubit stabilizer Hamiltonian, if (1) For each $\mathcal{H}_*^q$, by choosing an appropriate basis, $\mathcal{H}_*^q$ is factorized as a tensor of Hilbert spaces of dimension 2. Thus each $\mathcal{H}_*^q$ can be interpreted as several qubits. We allow $dim(\mathcal{H}_*^q) = 1$ which corresponds to 0 qubit. (2) Each term h_i acts as a Pauli operator up to phases, with respect to the basis of those "qubits". We say a subspace $\mathcal{H}_* \subseteq \mathcal{H} := \otimes_q \mathcal{H}_q$ is *simple*, if $\mathcal{H}_*$ is a tensor product of subspaces of each qudit, i.e. $\mathcal{H}_* = \otimes_q \mathcal{H}_*^q$. We say a commuting Hamiltonian H on n-qudit space $\mathcal{H} = \otimes_q \mathcal{H}^q$ is equivalent to a direct sum of qubit stabilizer Hamiltonian, if the $\mathcal{H}$ is a direct sum of simple subspaces $\{\mathcal{H}_*\}_*$, such that $\forall i$, h_i keeps each $\mathcal{H}_*$ invariant, and H is equivalent to qubit stabilizer Hamiltonian on $\mathcal{H}_*$.

Although one might conjecture that factorized commuting Hamiltonians (CHP-factorized) are equivalent to a direct sum of stabilizer Hamiltonians even when not restricted to 2D, this is still an open question even for the qubit case [BV03].

To illustrate the difficulty, consider two factorized terms acting on two qudits q and q': $h := h_q \otimes h_{q'}$ and $\hat{h} := \hat{h}_q \otimes \hat{h}_{q'}$. If $h_q \hat{h}_q \neq 0$ and $h_{q'} \hat{h}_{q'} \neq 0$, then it must be the case that the factors in each individual qudit must commute or anti-commute which gives rise to a stabilizer-like structure. By contrast, if $h_q \hat{h}_q = 0$, then $[h, \hat{h}] = 0$ for any choice of $h_{q'}, \hat{h}_{q'}$. This means that $h_{q'}$ and $\hat{h}_{q'}$ can have an arbitrary relationship to each other and the two commuting terms $h, \hat{h}$ may look very different from stabilizers. This second possibility suggests, alternatively, that factorized CLHs in 2D could have a different topological order from stabilizer Hamiltonians. More precisely, one may conjecture that there is a factorized Hamiltonian, such that no ground state can be prepared by applying a constant depth quantum circuit to a stabilizer state. We show a negative answer to this conjecture, by proving

that the qudit-CLHP-2D-factorized is equivalent to a direct sum of qubit stabilizer Hamiltonian.

Overview for Theorem 145

We start by reducing the more general CLH problem to a slightly restricted case where the commuting terms are projections and the energy lower bound a is equal to 0. We will argue that if the more restricted version is in **NP** then the more general CLHP is also in **NP**. Consider an instance of the more general problem with Hermitian terms $\{h_i\}_i$ and bounds a and b, where $b-a \geq 1/\text{poly}(n)$. The **NP** prover can provide a vector describing the energy eigenvalue λ_i for each individual term h_i such that $\sum_i \lambda_i \leq a$. Then the verifier can replace each term h_i with $\hat{h}_i = I - \Pi_{i,\lambda_i}$, where Π_{i,λ_i} is the projection onto the eigenspace of h_i corresponding to eigenvalue λ_i. The new instance has a state $|\psi\rangle$ where $\hat{h}_i |\psi\rangle = 0$ for all i if and only if $h_i |\psi\rangle = \lambda_i |\psi\rangle$ for all i. Since the $\hat{h}_i$ are all commuting projectors, the eigenvalues of the Hamiltonian are non-negative integers. Thus, the verifier can set the new b to be equal to 1, resulting in a promise gap of 1. Therefore, when describing the proof of Theorem 145, we shall assume the terms are projections and that the question is whether there is a $|\psi\rangle$ where $\hat{h}_i |\psi\rangle = 0$ for all i (i.e. a *frustration-free* ground state). Note that this reduction does not work for the factorized case because even if the h_i is factorized, the resulting $\hat{h}_i$ is not necessarily factorized.

We start by introducing the framework of induced algebras. The basic ideas are sketched here and specified in more detail in Section 6.3. Denote $\mathcal{L}(\mathcal{H})$ as the set of all operators on a Hilbert space $\mathcal{H}$. A C^*-algebra is any algebra $\mathcal{A} \subseteq \mathcal{L}(\mathcal{H})$ which is also closed under the $\dagger$ operations and includes the identity. Consider a Hermitian term p acting $\mathcal{H} \otimes \mathcal{H}^c$. The operator p can be decomposed as

$$p = \sum_{i,j} h_{ij}^{\mathcal{H}} \otimes |i\rangle\langle j|^{\mathcal{H}^c}.$$

The induced algebra of p on space $\mathcal{H}$, denoted as $\mathcal{A}_p^{\mathcal{H}}$, is the C^*-algebra generated by $\{I_{\mathcal{H}}\} \cup \{h_{ij}^{\mathcal{H}}\}_{ij}$. Note that the particular decomposition of p is not critical other than the fact that the $|i\rangle\langle j|$ terms acting on $\mathcal{H}^c$ are linearly independent. The key technique introduced by [BV03] is the Structure Lemma which decouples two commuting terms in their overlapping space: consider two terms $p, \hat{p}$ sharing only one qudit q. If $[p, \hat{p}] = 0$, then the induced algebras, $\mathcal{A}_p^{\mathcal{H}^q}$ and $\mathcal{A}_{\hat{p}}^{\mathcal{H}^q}$, must commute, meaning that every operator in $\mathcal{A}_p^{\mathcal{H}^q}$ commutes with every operator in

$\mathcal{A}_{\hat{p}}^{\mathcal{H}^q}$. Furthermore, p, p' can be decoupled in $\mathcal{H}^q$, in that is there exists a direct sum decomposition $\mathcal{H}^q = \bigoplus_i \mathcal{H}_i^q$, where

- each $\mathcal{H}_i^q$ has a factorized structure: $\mathcal{H}_i^q = \mathcal{H}_i^{q1} \otimes \mathcal{H}_i^{q2}$
- and p only acts non-trivially on $\mathcal{H}_i^{q1}$ and $\hat{p}$ only acts non-trivially on $\mathcal{H}_i^{q2}$.

Both proofs showing that the qubit-CLHP-2D is in **NP** [Sch11; AKV18] depend heavily on the properties of qubits. In particular, if q is a qubit, then there are only two ways to have a direct sum decomposition of $\mathcal{H}^q$, namely as the direct sum of two 1-dimensional spaces or as a single 2-dimensional space. Note that we must also consider the case in which an induced algebra is *trivial*, meaning that $\mathcal{A}_p^{\mathcal{H}^q} = \{cI_{\mathcal{H}}\}_c$ which implies that the operator p acts trivially on qubit q. Using the Structure Lemma the following statement is true.

Fact 147 *Any two commuting non-trivial induced algebras on a qubit must be diagonalizable in the same basis.*

Note that Fact 147 is not true for qutrits. One may understand this statement intuitively by only basic linear algebra. In particular, if h_1, h_2, h are Hermitian operators on a qubit, where h is not proportional to the identity and both h_1 and h_2 commute with h, then all three operators can be diagonalized in the same basis. This observation is not true for qutrits, as there exist operators h, h_1, h_2 on a qutrit with h nontrivial, such that h commutes with h_1 and h_2, but the three operators cannot be diagonalized simultaneously.

Since the structure of our proof follows the same outline as Schuch's proof, we briefly explain how [Sch11] used Fact 147 to prove qubit-CLHP-2D is in **NP**. Consider an arbitrary qudit-CLHP-2D instance $H = \sum_p p$. As argued above, we can assume $\{p\}_p$ are commuting projections, and the goal is to determine whether $\lambda(H) = 0$ or $\lambda(H) \geq 1$. Note that in this setting, proving $\lambda(H) = 0$ is equivalent to proving $tr(\prod_p (I - p)) > 0$.

Now consider a qubit q in a 2D lattice (as shown in Figure 6.1(b)). We name the terms acting on q as p_1, p_2, p_1', p_2'. We define the set of *removable* qubits (defined implicitly in [Sch11]) as follows:

- $q \in R$: If the induced algebras of p_1, p_2 on q can be diagonalized in the same basis; or this condition holds for p_1', p_2'.

The set R is called removable since they can be effectively traced out. Specifically, suppose $q \in R$ and assume p_1, p_2 can be diagonalized in basis $\{|\phi_1\rangle, |\phi_2\rangle\}$. Then

$$tr\left[\prod_p (I-p)\right] =$$
$$\sum_{i=1,2} tr\left[|\phi_i\rangle\langle\phi_i|(1-p_1)|\phi_i\rangle\langle\phi_i|(1-p_2)|\phi_i\rangle\langle\phi_i| \prod_{p\neq p_1,p_2}(1-p)\right].$$

Each quantity in the sum on the right is the trace of a product of two positive semi-definite Hermitian operators, and therefore, each quantity in the sum is non-negative. Thus, $tr(\prod_p (I-p)) > 0$ if and only if one of the two quantities in the sum is positive. The prover will then provide a $|\phi_1\rangle$ or $|\phi_2\rangle$ for qubit q for which the trace is positive. Note that the proof is non-constructive because the ground state may not lie entirely within either the space spanned by the space spanned by $|\phi_1\rangle$ or the space spanned by $|\phi_2\rangle$. Suppose that the witness for qubit q is $|\phi_1\rangle$. The verifier must verify that

$$tr\left[|\phi_1\rangle\langle\phi_1|(1-p_1)|\phi_1\rangle\langle\phi_1|(1-p_2)|\phi_1\rangle\langle\phi_1| \prod_{p\neq p_1,p_2}(1-p)\right] > 0.$$

The other two terms that might act non-trivially on qubit q are p'_1 and p'_2. By Fact 147 we know either one of the induced algebras of p'_1, p'_2 on q acts trivially on q, or they can be diagonalized in the same basis. By considering each case separately, it can be argued that the qubit q can be traced out of all the terms. This process can be applied simultaneously for all the removable qubits. The remaining Hamiltonian will only contain terms that operate non-trivially on qubits that are not removable.

Using Fact 147, we know that if q is not removable ($q \notin R$), then one of p_1, p_2 and one of p'_1, p'_2 act trivially on q. Now consider a graph where each vertex represents a plaquette term p and two terms are connected if they operate non-trivially on a common qubit. Schuch argues that if none of the qubits are removable then this graph cannot have a vertex with a degree larger than two, namely the graph is a set of disjoint chains and cycles. The trace of each chain or cycle can be computed in classical polynomial time by representing each term as a tensor and contracting the tensors along the chain.

When moving to qutrits, we need to address the fact that the Structural Lemma allows for more complex decompositions of the Hilbert space of a qutrit, and in particular, Fact 147 no longer holds. Our key observation to tackle the problem,

is to introduce a new way to decrease the dimension of the particle — *Decrease Dimension and Rounding* (Lemma 167, Sec. 6.4), which can be applied to qudits that have a property which we call *semi-separable*.

We first describe the stronger condition of a *separable* qudit, which was introduced in [AE11]. Consider a CLHP $H = \sum_i h_i$. A qudit is separable if there exists a non-trivial decomposition $\mathcal{H}^q = \bigoplus_j \mathcal{H}_j^q$ such that all the terms h_i keep all subspaces $\mathcal{H}_j^q$ invariant. Note that if there is a separable qudit and a solution (i.e. a frustration-free ground state) exists, then a ground state must lie entirely within one of the $\mathcal{H}_j^q$. Thus a prover can provide the projector Π_j^q onto the subspace of qudit q which contains the solution. The verifier can replace each term h_i with $\Pi_j^q \cdot h_i \cdot \Pi_j^q$ and the dimension of the problem has been reduced.

We now extend this notion and define a qudit to be *semi-separable* if we allow at most one term to **not** keep the decomposition invariant. Our key observation is that, for CLHP, even for a semi-separable qudit q, an **NP** prover can similarly decrease the dimension of the qudit q, in a non-constructive way. Specifically, the **NP** prover will choose a subspace $\mathcal{H}_j^q$ and restrict all the terms in this subspace. Since there is one term (call it h_0) that is inconsistent with the decomposition, such restriction can not be done naturally. Instead, we project h_0 on the subspace $\mathcal{H}_j^q$ and then round all the not-1-eigenvalue to 0. By doing this we claim that we again get a new CLHP instance with a smaller dimension in qudit q. More importantly, we prove that the original CLHP has a frustration-free ground state iff there exists a j such that the new CLHP $H|_j$ also has a frustration-free ground state. The reduction is non-constructive because the ground state in the new instance may not be the same as the ground state in the original instance. The "semi-separable" technique is powerful especially for CLHP-2D where $H = \sum_p p$, since on 2D lattice as in Figure 6.1(b), for any qudit q, if we consider the decomposition of $\mathcal{H}^q$ induced by the induced algebra of p_1 on q, there are at most 2 terms, i.e. p'_1, p'_2, which do not keep the decomposition invariant. This observation is also true for CLHP embedded on a planar graph.

By the above argument, to prove the qutrit-CLHP-2D is in **NP**, w.l.o.g we can assume that there are no semi-separable qutrits. We further prove that the condition — the qutrit-CLHP-2D without semi-separable qutrits — leads to strong restrictions on the form of the Hamiltonian. In particular, we show that for the qutrit-CLHP-2D without semi-separable qutrits, if we consider again the graph of plaquette terms where two terms are connected by an edge if they act non-trivially on a common qutrit, then

this graph must also consist of disjoint chains or cycles. The trace of the 0-energy space can be computed as before in classical polynomial time by contracting 1D chains of tensors. The **NP** witness will be the indexes of subspaces chosen when removing all semi-separable qudits, and the subspaces for the removable qutrits.

Overview of Theorem 146

Recall that Theorem 146 is a constructive proof for the qubit-CLHP-2D-factorized, where factorized means each term is a tensor product of single-qudit Hermitian operators. As we mentioned in the main results section, finding a constructive proof that CHP-factorized is in **NP** is made difficult because if the product of two terms h and $\hat{h}$ is equal to 0, then their terms on individual qudits can have an arbitrary relationship. Namely, it is possible that $h^q \hat{h}^q \neq \pm \hat{h}^q h^q$. On the other hand, [BV03] showed that if all the terms are commuting obey the condition that $h^q \hat{h}^q = \pm \hat{h}^q h^q$ for each qudit, then the Hamiltonian will be related to a qubit stabilizer Hamiltonian.

The key part to proving the Theorem 146, is to remove the possibility that $h^q \hat{h}^q \neq \pm \hat{h}^q h^q$ for some q, without changing the ground space, which will imply a constructive proof by showing a correspondence with stabilizer Hamiltonians. In general, this removal is hard to achieve. Even for the qubit-CHP-factorized, it is still an open question whether there exists a constructive proof. However surprisingly, we can give a constructive proof for qudit-CHP-factorized, when the underlying interaction graph is 2D, i.e. qudit-CLHP-2D-factorized. Specifically, by using proof of contradiction, firstly we prove that qudit-CLHP-2D-factorized, if there are no separable qudits, then all terms must commute in a regular way. With a slight clarification, we show that the qudit-CLHP-2D-factorized without separable qudits is equivalent to qubit stabilizer Hamiltonian. For the more general case where there are separable qudits, we then notice that there exists a partition of the n-qudit space into simple subspaces, such that the restricted Hamiltonian on each subspace has no separable qudits. This partition is achieved by the following: when there is a separable qudit q w.r.t decomposition $\mathcal{H}^q = \oplus_j \mathcal{H}^q_j$, we partition the whole space according to this decomposition. We recursively perform this partition until for each subspace, the restricted Hamiltonian has no separable qudits.

Structure of the manuscript

The manuscript is structured as follows. In Sec. 6.2 we give notations and definitions which are used throughout this manuscript. In Sec. 6.3 we review the necessary

definitions and techniques of C^*-algebra and the Structure Lemma required for proving that the qutrit-CLHP-2D is in **NP**. In Sec. 6.4 and Sec. 6.5, we give proofs for the qutrit-CLHP-2D and the qudit-CLHP-2D-factorized respectively.

The manuscript is written in a way that several proofs can be read separately. In summary, for readers interested in Lemma 167 (The Decrease Dimension and Rounding Lemma), the suggested order is Sec. 6.2 and Sec. 6.4. For readers interested in the qutrit-CLHP-2D, the suggested order is Sec 6.2, Sec. 6.3, Sec. 6.4. For readers interested in the qudit-CLHP-2D-factorized, the suggested order is Sec 6.2, Sec. 6.5, and then Sec. 6.3 if necessary.

Conclusion and future work

In this manuscript, we give two new results of the qudit-CLHP-2D. First, we proved that qutrit-CLHP-2D is in **NP**, by introducing a non-constructive way of self-reduction for the qudit-CLHP, when there are semi-separable qudits. This self-reduction (proven in Lemma 167) works for qudit and might be of independent interest. Second, we prove that qudit-CLHP-2D-factorized is in **NP**, by showing that the Hamiltonian is equivalent to a direct sum of qubit stabilizer Hamiltonian.

One direct question is whether our proof for qutrit-CLHP-2D can be made to be constructive, that is, whether one can prepare the ground state by polynomial-size quantum circuits. Aharonov, Kenneth and Vigdorovich [AKV18] proved that the qubit-CLHP-2D is equivalent to the Toric code permitting boundary. It is natural to ask whether qutrit-CLHP-2D, or general qudit-CLHP-2D, can have different ground space properties from the stabilizer Hamiltonians. Another question is whether our constructive proof for the qudit-CLHP-2D-factorized can be modified to prepare the ground states of the qubit-CHP (without 2D geometry). Recall that the qubit-CHP is in **NP** by a non-constructive method [BV03].

A further question is to extend the frontier of the complexity of CLHP. In particular, we conjecture that Lemma 167 can be used in more general settings. As we have mentioned, Lemma 167 works for qudits of any finite dimension, which implies w.l.o.g we can assume that there are no semi-separable qudits in the Hamiltonian. To prove qudit-CLHP-2D$\in$ **NP**, one must further prove that qudit-CLHP-2D without semi-separable qudits is restricted in such a way that there exists an **NP** proof. We are only able to prove this restriction for the qutrit case. The intuitive reason is that 3 is a prime while 4 is not. Despite this, Lemma 167, viewed as a techinque for simplifying qudit-CLHPs by removing semi-separable qudits, might become

more powerful when combined with other simplification techniques like removing terms [AKV18]. It is interesting to see whether one can prove that the qudit-CLHP-2D$\in$ **NP** by combining Lemma 167 and other techinques. Another promising setting of further utilizing Lemma 167 is considering 3-local qutrit-CLHP, without any geometry constraints. Recall that [AE11] proved for 3-local qubit-CLHP is in **NP**, by showing that after removing all separable qubits, the resulting Hamiltonian can be viewed as a 2-local qudit-CLHP. It is possible that for 3-local qutrit-CLHP, if we remove all semi-separable qutrits, the Hamiltonian is again of a 2-local structure, which will imply 3-local qutrit-CLHP is in **NP**.

Most known results are trying to show that CLHP is in **NP**. On the other side, it will be very interesting to provide any evidence that CLHP might be harder than **NP**.

6.2 Preliminaries

Notation

Given a Hermitian operator H, we use $\lambda(H)$ to denote its ground energy, i.e. minimum eigenvalue. For two operators, h and h', we use $[h, h']$ to denote its commutator $hh' - h'h$. In particular, $[h, h'] = 0$ means h, h' are commuting. Two sets of operators, S and $\hat{S}$, commute if $[h, \hat{h}] = 0, \forall h \in S, \hat{h} \in \hat{S}$. For a set of Hermitian operators $\{h_i\}_i$, we use $\ker\{h_i\}_i$ to denote its common 0-eigenspace, i.e. $\ker\{h_i\}_i := \{|\psi\rangle \mid h_i |\psi\rangle = 0, \forall i\}$. We say $\ker\{h_i\}_i$ is non-trivial iff $\ker\{h_i\}_i \neq \{0\}$.

Here 0 refers the zero vector. With some abuse of notations, we use 0 both for real number zero, and zero vector.

For ease of illustration, we also denote a Hermitian $H = \sum_i h_i$ as a set $\{h_i\}_i$. We say a Hermitian operator Π is a projection if $\Pi^2 = \Pi$. When $\{h_i\}_i$ are commuting projections, we have $\lambda(H) = 0$ iff $\ker \{h_i\}_i$ is non-trivial.

Let $\mathcal{H}$ be a finite-dimensional Hilbert space. We use $\mathcal{L}(\mathcal{H})$ to denote the set of linear operators on $\mathcal{H}$. For Hermitian h, we use $h \succeq 0$ to denote h is positive semidefinite, that is all of its eigenvalues are non-negative. We use I to denote the identity matrix. Let h be a Hermitian operator on Hilbert space $\mathcal{X} = \mathcal{H} \otimes \mathcal{Z}$, we say h keeps the decomposition $\mathcal{H} = \bigoplus_i \mathcal{H}_i$ invariant if h keeps the subspace $\mathcal{H}_i \otimes \mathcal{Z}$ invariant, $\forall i$. We say the decomposition $\mathcal{H} = \bigoplus_{i=1}^m \mathcal{H}_i$ is non-trivial if $m \geq 2$.

In the following, we use q to denote a qudit, and $\mathcal{H}^q$ to denote the Hilbert space of the qudit q. Consider an operator h acting on n qudits. We say that h acts trivially on a qudit q if h acts as identity on $\mathcal{H}^q$. When h acts non-trivially on only k of

the n qudits, we will interchangeably view h as an operator on k qudits or a global operator on n qudits. The meaning will be clear in the context. We use $tr_S()$ for tracing out the qudits in S, and use $tr()$ for tracing out all the qudits. We use S^c to denote the set of qudits outside S.

Formal Problem Definitions

Commuting k-Local Hamiltonian We say a Hermitian operator H on n qudits is a commuting k-local Hamiltonian, if $H = \sum_{i=1}^{m} h_i$ for $m = poly(n)$, where each h_i only acts non-trivially on k qudits, and $h_i h_j = h_j h_i, \forall i, j$. We allow different qudits to have different dimensions. In particular, for qutrit k-local commuting local Hamiltonian, we allow the dimension of each qudit to be either 1, 2 or 3.

2D and Factorized Variants Consider a 2D square lattice as in Figure 6.1(a), on each vertex there is a qudit q, and on each plaquette p there is a Hermitian term acting on the qudits on its four vertices. With some abuse of notations, we also use p to denote the Hermitian term on the plaquette p. We say a commuting (4-local) Hamiltonian is on 2D if there is an underlying 2D square lattice and plaquette terms defined as above such that $H = \sum_p p$ and all $\{p\}_p$ are pairwise commuting.

We further say a commuting (4-local) Hamiltonian on 2D is factorized, if each p is factorized on its vertices, that is $p = p^{q_1} \otimes p^{q_2} \otimes p^{q_3} \otimes p^{q_4}$ for Hermitian terms p^{q_i} acting on qudit q_i, as shown in Figure 6.1(a). We call p^{q_i} factors. For the Toric code, $p \in \{X^{\otimes 4}, Z^{\otimes 4}\}$.

Commuting k-Local Hamiltonian problem Given a family of commuting k-local Hamiltonian $H = \sum_i h_i$ on n qudits and parameters $a, b \in \mathbb{R}$ with $b - a \geq 1/poly(n)$. The commuting k-local Hamiltonian problem w.r.t (H, a, b) is a promise problem that decides whether $\lambda(H) \leq a$ or $\lambda(H) \geq b$. We denote this problem as k-qudit-CLHP(H, a, b), abbreviated as k-qudit-CLHP when H, a, b are clear in the context. For 2D and 2D-factorized variants of commuting k-local Hamiltonian problem, $k = 4$ is clear and we abbreviate them as qudit-CLHP-2D and qudit-CLHP-2D-factorized respectively.

We define a special case of the k-qudit-CLHP called k-qudit-CLHP-projection where each term h_i is a projection, $b = 1$, and $a = 0$. Note that since $\{h_i\}_i$ are commuting projections, we know $\lambda(H)$ must be a non-negative integer. Thus in the No instance we use $\lambda(H) \geq 1$ rather than $\lambda(H) \geq 1/poly(n)$. We define qudit-CLHP-2D-projection similarly.

More Definitions

Consider a commuting k-local Hamiltonian $H = \sum_i h_i$ on n qudits with Hermitian terms $\{h_i\}_i$. Although h_i acts non-trivially only on k qudits, in this section we view it as an operator on n qudits. We will name the n qudits as $q_1, q_2, ..., q_n$. When we refer to an arbitrary qudit, we name it as q.

Definition 148 (Separable qudit) *A qudit q is separable w.r.t Hermitian terms $\{h_i\}_i$ if there exists a non-trivial decomposition of its Hilbert space $\mathcal{H}^q = \bigoplus_{j=1}^m \mathcal{H}_j^q$ s.t all h_i keep the decomposition invariant. Here non-trivial means $m \geq 2$. We use Π_j to denote the projection onto $\mathcal{H}_j^q$.*

The definition of separable is first introduced by [AE11]. Roughly speaking, it says all the Hermitian terms $\{h_i\}_i$ are block-diagonalized in the same way. We introduce the notion of semi-separable qudit, which will play a key role in the proof of the qutrit-CLHP-2D.

Definition 149 (Semi-separable qudit) *A qudit q is semi-separable w.r.t Hermitian terms $\{h_i\}_i$ if there exists a non-trivial decomposition of its Hilbert space $\mathcal{H}^q = \bigoplus_{j=1}^m \mathcal{H}_j^q$ s.t all but one h_i keeps the decomposition invariant. Here non-trivial means $m \geq 2$. We use Π_j as the projection onto $\mathcal{H}_j^q$. By convention when referring to a specific qudit, we will denote the term which does not keep the decomposition invariant as h_0.*

Semi-separable qudit is a relaxation of separable qudits, in the sense that we allow one term to be not block-diagonalized w.r.t the decomposition $\mathcal{H}^q = \bigoplus_{j=1}^m \mathcal{H}_j^q$. Note that by the definition of semi-separable, h_i is Hermitian and we have $[h_i, \Pi_j] = 0, \forall i \neq 0, \forall j$. We will repeatedly use this fact. It is also important to keep in mind that $[h_0, \Pi_j]$ might not be equal to 0, since we allow h_0 not keeping $\mathcal{H}_j^q$ invariant.

6.3 Review of C^*-algebras and the Structure Lemma

This section is a review of C^*-algebra and the Structure Lemma [BV03], which is the key tool to analyze the structures in the commuting local Hamiltonians. A more detailed proof on those techniques can be seen in Sec. 7.3 of [Gha+15]. The following notations and lemmas are rephrased from [AKV18] and Sec. 7.3 of [Gha+15].

Basics of C^*-algebras

Definition 150 (C^*-algebra) *Let $\mathcal{H}$ be a finite dimensional Hilbert space, a C^*-algebra is any algebra $\mathcal{A} \subseteq \mathcal{L}(\mathcal{H})$ which is closed under the $\dagger$ operations and includes the identity. We say that two C^*-algebras, $\mathcal{A}$ and $\mathcal{A}'$, commute if $[a, a'] = 0, \forall a \in \mathcal{A}, a' \in \mathcal{A}'$.*

Definition 151 (Trivial operator and algebra) *Let $\mathcal{H}$ be a finite-dimensional Hilbert space. We say an operator $h \in \mathcal{L}(\mathcal{H})$ acting trivially on $\mathcal{H}$ if $h = cI_{\mathcal{H}}$ for some constant c. We say a C^*-algebra on $\mathcal{A} \subseteq \mathcal{L}(\mathcal{H})$ is trivial if every operators in $\mathcal{A}$ is trivial, i.e. $\mathcal{A} = \{cI_{\mathcal{H}}\}_c$. If $\mathcal{H} = \mathcal{H}_1 \otimes \mathcal{H}_2$, we say h acts trivially on $\mathcal{H}_1$ if $h = cI_{\mathcal{H}_1} \otimes h_2$ for $h_2 \in \mathcal{L}(\mathcal{H}_2)$.*

Definition 152 (Center of C^*-algebra) *The center of a C^*-algebra $\mathcal{A}$ is defined as the set of operators in $\mathcal{A}$ which commutes with $\mathcal{A}$, that is*

$$\mathcal{Z}(\mathcal{A}) := \{a \in \mathcal{A} | [a, a'] = 0, \forall a' \in \mathcal{A}\}. \tag{6.1}$$

Then we introduce the induced algebra, which connects a Hermitian operator and a C^*-algebra.

Definition 153 (Induced algebra) *Let h be a Hermitian operator acting on Hilbert space $\mathcal{H} \otimes \mathcal{H}'$. Consider the decomposition*

$$h = \sum_{i,j} h_{ij}^{\mathcal{H}} \otimes |i\rangle\langle j|^{\mathcal{H}'}, \tag{6.2}$$

where $\{|i\rangle\}_i$ is an orthogonal basis of $\mathcal{H}'$. The induced algebra of h on $\mathcal{H}$, denoted as $\mathcal{A}_h^{\mathcal{H}}$, is defined as the C^*-algebra generated by $\{h_{ij}^{\mathcal{H}}\}_{ij} \cup \{I_{\mathcal{H}'}\}$. We abbreviate $\mathcal{A}_h^{\mathcal{H}}$ as $\mathcal{A}_h$ when $\mathcal{H}$ is clear in the context. We abbreviate $\mathcal{A}_h^{\mathcal{H}_q}$ as $\mathcal{A}_h^q$ for qudit q.

The induced algebra is independent of the chosen decomposition for Hermitian h.

Lemma 154 (Claim B.3 of [AKV18]) *In Definition 153 consider two decompositions*

$$h = \sum_{ij} h_{ij}^{\mathcal{H}} \otimes g_{ij}^{\mathcal{H}'} = \sum_{ij} \hat{h}_{ij}^{\mathcal{H}} \otimes \hat{g}_{ij}^{\mathcal{H}'}, \tag{6.3}$$

where the sets $\{g_{ij}^{\mathcal{H}'}\}_{ij}, \{\hat{g}_{ij}^{\mathcal{H}'}\}_{ij}$ are linearly independent respectively. Then the C^*-algebra generated by $\{h_{ij}^{\mathcal{H}}\}_{ij}$ is the same as the one generated by $\{\hat{h}_{ij}^{\mathcal{H}'}\}_{ij}$.

By Lemma 154 we know the induced algebra of h on $\mathcal{H}$, i.e. $\mathcal{A}_h^{\mathcal{H}}$ in Definition 153, is independent of the decomposition we choose, thus $\mathcal{A}_h^{\mathcal{H}}$ is well-defined. Note that if there is a decomposition $\mathcal{H} = \bigoplus_i \mathcal{H}_i$ such that $\mathcal{A}_h^{\mathcal{H}}$ keeps $\mathcal{H}_i$ invariant, $\forall i$, then it follows that h keeps $\mathcal{H}_i$ invariant, $\forall i$.

The Structure Lemma

The Structure Lemma [Tak+03] says that every finite-dimensional C^*-algebra is a direct sum of algebras of all operators on a Hilbert space. See Sec. 7.3 of [Gha+15] for an accessible proof of the Structure Lemma. The following statement is taken from [AKV18], which is a classification of finite dimensional C^*-algebras.

Lemma 155 (The Structure Lemma) *Let $\mathcal{A} \subseteq L(\mathcal{H})$ be a C^*-algebra where $\mathcal{H}$ is finite dimensional. There exists a direct sum decomposition: $\mathcal{H} = \bigoplus_i \mathcal{H}_i$ and a tensor product structure $\mathcal{H}_i = \mathcal{H}_i^1 \otimes \mathcal{H}_i^2$ such that*

$$\mathcal{A} = \bigoplus_i \mathcal{L}(\mathcal{H}_i^1) \otimes \mathcal{I}(\mathcal{H}_i^2).$$

Furthermore, the center of $\mathcal{A}$ is spanned by $\{\Pi_i\}_i$, where Π_i is the projection onto the subspace $\mathcal{H}_i$.

Given a C^*-algebra $\mathcal{A}$, we denote the decomposition $\mathcal{H} = \bigoplus_i \mathcal{H}_i$ in Lemma 155 as *the decomposition induced by* $\mathcal{A}$. Note that here we do not argue whether the decomposition in Lemma 155 is unique or not. However, for clarity when we mention the decomposition induced by $\mathcal{A}$, we always refer to the same canonical decomposition. For example, we can set the canonical decomposition to be the one obtained by the proof in Sec. 7.3 of [Gha+15]. In the following we give some definitions of decompositions, and a further remark on Lemma 155.

Definition 156 (Trivial, Better decomposition) *Consider the decomposition of a finite-dimensional space $\mathcal{H} = \bigoplus_{i=1}^m \mathcal{H}_i$. We say the decomposition is trivial if $m = 1$. We say one decomposition is better [1] than another if it has a bigger m.*

Lemma 155 implies all operators in $\mathcal{A}$ keep the decomposition $\mathcal{H} = \bigoplus_i \mathcal{H}_i$ invariant. It is worth noting that the decomposition induced by $\mathcal{A}$ might **not** be the

[1] Here we measure "better" only in terms of m. We do not require any relationship between the subspaces of the better decomposition $\mathcal{H} = \bigoplus_{i=1}^m \mathcal{H}_i$ and the worse $\mathcal{H} = \bigoplus_{i=1}^{m'} \mathcal{H}'_i$ for $m > m'$. Note that even for two commuting algebras $\mathcal{A}, \hat{\mathcal{A}} \subseteq L(\mathcal{H})$, the two decompositions of $\mathcal{H}$ induced by $\mathcal{A}, \hat{\mathcal{A}}$ might **not** be finer than each other. That's why we use "better" rather than "finer" here. We define in this way just to ease notations and make our proof more precise.

best decomposition that $\mathcal{A}$ keeps invariant. In particular, consider the C^*-algebra $\mathcal{A}$ generated by I, i.e. $\{cI\}_{c\in\mathbb{C}}$. The decomposition of $\mathcal{H}$ induced by $\mathcal{A}$ is trivial, i.e. $\mathcal{H} = \mathcal{H}_1$, but $\mathcal{A}$ keeps any decomposition of $\mathcal{H}$ invariant. Using Lemma 155, we can analyze how two induced algebras can commute with each other.

Corollary 157 (The Structure Lemma) *Let $\mathcal{A}_h$ be a C^*-algebra acting on a finite dimensional $\mathcal{H}$. Let $\mathcal{H} = \bigoplus_i \mathcal{H}_i$, $\mathcal{H}_i = \mathcal{H}_i^1 \otimes \mathcal{H}_i^2$, is the decomposition induced by $\mathcal{A}_h$ by Lemma 155. Consider another C^*-algebra $\mathcal{A}_{h'}$ on $\mathcal{H}$ which commutes with $\mathcal{A}_h$, we have*

$$\mathcal{A}_h = \bigoplus_i \mathcal{L}(\mathcal{H}_i^1) \otimes \mathcal{I}(\mathcal{H}_i^2)$$
$$\mathcal{A}_{h'} \subseteq \bigoplus_i \mathcal{I}(\mathcal{H}_i^1) \otimes \mathcal{L}(\mathcal{H}_i^2).$$

In particular, all operators in $\mathcal{A}_h, \mathcal{A}_{h'}$ keep the decomposition $\mathcal{H} = \bigoplus_i \mathcal{H}_i$ invariant.

Proof: Firstly by Lemma 155 we can get the decomposition of $\mathcal{H}$ induced by $\mathcal{A}_h$. Further let Π_i be the projection onto $\mathcal{H}_i$, by Lemma 155 we know $\Pi_i \in \mathcal{Z}(\mathcal{A}_h) \subseteq \mathcal{A}_h$. Since $\mathcal{A}_{h'}$ commutes with $\mathcal{A}_h$, thus $\mathcal{A}_{h'}$ commutes with Π_i, thus $\mathcal{A}_{h'}$ keeps $\mathcal{H}_i$ invariant. Since only cI can commute with all operators in a Hilbert space, i.e. $\mathcal{L}(H_i^1)$, thus we finish the proof. ∎

In the following, we give a sufficient condition that implies the decomposition of space induced by the C^*-algebra is non-trivial.

Lemma 158 (Non-trivial decomposition) *Let $\mathcal{A}$ be a C^*-algebra on a finite dimensional $\mathcal{H}$. Denote the decomposition induced by $\mathcal{A}$ in Lemma 155 be $\mathcal{H} = \bigoplus_i \mathcal{H}_i$. Consider another C^*-algebra $\mathcal{A}'$ on $\mathcal{H}$ which commutes with $\mathcal{A}$. If $\exists h \neq 0 \in \mathcal{A}, h' \neq 0 \in \mathcal{A}'$ such that $hh' = 0$. Then the decomposition $\mathcal{H} = \bigoplus_i H_i$ is non-trivial.*

Proof: With contradiction suppose the decomposition is trivial, i.e.

$$\mathcal{H} = \mathcal{H}_1 = \mathcal{H}_1^1 \otimes \mathcal{H}_1^2.$$

By Corollary 157 we have

$$h \in \mathcal{A} = \mathcal{L}(\mathcal{H}_1^1) \otimes \mathcal{I}_{\mathcal{H}_1^2},$$
$$h' \in \mathcal{A}' \subseteq \mathcal{I}_{\mathcal{H}_1^1} \otimes \mathcal{L}(\mathcal{H}_1^2).$$

Since $h \neq 0, h' \neq 0$, we have $hh' \neq 0$ which leads to a contradiction. ∎

Partitions Inducted by Commuting Operators

The following definitions will be used throughout Sec. 6.5.

Definition 159 *Let h, h' be two Hermitian terms acting on $\mathcal{X} \otimes \mathcal{H} \otimes \mathcal{Z}$ where $dim(\mathcal{H}) = d$. Suppose that h acts trivially on $\mathcal{Z}$, h' acts trivially on $\mathcal{X}$, $[h, h'] = 0$, and at least one of h, h' acts non-trivially on $\mathcal{H}$. Let the decomposition $\mathcal{H} = \bigoplus_i \mathcal{H}_i$ be the better one induced by $\mathcal{A}_h^{\mathcal{H}}$ or $\mathcal{A}_{h'}^{\mathcal{H}}$. We say that h, h' commute in $(d_1, ..., d_m)$-way on $\mathcal{H}$ if $dim(\mathcal{H}_i) = d_i$.*

Note that by Corollary 157, h, h' have a tensor-product structure on $\mathcal{H}_i$. Since the dimension of any Hilbert space must be an integer, two terms on a qutrit q of dimension 3 can only commute in the following ways.

Lemma 160 *Let h, h' be two Hermitian terms acting on $\mathcal{X} \otimes \mathcal{H} \otimes \mathcal{Z}$ where $dim(\mathcal{H}) = 3$. If h acts trivially on $\mathcal{Z}$, h' acts trivially on $\mathcal{X}$, $[h, h'] = 0$, and at least one of h, h' acts non-trivially on $\mathcal{H}$. Let the decomposition $\mathcal{H} = \bigoplus_i \mathcal{H}_i$ be the better one induced by $\mathcal{A}_h^{\mathcal{H}}$ or $\mathcal{A}_{h'}^{\mathcal{H}}$. then h, h' must commute on $\mathcal{H}$ via one of the following ways*

- $(1, 1, 1)$*-way:*
$$\mathcal{H} = \mathcal{H}_1 \bigoplus \mathcal{H}_2 \bigoplus \mathcal{H}_3,$$
where $dim(\mathcal{H}_i) = 1, \forall i$.

- $(1, 2)$*-way:*
$$\mathcal{H} = \mathcal{H}_1 \bigoplus \mathcal{H}_2,$$
where $dim(\mathcal{H}_1) = 1, dim(\mathcal{H}_2) = 2$.

- (3)*-way:*
$$\mathcal{H} = \mathcal{H}_1,$$
where $dim(\mathcal{H}_1) = 3$. One of h, h' acts trivially on $\mathcal{H}$, and for another the induced algebra on $\mathcal{H}$ is the full algebra $\mathcal{L}(\mathcal{H})$.

Proof: By corollary 157, we get a decomposition

$$\mathcal{H} = \bigoplus_i \mathcal{H}_i,$$

where $\mathcal{H}_i = \mathcal{H}_i^1 \otimes \mathcal{H}_i^2$. Since the dimension of a subspace must be an integer we get the above 3 possible ways. Further for the (3)-way, by assumption the decomposition

induced by both $\mathcal{A}_h^{\mathcal{H}}, \mathcal{A}_{h'}^{\mathcal{H}}$ are $\mathcal{H} = \mathcal{H}_1$. Since $dim(\mathcal{H}_1) = dim(\mathcal{H}) = 3$ is a prime, which means it can only be a tensor product of a one-dimensional Hilbert space and a three-dimensional Hilbert space. Thus for both $\mathcal{A}_h^{\mathcal{H}}, \mathcal{A}_{h'}^{\mathcal{H}}$, they equal to either $\{cI\}_c$ or $\mathcal{L}(\mathcal{H})$. Since at least one of h, h' acts non-trivially on $\mathcal{H}$, we know one of the induced algebra are $\mathcal{L}(\mathcal{H})$, w.l.o.g suppose $\mathcal{A}_h^{\mathcal{H}} = \mathcal{L}(\mathcal{H})$. Again by corollary 157, $\mathcal{A}_{h'}^{\mathcal{H}}$ should be $\{cI\}_c$, thus h' acts trivially on $\mathcal{H}$. ∎

Similar arguments for qubits are widely used in the proof of the qubit-CLHP-2D is in **NP** [Sch11; AKV18]. We summarize it as below.

Lemma 161 *If we change $dim(\mathcal{H})$ to be 2 in the statement of Lemma 160, then h, h' must commute on $\mathcal{H}$ via one of the following ways*

- $(1,1)$*-way if* $\mathcal{H} = \mathcal{H}_1 \bigoplus \mathcal{H}_2$ *where* $dim(\mathcal{H}_1) = dim(\mathcal{H}_2) = 1$.
- (2)*-way if* $\mathcal{H} = \mathcal{H}_1$ *where* $dim(\mathcal{H}_1) = 2$. *One of h, h' acts trivially on $\mathcal{H}$, and for another the induced algebra on $\mathcal{H}$ is the full algebra $\mathcal{L}(\mathcal{H})$.*

Note that Lemma 160 and Lemma 161 only involve 2 commuting terms h, h', and their overlapping space is only $\mathcal{H}$. Those techniques do not directly apply to 2D Hamiltonians, where some of the terms overlap on 2 qudits.

6.4 Qutrit Commuting Local Hamiltonian on 2D

In this section, we will prove that the qutrit-CLHP-2D is in **NP**. This proof is non-constructive. Note that if the qutrit-CLHP-2D-projection is in **NP**, then the qutrit-CLHP-2D is in **NP**. The proof of this statement is in Appendix. 6.6. Thus in this section, w.l.o.g we assume that all the terms p are projections and prove that the qutrit-CLHP-2D-projection is in **NP**.

The proof sketch is as follows. In Sec. 6.4 we prove that we can further assume that there are no semi-separable qudits. In Sec. 6.4 we prove that for the qutrit-CLHP-2D without semi-separable qudits, there are strong restrictions on the form of Hamiltonian. Finally, in Sec. 6.4 we prove that with such restrictions, we can use Schuch's method [Sch11] again.

Self-reduction for CLHP with semi-separable qudits

Lemmas in this section work for k-qudit-CLHP-projection [2] , that is we do not assume that each particle is a qutrit, or the Hamiltonian is on 2D. Recall that k-

[2] W.l.o.g we can assume that all terms are projections by Lemma 191.

qudit-CLHP-projection is as follows: consider a k-qudit-CLHP $H = \sum_i h_i$ where $\{h_i\}_i$ are k-local projections for some constant k, $[h_i, h_j] = 0$ for $i \neq j$. The question is to determine whether $\lambda(H) = 0$ or $\lambda(H) \geq 1$. Note that $\lambda(H) = 0$ iff all the commuting projections $\{h_i\}_i$ have a common 0-eigenvector, i.e. $ker\{h_i\}_i$ is non-trivial. When $\lambda(H) = 0$, the common 0-eigenvectors of $\{h_i\}_i$ are the ground states of H. We also denote the Hamiltonian H as $\{h_i\}_i$.

The key lemma in this section, Lemma 167, is to prove that when there is a semi-separable qudit, the prover can perform a non-constructive self-reduction for the k-qudit-CLHP-projection. Here self-reduction means reducing the k-qudit-CLHP-projection to another k-qudit-CLHP-projection, where the Hilbert space of the qudit has a smaller dimension. Before going into the formal proofs, in the following, we intuitively explain how Lemma 167 works. Specifically, temporarily assume that $\lambda(H) = 0$, and thus we are in the Yes instance and try to prove $\lambda(H) = 0$. We begin with the example when there is a separable qudit, then generalize this idea to the case of semi-separable qudit, and after that we give the formal proofs.

When there is a separable qudit q, the prover can easily perform a constructive self-reduction. Suppose q is a separable qudit, by definition, there exists a non-trivial decomposition

$$\mathcal{H}^q = \bigoplus_j \mathcal{H}^q_j$$

such that all the terms $\{h_i\}_i$ keep the decomposition invariant. Then there must be a subspace $\mathcal{H}_{j_0}$ which contains a common 0-eigenstate of $\{h_i\}_i$. Denote the projector onto $\mathcal{H}^q_{j_0}$ as Π_{j_0}. The prover can give the decomposition

$$\mathcal{H}^q = \bigoplus_j \mathcal{H}^q_j$$

and the index j_0. The verifier checks that q is a separable qudit, then restricts the space of q from $\mathcal{H}^q$ to $\mathcal{H}^q_{j_0}$, and restricts all terms $\{h_i\}_i$ to

$$\{h_i^{<j_0>} := \Pi_{j_0} h_i \Pi_{j_0}\}_i,$$

and ask the prover to prove that $\{h_i^{<j_0>}\}_i$ has a common 0-eigenstate. By definition of separable, the decomposition is non-trivial, and thus the new instance $\{h_i^{<j_0>}\}_i$ is strictly simpler in the sense that we strictly decrease the dimension of the qudit q. Note that this method is constructive — the common 0-eigenstate of the new instance $\{h_i^{<j_0>}\}_i$ is also the common 0-eigenstate of the original instance $\{h_i\}_i$.

Our key observation is, for k-qudit-CLHP-projection, even for semi-separable qudit, the **NP** prover is able to perform a similar self-reduction, via a non-constructive way. If we follow the intuition of the separable qudit case, one might try restricting $\mathcal{H}^q \to \mathcal{H}^q_j$, and transform every term to be $\Pi_j h_i \Pi_j$. The problem is that since h_0 does not keep $\mathcal{H}^q_j$ invariant, and does not commute with Π_j, the $\Pi_j h_0 \Pi_j$ is no longer a projection. One may also doubt whether the resulting Hamiltonian is commuting. A more serious problem is that, unlike the case for separable qudit, since h_0 does not keep $\mathcal{H}^q_j$ invariant, it is not clear how to connect the ground states of the original Hamiltonian to the ground states of the new Hamiltonian. We circumvent the problems by slightly changing the construction — rounding $\Pi_j h_0 \Pi_j$ to its 1-eigenspace.

Definition 162 (Reduced Hamiltonian) *Consider a semi-separable qudit q w.r.t commuting projections $\{h_i\}_i$ and a non-trivial decomposition $\mathcal{H}^q = \bigoplus_{j=1}^m \mathcal{H}^q_j$, where Π_j is the projection onto $\mathcal{H}^q_j$. For any j, we define its j-th reduced Hamiltonian to be $\{h_i^{(j)}\}_i$, or written as $H^{(j)} := \sum_i h_i^{(j)}$, where*

- $h_i^{(j)} = \Pi_j h_i \Pi_j$, *for* $i \geq 1$.
- $h_0^{(j)}$ *is the projection onto the* 1*-eigenspace of* $\Pi_j h_0 \Pi_j$. *Assign* $h_0^{(j)}$ *to be* 0 *when the* 1*-eigenspace is empty. It is equivalent to interpret* $h_0^{(j)}$ *is obtained by rounding all the strictly-smaller-than-1-eigenvalues of* $\Pi_j h_0 \Pi_j$ *to 0.*
- *We restrict the space of q from $\mathcal{H}^q$ to $\mathcal{H}^q_j$. Note that all terms $h_i^{(j)}$ including $h_0^{(j)}$ keeps $\mathcal{H}^q_j$ invariant, thus this restriction of space is well-defined. In summary, the original Hamiltonian H acts on $\mathcal{H}^q \otimes \left(\otimes_{q' \neq q} \mathcal{H}^{q'}\right)$, the j-th reduced Hamiltonian $H^{(j)}$ acts on $\mathcal{H}^q_j \otimes \left(\otimes_{q' \neq q} \mathcal{H}^{q'}\right)$.*

Note that the construction of reduced Hamiltonian is consistent with our previous intuition for the separable qudit — If q is separable and h_0 also keeps $\mathcal{H}^q_j$ invariant, then $\Pi_j h_0 \Pi_j$ is a projection thus $h_0^{(j)} = \Pi_j h_0 \Pi_j$. It is worth noting that the reduced Hamiltonian keeps the "geometry" of the original Hamiltonian.

Lemma 163 *In Def. 162, if h_i acts trivially on qudit q' w.r.t space $\mathcal{H}^{q'}$, then $h_i^{(j)}$ acts trivially on qudit q' w.r.t space $\mathcal{H}^{q'}$ if $q' \neq q$ or $\mathcal{H}^{q'}_j$ if $q' = q$. In particular, if $H = \sum_i h_i$ is k-local (or on 2D), so does the j-th reduced Hamiltonian $H^{(j)} = \sum_i h_i^{(j)}$.*

Proof: We prove that if h_0 acts trivially on qudit q', then so does $h_0^{(j)}$, the proof for $i \neq 0$ is similar. By assumption, h_0 acts trivially on q', thus $h_0 = I_{\mathcal{H}^{q'}} \otimes h$ for some projection h. Recall that q is the semi-separable qudit in Def. 162. If $q' = q$, then

$$\Pi_j h_0 \Pi_j = \Pi_j \otimes h \tag{6.4}$$

$$= I_{\mathcal{H}_j^{q'}} \otimes h, \tag{6.5}$$

is a projection and acts trivially on $\mathcal{H}_j^{q'}$. If $q' \neq q$, $\Pi_j h_0 \Pi_j = I_{\mathcal{H}^{q'}} \otimes \Pi_j h \Pi_j$, the 1-eigenspace of $\Pi_j h_0 \Pi_j$ is also of form $I_{q'} \otimes ...$ thus acts trivially on q'. ∎

Besides, the terms in the reduced Hamiltonian are commuting projections.

Lemma 164 *If in Def. 162, $\{h_i\}_i$ are commuting projections, then for any j, the j-th reduced Hamiltonian $\{h_i^{(j)}\}_i$ are commuting projections.*

Proof: Notice that, by the definition of semi-separable, we have $[h_i, \Pi_j] = 0, \forall i \neq 0$. It is also important to keep in mind that $[h_0, \Pi_j]$ might not equal 0.

Firstly we can check that all terms $\{h_i^{(j)}\}_i$ are projections. Notice that $h_0^{(j)}$ is a projection by definition. For $i \neq 0$, since h_i is a projection, and $[h_i, \Pi_j] = 0$, we know $h_i^{(j)} := \Pi_j h_i \Pi_j$ is a projection [3]. In summary, all the terms are projections.

Then we prove that all terms are commuting. Notice that for any $i \neq 0$, for any i', where i' can be 0, and we have

$$(\Pi_j h_i \Pi_j)(\Pi_j h_{i'} \Pi_j) = (\Pi_j \Pi_j)(\Pi_j h_i h_{i'} \Pi_j) \tag{6.6}$$

$$= (\Pi_j \Pi_j)(\Pi_j h_{i'} h_i \Pi_j \Pi_j) \tag{6.7}$$

$$= (\Pi_j \Pi_j)(\Pi_j h_{i'} \Pi_j h_i \Pi_j) \tag{6.8}$$

$$= (\Pi_j h_{i'} \Pi_j)(\Pi_j h_i \Pi_j), \tag{6.9}$$

where Eq. (6.6) is from $[h_i, \Pi_j] = 0$, Eq. (6.7) is from $[h_i, h_{i'}] = 0$ and $\Pi_j^2 = \Pi_j$, Eq. (6.8) is from $[h_i, \Pi_j] = 0$. Note that we never assume that $h_{i'}$ commutes with Π_j.

From the Eq. (6.9), we know $[h_i^{(j)}, h_{i'}^{(j)}] = 0$ if $i \neq 0, i' \neq 0$. Besides, we know for $i \neq 0$, $[h_i^{(j)}, \Pi_j h_0 \Pi_j] = 0$. Thus $h_i^{(j)}$ keeps the 1-eigenspace [4] of $\Pi_j h_0 \Pi_j$

[3] The most direct way to understand this is imagining Π_j, h_i are diagonal 0, 1 matrix, since they are commuting they can be diagonalized simultaneously.

[4] We emphasize it is the space spanned by all 1-eigenvector, rather than one of the eigenvector.

invariant. Since $h_i^{(j)}$ is Hermitian, this implies $h_i^{(j)}$ commutes with the projection onto this 1-eigenspace of $\Pi_j h_0 \Pi_j$, thus $[h_i^{(j)}, h_0^{(j)}] = 0$. In summary, all terms are commuting. ∎

In summary, we have

Corollary 165 (of Lemma 163 and Lemma 164) *If $\{h_i\}_i$ are k-local (or on 2D) qudit commuting projections, then so does the j-th reduced Hamiltonian $\{h_i^{(j)}\}_i$.*

In addition, we give a cute lemma – Lemma 166. In the lemma description, the right side of Eq. (6.10) is just rounding all non-zero coefficients $(1-\lambda)$ to 1. This Lemma is simple itself but captures the key idea of "rounding" used in Lemma 167. It will explain why we can round all non-1 eigenvalue of $\Pi_j h_0 \Pi_j$ to 0, and only use the 1-eigenspace.

Lemma 166 *Let $f(j,\lambda)$ be a non-negative function. Then*

$$\sum_j \sum_{\lambda \le 1} (1-\lambda) f(j,\lambda) > 0 \text{ iff } \sum_j \sum_{\lambda < 1} f(j,\lambda) > 0. \tag{6.10}$$

Proof: Since $f(j,\lambda)$ is non-negative. It suffices to notice that both the left and the right inequalities are equivalent to $\exists j, \exists \lambda < 1$ s.t. $f(j,\lambda) > 0$. ∎

Now we are prepared to state our key lemma, which connects the original Hamiltonian to the reduced Hamiltonians. Inspired by Schuch's idea [Sch11], we will decompose a non-negative term into summation over many non-negative terms. However, our method here uses very different decomposition rules, and has key differences from his, which will be discussed in more detail after the proof.

Lemma 167 (Decrease dimension and rounding) *Consider an instance of k-qudit-CLHP-projection on n qudits, where the k-local Hamiltonian is denoted as $H = \sum_i h_i$ for commuting projections $\{h_i\}_i$. Suppose there is a semi-separable qudit q w.r.t. $\{h_i\}_i$ and non-trivial decomposition $\mathcal{H}^q = \bigoplus_{j=1}^m \mathcal{H}_j^q$. For every j, define the j-th reduced Hamiltonian $H^{(j)} = \sum_i h_i^{(j)}$ as in Definition 162. Then*

$$\lambda(H) = 0 \quad \text{iff} \quad \exists j \text{ s.t } \lambda(H^{(j)}) = 0. \tag{6.11}$$

Proof: Denote the n-qudit space as $\mathcal{H} = \otimes_q \mathcal{H}^q$. Define $\mathcal{H}_j = \mathcal{H}_j^q \otimes_{q' \neq q} \mathcal{H}^{q'}$. For clarity, in this proof, we use I for the identity on $\mathcal{L}(\mathcal{H})$. When using $tr(h)$ we always view h as an operator on $\mathcal{H}$, and we project out all the qudits, i.e. $tr(h) := \sum_i \langle i|h|i\rangle$

where $\{|i\rangle\}_i$ is the computational basis for $\mathcal{H}$. Especially, we view Π_j as an operator in $\mathcal{H}$, while view $I_{\mathcal{H}_j}$ as an operator in $\mathcal{H}_j$.

Note that $\{h_i\}_i$ are commuting projections, proving $\lambda(H) = 0$ is equivalent to show that

$$tr\left[\prod_i (I - h_i)\right] > 0. \tag{6.12}$$

Since $\{h_i\}_i$ are commuting, the relative order in the above formula is unimportant. Recall that Π_j is the projection onto $\mathcal{H}_j^q$. By assumption $\forall i \neq 0$, h_i keeps $\mathcal{H}_j^q$ invariant, and thus

$$\begin{aligned} tr\left[\prod_i (I - h_i)\right] &= tr\left[(I - h_0)\prod_{i\neq 0}\left(\sum_j \Pi_j (I - h_i)\Pi_j\right)\right] \\ &= tr\left[(I - h_0)\sum_j \prod_{i\neq 0}\left(\Pi_j (I - h_i)\Pi_j\right)\right] \\ &= \sum_j tr\left[(I - h_0)\prod_{i\neq 0}\left(\Pi_j (I - h_i)\Pi_j\right)\right] \end{aligned}$$

The first equation is from $\sum_j \Pi_j = I$, and for $i \neq 0$, h_i is Hermitian and keeps $\mathcal{H}_j^q$ invariant thus $\sum_j \Pi_j h_i \Pi_j = h_i$. The second equation is from $\{\Pi_j\}_j$ are orthogonal from each other.

Besides, since $\Pi_j^2 = \Pi_j$ and $tr(M\Pi_j) = tr(\Pi_j M)$ for arbitrary M, we have

$$\begin{aligned} & tr\left[\prod_i (I - h_i)\right] \\ &= \sum_j tr\left[(I - h_0)\,\Pi_j \prod_{i\neq 0}\left(\Pi_j (I - h_i)\Pi_j\right)\Pi_j\right] \\ &= \sum_j tr\left[\Pi_j\,(I - h_0)\,\Pi_j \prod_{i\neq 0}\left(\Pi_j (I - h_i)\Pi_j\right)\right] \\ &= \sum_j tr\left[\Pi_j\left(I - \Pi_j h_0 \Pi_j\right)\Pi_j \prod_{i\neq 0}\left(\Pi_j (I - h_i)\Pi_j\right)\right]. \end{aligned} \tag{6.13}$$

The last equation is from $\Pi_j^2 = \Pi_j$. Note that $\Pi_j h_0 \Pi_j$ is an n-qudit Hermitian operator, and thus it can be diagonalized by a unitary matrix. Consider its eigenvalue

decomposition, and denote the eigenvalues and projections onto the corresponding eigenspace as $\lambda, \Pi_{j,\lambda}$. That is

$$\Pi_j h_0 \Pi_j = \sum_\lambda \lambda \Pi_{j,\lambda}. \tag{6.14}$$

Note that it might be possible that $\Pi_{j,\lambda}$ acts non-trivially on some $q' \neq q$ as long as h_0 acts non-trivially on q'. Besides, by definition of $\Pi_{j,\lambda}$ we have

$$\sum_\lambda \Pi_{j,\lambda} = I. \tag{6.15}$$

Since h_0 is a projection, we have $\Pi_j h_0 \Pi_j \succeq 0$ and $\lambda \in [0,1]$. Use Eqs. (6.14),(6.15), and we have that Eq. (6.13) becomes

$$\begin{aligned}
&tr\left[\prod_i (I - h_i)\right] \\
&= \sum_j tr\left[\left(\Pi_j \left(\sum_{\lambda \leq 1} (1-\lambda)\Pi_{j,\lambda}\right)\Pi_j\right) \prod_{i\neq 0} \left(\Pi_j (I - h_i)\Pi_j\right)\right] \\
&= \sum_j \sum_{\lambda \leq 1} (1-\lambda)\, tr\left[\left(\Pi_j \Pi_{j,\lambda} \Pi_j\right) \prod_{i\neq 0} \left(\Pi_j (I - h_i)\Pi_j\right)\right].
\end{aligned}$$

Define

$$f(j,\lambda) := tr\left[\left(\Pi_j \Pi_{j,\lambda} \Pi_j\right) \prod_{i\neq 0} \left(\Pi_j (I - h_i)\Pi_j\right)\right] \tag{6.16}$$

Since $\{\Pi_j (I - h_i)\Pi_j\}_{i\neq 0}$ are commuting projections, we know $\prod_{i\neq 0} \left(\Pi_j (I - h_i)\Pi_j\right) \succeq 0$ and is Hermitian. Note that $\Pi_j \Pi_{j,\lambda} \Pi_j \succeq 0$ and is Hermitian. Since the trace of the product of two positive semi-definite Hermitian matrices is non-negative [5] , we have that $f(j,\lambda)$ is non-negative,

$$f(j,\lambda) \geq 0, \forall j, \lambda. \tag{6.17}$$

By Lemma 166, $tr\left[\prod_i (I - h_i)\right] > 0$ is equivalent to rounding all the non-zero coefficients in Eq. (6.16) to 1, that is, equivalent as showing that

[5] Let A, B to be arbitrary two Hermitian matrices where $A \succeq 0, B \succeq 0$. Since A, B are Hermitian, consider the eigenvalue decompositions $A = \sum_i a_i |\phi_i\rangle\langle\phi_i|, a_i \geq 0$, $B = \sum_j b_j |\psi_j\rangle\langle\psi_j|, b_j \geq 0$. Then $tr(AB) = \sum_{i,j} a_i b_j |\langle\phi_i|\psi_j\rangle|^2 \geq 0$.

$$\sum_j \sum_{\lambda<1} tr\left[\left(\Pi_j \Pi_{j,\lambda} \Pi_j\right) \prod_{i\neq 0} \left(\Pi_j (I - h_i)\Pi_j\right)\right] > 0 \tag{6.18}$$

$$\Leftrightarrow \sum_j tr\left[\left(\Pi_j \left(\sum_{\lambda<1} \Pi_{j,\lambda}\right) \Pi_j\right) \prod_{i\neq 0} \left(\Pi_j (I - h_i)\Pi_j\right)\right] > 0$$

$$\Leftrightarrow \sum_j tr\left[\left(\Pi_j \left(I - h_0^{(j)}\right) \Pi_j\right) \prod_{i\neq 0} \left(\Pi_j (I - h_i)\Pi_j\right)\right] > 0 \tag{6.19}$$

$$\Leftrightarrow \sum_j tr\left[\left(\Pi_j - h_0^{(j)}\right) \prod_{i\neq 0} \left(\Pi_j (I - h_i)\Pi_j\right)\right] > 0 \tag{6.20}$$

$$\Leftrightarrow \sum_j tr\left[\prod_i \left(\Pi_j - h_i^{(j)}\right)\right] > 0. \tag{6.21}$$

Eq. (6.19) is from Eq. (6.15) and the definition of $h_0^{(j)}$. Eq. (6.20) is from $\Pi_j h_0^{(j)} \Pi_j = h_0^{(j)}$. Note that Eqs. (6.18-6.21) and Eq. (6.17) imply that

$$tr\left[\prod_i \left(\Pi_j - h_i^{(j)}\right)\right] = \sum_{\lambda<1} f(j,\lambda)$$
$$\geq 0.$$

Thus we further have

$$(6.21) \Leftrightarrow \exists j \text{ s.t } tr\left[\prod_i \left(\Pi_j - h_i^{(j)}\right)\right] > 0 \tag{6.22}$$

$$\Leftrightarrow \exists j \text{ s.t } tr^{(j)}\left[\prod_i \left(I_{\mathcal{H}_j} - h_i^{(j)}\right)\right] > 0 \tag{6.23}$$

where in Eq. (6.23), the notation $tr^{(j)}$ means now we restrict the space of qudit q from $\mathcal{H}^q \to \mathcal{H}_j^q$, and the trace is over $\mathcal{H}_j$. Note that Eq. (6.23) is well defined since $\prod_i \left(I_{\mathcal{H}_j} - h_i^{(j)}\right)$ keeps $\mathcal{H}_j$ invariant.

By Lemma 164 we know $\{h_i^{(j)}\}$ are commuting projections on $\mathcal{H}_j = \mathcal{H}_j^q \otimes \left(\otimes_{q'\neq q} \mathcal{H}^q\right)$. Eq. (6.23) is equivalent to say $\exists j$ such that the j-th reduced Hamiltonian $\{h_i^{(j)}\}_i$ has a common 0-eigenvector, where the space of q is $\mathcal{H}_j^q$ and the n-qudit space is $\mathcal{H}_j$. ∎

Corollary 168 *If k-qudit-CLHP-projection without semi-separable qudit is in **NP**, then k-qudit-CLHP-projection is in **NP**.*

Proof: Lemma 167 says when there is a semi-separable qudit, an **NP** prover can efficiently reduce the k-qudit-CLHP-projection to a new k-qudit-CLHP-projection by strictly decreasing the dimension of q. Since $dim(q)$ is a constant, an **NP** prover can reach to a k-qudit-CLHP-projection without semi-separable qudit by repeatedly performing Lemma 167 in $poly(n)$ time. ■

To help better understand Lemma 167, let us discuss the differences between Lemma 167 and the method used in Schuch's paper [Sch11]. We both decompose some of the terms and get a summation of non-negative quantities, but we do such decomposition and projection in different ways. The key difference is that in our method, we guarantee all the quantities still correspond to a commuting local Hamiltonian problem. Our method is more like self-reduction, and we can perform such self-reduction *sequentially* until this are no semi-separable qudits. On the contrary, in Schuch's method, the two projections he used for each qudit [6] , are not commuting with each other, and the quantity does not correspond to commuting local Hamiltonian anymore, and thus this decomposition technique can only be performed once rather than sequentially.

Restrictions on the qutrit-CLHP-2D without semi-separable qudit

From now on we will consider the 2D geometry and start our proof for the qutrit-CLHP-2D-projection is in **NP**. Recall that we allow qudits in the qutrit-CLHP-2D-projection to have different dimensions, i.e. either 1,2 or 3. By Corollary 168, we can assume that there are no semi-separable qudits. This "no semi-separable qudits condition", combined with the 2D geometry, will lead to strong restrictions on the form of the Hamiltonian, i.e. Lemma 170.

We define some notations. As shown in Figure 6.2, when considering the qutrit-CLHP-2D-projection, for a qudit q, we use p_1, p_2 to denote the two plaquette projections in the diagonal direction, p_1', p_2' to denote the two plaquettes projections in the anti-diagonal direction. In the whole Sec. 6.4, the relative positions of q, p_1, p_2, p_1', p_2' will always obey Figure 6.2. We give the following definitions.

Definition 169 *Consider a qutrit-CLHP-2D-projection instance, for any qutrit q of dimension 3, use the notations in Figure 6.2. For $A, B \in \{(1,1,1),(1,2),(3)\}$, we say that p_1, p_2, p_1', p_2' acting on q via $A \times B$-way, if p_1, p_2 commute in A-way [7] on*

[6] Schuch wrote his proof in terms of the qubit, but the first decomposition part also works for qudit.

[7] See Definition 159. More specifically, denote the set of qudits that p_1, p_2 acting non-trivially on as S_1, S_2, then in Definition 159 $\mathcal{H} := \mathcal{H}^q, \mathcal{X} := \otimes_{q' \in S_1/q} \mathcal{H}^{q'}, \mathcal{Z} := \otimes_{q' \in S_2/q} \mathcal{H}^{q'}$.

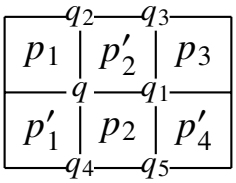


Figure 6.2: Notations for the qutrit-CLHP-2D-projection.

$\mathcal{H}^q$, and p'_1, p'_2 commute in B-way on $\mathcal{H}^q$; or verse visa, i.e. p_1, p_2 commute in B-way on $\mathcal{H}^q$, and p'_1, p'_2 commute in A-way on $\mathcal{H}^q$.

Note that p_1, p_2 only overlap on one qudit – q – and thus the above sentence "p_1, p_2 commute in A-way on $\mathcal{H}^q$" is well-defined. Same for p'_1, p'_2. On the other hand, p_1, p'_1 overlap on two qudits, thus we cannot say p_1, p'_1 commute in some way on $\mathcal{H}^q$. Another clarification is, that it is possible that some of the terms are identity, e.g. $p_1 = p_2 = I$, then the situation does not belong to Definition 169. Those cases will be considered separately and solved easily in the related proofs.

Recall that by Corollary 168, we can assume that there are no semi-separable qudits. This will imply certain ways of commuting cannot exist for the qutrit-CLHP-2D-projection.

Lemma 170 (Legal ways of commuting) *Consider a qutrit-CLHP-2D-projection Hamiltonian $\{p\}_p$, if there is no semi-separable qudit, then there is no qutrit q with $dim(\mathcal{H}^q) = 3$ such that terms p_1, p_2, p'_1, p'_2 acting on q via $(1,2) \times (3)$-way or $(1,2) \times (1,2)$-way.*

Proof: To ease notations, in this proof we use $\mathcal{H}$ to denote the Hilbert space of q, instead of using $\mathcal{H}^q$. With contradiction, assume that there is a qudit q with $dim(\mathcal{H}^q) = 3$ such that

- p_1, p_2 commute via $(1,2)$-way on $\mathcal{H}^q$. The decomposition w.r.t p_1, p_2 is $\mathcal{H} = \mathcal{H}_1 \bigoplus \mathcal{H}_2$, where

$$dim(\mathcal{H}_1) = 1, \mathcal{H}_1 = span(|\psi\rangle)$$

[8] for some $|\psi\rangle \in \mathcal{H}^q$, and

$$dim(\mathcal{H}_2) = 2, \mathcal{H}_2 = \mathcal{H}_2^1 \otimes \mathcal{H}_2^2.$$

[8] One may wonder how could $\mathcal{H}_1$ written as $\mathcal{H}_1^1 \otimes \mathcal{H}_1^2$ in the Structure Lemma. Conceptually one can interpret $\mathcal{H}_1^1 = span\{|\psi\rangle\}$, and $\mathcal{H}_1^2$ as a one-dimensional space as scalars $\{c\}_c$.

Since 2 is a prime and $2 = 2 \times 1$, by definition the decomposition $\mathcal{H} = \mathcal{H}_1 \bigoplus \mathcal{H}_2$ is induced by p_1 or p_2, and $[p_1, p_2] = 0$, by Corollary 157, one can check that one of p_1, p_2 must act trivially on $\mathcal{H}_2$. W.l.o.g assume that

$$dim(\mathcal{H}_2^1) = 2, dim(\mathcal{H}_2^2) = 1,$$

and p_2 acts trivially on $\mathcal{H}_2$.

- For p_1', p_2', consider the following cases:

  (a) p_1', p_2' commute via (3)-way. In this case one of p_1', p_2' must act trivially on q. W.l.o.g assume that p_1' is the term.

  (b) p_1', p_2' commute via $(1, 2)$-way. Similarly as above notaions for p_1, p_2 we have

  $$\mathcal{H} = \mathcal{H}_1' \bigoplus \mathcal{H}_2',$$

  and define $|\psi'\rangle, \mathcal{H}_i'$ similarly. And similarly assume that p_2' acts trivially on $\mathcal{H}_2'$.

□ **Case (a):** We will prove q is semi-separable, which leads to a contradiction. Consider the decomposition from $(1, 2)$-way for p_1, p_2, that is $\mathcal{H} = \mathcal{H}_1 \bigoplus \mathcal{H}_2$. Since p_1' acts trivially on q, it keeps those subspaces invariant as well. In summary, all terms but p_2' keep the decomposition invariant. Since the decomposition $\mathcal{H} = \mathcal{H}_1 \bigoplus \mathcal{H}_2$ is non-trivial, by definition q is semi-separable.

□ **Case (b):** Consider term p_2, by definition, p_2 is Hermitian, keeps $\mathcal{H}_1, \mathcal{H}_2$ invariant and acts trivially on $\mathcal{H}_2$. We can write

$$p_2 = |\psi\rangle \langle\psi| \otimes A + (I_q - |\psi\rangle \langle\psi|) \otimes B. \tag{6.24}$$

Here I_q is the identity on $\mathcal{H}^q$. And A, B (might be 0) act non-trivially at most on the remaining three qudits q_1, q_4, q_5, as in Figure 6.2. Rewriting $h := A - B, \hat{h} = B$, we have

$$p_2 = |\psi\rangle \langle\psi| \otimes h + I_q \otimes \hat{h}. \tag{6.25}$$

Since p_2 is Hermitian, we know $h, \hat{h}$ are Hermitian. Similarly, we can write

$$p_2' = |\psi'\rangle \langle\psi'| \otimes h' + I_q \otimes \hat{h}', \tag{6.26}$$

where $h', \hat{h}'$ are Hermitian and act non-trivially at most on q_1, q_2, q_3. If $h = 0$ then p_2 acts trivially on q. Using a similar argument as case (a) we conclude q is semi-separable. The case for $h' = 0$ is similar. So w.l.o.g, we assume that

$$h \neq 0, h' \neq 0. \tag{6.27}$$

In the following, we are going to prove that either q or q_1 is semi-separable, which will lead to a contradiction. W.l.o.g assume that both $|\psi\rangle, |\psi'\rangle$ are unit vectors.

- If $|\psi\rangle = e^{i\theta} |\psi'\rangle$ for some θ: then by definition, p_1, p_2, p_1', p_2' keep $\mathcal{H}_1, \mathcal{H}_2$ invariant. Thus q is separable.
- If $|\psi\rangle \perp |\psi'\rangle$: then one can verify that both p_2, p_2' keeps $\mathcal{H}_1, \mathcal{H}_2$ invariant, since p_1 also keeps $\mathcal{H}_1, \mathcal{H}_2$ invariant. By definition we know q is semi-separable.
- If $|\langle\psi|\psi'\rangle| \neq 0, \neq 1$: notice that

$$\begin{aligned} p_2 p_2' &= \langle\psi|\psi'\rangle |\psi\rangle \langle\psi'| \otimes hh' + |\psi\rangle \langle\psi| \otimes h\hat{h}' \\ &\quad + |\psi'\rangle \langle\psi'| \otimes \hat{h}h' + I \otimes \hat{h}\hat{h}', \\ p_2' p_2 &= \langle\psi'|\psi\rangle |\psi'\rangle \langle\psi| \otimes h'h + |\psi'\rangle \langle\psi'| \otimes h'\hat{h} \\ &\quad + |\psi\rangle \langle\psi| \otimes \hat{h}'h + I \otimes \hat{h}'\hat{h}. \end{aligned}$$

  Since $dim(q) = 3$, there exists $|\zeta\rangle \neq 0 \in \mathcal{H}^q$ s.t

$$|\zeta\rangle \perp |\psi\rangle, |\zeta\rangle \perp |\psi'\rangle.$$

  By $p_2 p_2' = p_2' p_2$ we have $\langle\zeta|p_2 p_2'|\zeta\rangle = \langle\zeta|p_2' p_2|\zeta\rangle$, and thus

$$\hat{h}\hat{h}' = \hat{h}'\hat{h}. \tag{6.28}$$

- Since $|\langle\psi|\psi'\rangle| \neq 1$, there exists $|\phi\rangle$ such that

$$|\phi\rangle \perp |\psi\rangle, |\phi\rangle \not\perp |\psi'\rangle.$$

  By Eq. (6.28) and $\langle\phi|p_2 p_2'|\phi\rangle = \langle\phi|p_2' p_2|\phi\rangle$ we have

$$\hat{h}h' = h'\hat{h}. \tag{6.29}$$

- Similarly to the above point, we have

$$h\hat{h}' = \hat{h}'h. \tag{6.30}$$

- Finally by Eqs. (6.28) (6.29) (6.30) and $p_2 p_2' = p_2' p_2$ we have

$$\langle\psi|\psi'\rangle\,|\psi\rangle\,\langle\psi'|\otimes hh' = \langle\psi'|\psi\rangle\,|\psi'\rangle\,\langle\psi|\otimes h'h. \tag{6.31}$$

  Left multiplying $\langle\phi|$ to both sides of Eq. (6.31), and use the fact that $\langle\psi'|\psi\rangle \neq 0, \langle\phi|\psi\rangle = 0, \langle\phi|\psi'\rangle \neq 0$, we conclude that

$$h'h = 0.$$

  Similarly, we would get

$$hh' = 0.$$

In summary, we showed that

(i) $h, \hat{h}, h', \hat{h}'$ are Hermitian.

(ii) $\{h, \hat{h}\}$ commute with $\{h', \hat{h}'\}$.

(iii) $hh' = h'h = 0$.

To ease the notations, we abbreviate the induced algebra of Hermitian term p on q, i.e. $\mathcal{A}_p^{\mathcal{H}^q}$, as $\mathcal{A}_p^q$.

Intuitively the above $(i)(ii)(iii)$ say that $\mathcal{A}_{p_2}^{q_1}$ and $\mathcal{A}_{p_2'}^{q_1}$ should commute with each other, and some elements are orthogonal to each other. We will show that this implies q_1 is semi-separable. In the following we write down a careful proof, especially because $h, \hat{h}$ are operators which might act non-trivially on three qudits q_1, q_4, q_5 instead of only on q_1.

Let $\{|i\rangle\langle j|^{q_4,q_5}\}_{ij}$ be the computational basis for q_4, q_5, similarly $\{|k\rangle\langle l|^{q_2,q_3}\}_{kl}$ for q_2, q_3. Consider the decomposition of $h, \hat{h}$, we rewrite p_2 as

$$\begin{aligned} p_2 = |\psi\rangle\,\langle\psi|\otimes\sum_{ij} h_{ij}^{q_1}\otimes|i\rangle\langle j|^{q_4,q_5} \\ + I_q\otimes\sum_{ij}\hat{h}_{ij}^{q_1}\otimes|i\rangle\langle j|^{q_4,q_5}, \end{aligned}$$

where $\{|i\rangle\langle j|^{q_4,q_5}\}_{ij}$ are linearly independent. Since $\{|\psi\rangle\,\langle\psi|, I_q\}$ are linearly independent, we know

$$\{|\psi\rangle\,\langle\psi|\otimes|i\rangle\langle j|^{q_4,q_5}\}_{ij}\cup\{I\otimes|i\rangle\langle j|^{q_4,q_5}\}_{ij}$$

are linearly independent. By lemma 154, $\mathcal{A}_{p_2}^{q_1}$ is generated by $\{h_{ij}^{q_1}\}_{ij} \cup \{\hat{h}_{ij}^{q_1}\}_{ij}$. Define similar notations for p_2', we will have $\mathcal{A}_{p_2'}^{q_1}$ is generated by $\{h_{kl}^{\prime q_1}\}_{kl} \cup \{\hat{h}_{kl}^{\prime q_1}\}_{kl}$.

Note that $\{q_4, q_5\}$ and $\{q_2, q_3\}$ are disjoint, the fact that $[h, h'] = 0$ implies the two sets $\{h_{ij}^{q_1}\}_{ij}$ and $\{h_{kl}^{'q_1}\}_{kl}$ commute. Similarly, (ii) implies the generators of $\mathcal{A}_{p_2}^{q_1}$ and $\mathcal{A}_{p_2'}^{q_1}$ commute, thus $\mathcal{A}_{p_2}^{q_1}$ and $\mathcal{A}_{p_2'}^{q_1}$ commute. If we name the 4 terms on q_1 as p_2', p_2, p_3, p_4' as in Figure 6.2. Let $\widetilde{\mathcal{H}}$ be the Hilbert space of q_1, consider the decomposition $\widetilde{\mathcal{H}} = \bigoplus_i \widetilde{\mathcal{H}}_i$ induced by the induced algebra of p_2 on q_1, i.e. $\mathcal{A}_{p_2}^{q_1}$. By Corollary 157 we know all $\mathcal{A}_p^{q_1}$, $p \neq p_4'$ keeps the decomposition invariant. Thus all terms expect that p_4' keeps the decomposition $\widetilde{\mathcal{H}} = \bigoplus_i \widetilde{\mathcal{H}}_i$ invariant.

Furthermore, (iii) implies

$$h_{ij}^{q_1} h_{kl}^{\prime q_1} = 0, \forall i, j, k, l.$$

By Eq. (6.27) we assume that $h \neq 0, h' \neq 0$, thus there exist ij, kl such that

$$h_{ij}^{q_1} \neq 0, h_{kl}^{\prime q_1} \neq 0.$$

Consider Lemma 158, let

$$\mathcal{A} := \mathcal{A}_{p_2}^{q_1}, \mathcal{A}' := \mathcal{A}_{p_2'}^{q_1},$$

we know the previous decomposition $\widetilde{\mathcal{H}} = \bigoplus_i \widetilde{\mathcal{H}}_i$ induced by $\mathcal{A} = \mathcal{A}_{p_2}^{q_1}$ is non-trivial.

Combining the implications of $(i)(ii)(iii)$, we conclude q_1 is semi-separable, which leads to a contradiction. ∎

Schuch's method and its extensions

Schuch [Sch11] proved that the qubit-CLHP-2D-projection is in **NP**. In this section, we illustrate that his idea can be generalized to prove that a subclass of qudit-CLHP-2D-projection is in **NP**, see Theorem 175. In the next section, i.e. Sec. 6.4, we will show that the qutrit-CLHP-2D-projection without semi-separable qudits falls into this subclass.

The proof for Theorem 175 is similar to [Sch11]. The main difference is that we generalize the definitions of removable qudits from Lemma 173 to Lemma 172. This generalization brings some subtlety so we write the proof in detail even though the proof proceeds in a similar way as [Sch11].

Definition 171 (Removable qudit) *For qudit-CLHP-2D-projection, consider a qudit q. Denote the terms acting on it as p_1, p_2, p'_1, p'_2, as shown in Figure 6.2. Denote $\mathcal{H}^q$ as $\mathcal{H}$. Suppose there exists a decomposition $\mathcal{H} = \bigoplus_i \mathcal{H}_i$ such that both p_1, p_2 keep the decomposition invariant. Let P_i be the projection onto $\mathcal{H}_i$. Similarly define the decomposition $\mathcal{H} = \bigoplus_j \mathcal{H}'_j$ and P'_j for for p'_1, p'_2. We say a qudit q is removable if one of the following holds:*

(i) $\forall i, j$, *at most one of* p_1, p_2 *acts non-trivially on* $\mathcal{H}_i$, *at most one of* p'_1, p'_2 *acts non-trivially on* $\mathcal{H}'_j$, *and* $rank(P_i P'_j) \leq 1$ [9].

(ii) p'_1, p'_2 *act trivially on* $\mathcal{H}$. *Besides,* $\mathcal{H}$ *has a tensor-product structure as* $\mathcal{H} = \hat{\mathcal{H}}_1 \otimes \hat{\mathcal{H}}_2$, *such that* $p_1 \in \mathcal{L}(\hat{\mathcal{H}}_1) \otimes \mathcal{I}_{\hat{\mathcal{H}}_2}$, $p_2 \in \mathcal{I}_{\hat{\mathcal{H}}_1} \otimes \mathcal{L}(\hat{\mathcal{H}}_2)$. *Or similar conditions hold when exchanging the name of* p_1, p_2 *with* p'_1, p'_2.

We give some examples of removable qudits.

Lemma 172 *For qudit-CLHP-2D-projection, for any qudit q, if p_1, p_2 commute in $(1, 1, ..., 1)$-way on $\mathcal{H}^q$, p'_1, p'_2 commute in $(d'_1, ..., d'_t)$-way on $\mathcal{H}^q$ where d'_i is a prime, $\forall i$. Then q is removable.*

Proof: Denote $\mathcal{H} := \mathcal{H}^q$. Let $\mathcal{H} = \bigoplus_i \mathcal{H}_i$ be the decomposition w.r.t to p_1, p_2 and $(1, 1, ..., 1)$-way. Let $\mathcal{H}' = \bigoplus_i \mathcal{H}'_i$ be the decomposition w.r.t to p'_1, p'_2 and $(d'_1, ..., d'_t)$-way. Let P_i,P'_j be the projections onto $\mathcal{H}_i, \mathcal{H}'_j$.

By Definition 159, the way of commuting is obtained by the Structure Lemma. Then by Corollary 157, we know $\mathcal{H}'_j$ has a tensor-product structure as $\mathcal{H}'_j = \mathcal{H}'^1_j \otimes \mathcal{H}'^2_j$, where

$$p'_1 = \bigoplus_i \mathcal{L}(\mathcal{H}'^1_j) \otimes I_{\mathcal{H}'^2_j} \tag{6.32}$$

$$p'_2 \subseteq \bigoplus_i I_{\mathcal{H}'^1_j} \otimes \mathcal{L}(\mathcal{H}'^2_j). \tag{6.33}$$

Since d'_j is a prime, we know that at most one of p'_1, p'_2 acts non-trivially on $\mathcal{H}'_j$. Similarly, since 1 is a prime, at most one of p_1, p_2 acts non-trivially on $\mathcal{H}_i$. Furthermore, note that $rank(P_i) = 1, \forall i$. Thus $rank(P_i P'_j) \leq \min\{rank(P_i), rank(P'_j)\} = 1$. ■

Lemma 173 *For the qubit-CLHP-2D-projection, for any qubit q, if one of $\{p_1, p_2\}$ or $\{p'_1, p'_2\}$ commute in $(1, 1)$-way on $\mathcal{H}^q$, then q is removable.*

[9] The rank is w.r.t viewing P_i, P'_j as local operators in $\mathcal{L}(\mathcal{H}^q)$.

Proof: Denote $\mathcal{H} := \mathcal{H}^q$. W.l.o.g assume p_1, p_2 commute in $(1,1)$-way w.r.t $\mathcal{H} = \bigoplus_i \mathcal{H}_i$. If both p_1', p_2' acts trivially on q, then p_1', p_2' also keep $\mathcal{H} = \bigoplus_i \mathcal{H}_i$ invariant. One can check that Definition. 171 (i) holds. If at least one of p_1', p_2' acts non-trivially on $\mathcal{H}$, the by Lemma 161 p_1', p_2' must commute on $\mathcal{H}$ via $(1,1)$ or (2)-way. By Lemma 172, q is removable. ∎

We name those qudits as removable since we will "remove" them in the proof of Theorem 175. Before proving Theorem 175, we summarize [Sch11]'s result as below. Although written in terms of qubit, [Sch11]'s proof directly works for the following lemma for qudits.

Lemma 174 (One-dimensional structure [Sch11]) *Consider a qudit-CLHP-2D-projection Hamiltonian $H = \sum_p p$ on n qudits. Let S be the set of qudits where $\forall q \in S$, there exist $p \in \{p_1, p_2\}$ and $p' \in \{p_1', p_2'\}$, such that both the induced algebra of p, p' on $\mathcal{H}^q$ are the full algebra $\mathcal{L}(\mathcal{H}^q)$.*

Then for any quantum channels $\{\mathcal{N}_p^q : \mathcal{L}(\mathcal{H}^q) \to \mathbb{C}\}_{p,q}$, the product

$$\prod_p \left(\otimes_{q \in S^C} \mathcal{N}_p^q\right)[I - p]$$

has a one-dimensional structure, and thus

$$tr\left(\prod_p \left(\otimes_{q \in S^C} \mathcal{N}_p^q\right)[I - p]\right)$$

can be computed in classical polynomial time.

Here $\left(\otimes_{q \in S^C} \mathcal{N}_p^q\right)[I - p]$ means applying $\otimes_{q \in S^C} \mathcal{N}_p^q$ on $(I - p)$, which can be interpreted as tracing out qudits in S^c.

Theorem 175 *Consider a qudit-CLHP-2D-projection instance, if for each qudit q, either*

(a) q is removable, or

(b) There exists $p \in \{p_1, p_2\}$ and $p' \in \{p_1', p_2'\}$ such that both the induced algebra of p, p' on $\mathcal{H}^q$ are the full algebra $\mathcal{L}(\mathcal{H}^q)$.

*Then the corresponding qudit-CLHP-2D-projection instance is in **NP**.*

Proof: The proof follows the idea in [Sch11]. The main difference is that we generalize the definitions of removable qudits from Lemma 173 to Lemma 172. This generalization brings some subtlety so we write the proof in detail even though the proof proceeds in a similar way as Schuch's.

Imagine the 2D grid as a chess board and color the plaquettes as black and white. Denote the set of the black plaquettes as $\mathcal{P}_B$, the white plaquettes as $\mathcal{P}_W$. Use the same notations as Definition 171, for any removable qudit q, w.l.o.g assume that p_1, p_2 are black, p_1', p_2' are white. If q satisfies Definition 171 (i), one can notice that

$$\begin{aligned} & tr\left[(I-p_1)(I-p_2)(I-p_1')(I-p_2')\right] && (6.34)\\ =& tr\left[\left(\sum_i P_i(I-p_1)(I-p_2)P_i\right)\left(\sum_j P_j'(I-p_1')(I-p_2')P_j'\right)\right] \\ =& \sum_{i,j} tr\left[\left(P_i(I-p_1)(I-p_2)P_i\right)\left(P_j'(I-p_1')(I-p_2')P_j'\right)\right]. && (6.35)\end{aligned}$$

Note that by definition of removable qudit (i), at most one of p_1, p_2 acts non-trivially on $\mathcal{H}_i$, w.o.l.g assume that it is p_1. This means $P_i(I-p_2)P_i$ is P_i tensor some operator. Formally,

$$\begin{aligned} P_i(I-p_2)P_i &= tr_q(P_i(I-p_2)P_i)/tr_q(P_i)\otimes P_i \\ &= tr_q(P_i(I-p_2)P_i)/tr_q(P_i)\otimes I_q\cdot P_i \end{aligned}$$

Similarly, we assume that p_1' acts non-trivially on $\mathcal{H}_j'$. We have

$$\begin{aligned} & tr\left(P_i(I-p_1)(I-p_2)P_iP_j'(I-p_1')(I-p_2')P_j'\right) && (6.36)\\ &= tr\left(P_i(I-p_1)P_i(I-p_2)P_iP_j'(I-p_1')P_j(I-p_2')P_j'\right) \\ &= tr\left(P_i(I-p_1)\cdot tr_q(P_i(I-p_2)P_i)/tr_q(P_i)\otimes I_q\right. \\ & \left. P_i\cdot P_j'(I-p_1')\cdot tr_q(P_j'(I-p_2')P_j')/tr_q(P_j')\otimes I_q\cdot P_j'\right) \end{aligned}$$

Note that $Tr(MN) = Tr(NM)$ for any M, N. Further, by definition of (i), there exist un-normalized vectors $|\alpha\rangle_q, |\beta\rangle_q$ on q such that $P_iP_j' = |\alpha\rangle_q\langle\beta|_q$. Then:

$$\begin{aligned} =& tr\left(|\beta\rangle_q\langle\alpha|_q(I-p_1)\cdot tr_q(P_i(I-p_2)P_i)/tr_q(P_i)\otimes I_q\right. \\ & \left.|\alpha\rangle_q\langle\beta|_q(I-p_1')\cdot tr_q(P_j'(I-p_2')P_j')/tr_q(P_j')\otimes I_q\right) \\ =& tr\left(\langle\alpha|_q(I-p_1)|\alpha\rangle_q\cdot tr_q(P_i(I-p_2)P_i)/tr_q(P_i)\right. \\ & \left.\langle\beta|_q(I-p_1')|\beta\rangle_q\cdot tr_q(P_j'(I-p_2')P_j')/tr_q(P_j')\right) && (6.37)\end{aligned}$$

Recall that $\{p_1, p_2, p_1', p_2'\}$ are commuting projections. In summary, the above calculations show two things:

- In Eq. (6.35), each quantity, i.e. Eq. (6.36), is the trace of the product of two positive semi-definite Hermitian matrices, and thus each quantity is non-negative. Proving

  $$tr((I - p_1)(I - p_2)(I - p_1')(I - p_2')) > 0$$

  is equivalent to show that one of the quantitys is > 0.

- In Eq. (6.37), we somehow "project out qudit q" for all terms. Then calculating Eq. (6.36) is equivalent to computing the trace of a term without qudit q, i.e. Eq. (6.37). This is why q is named removable.

- Also note that in Eqs. (6.36)-(6.37), we do not change the relative order of $(I - p_i)$ or $(I - p_i')$. This is important when considering multiple removable qudits at the same time.

If q satisfies Definition 171 (ii), we can also "tracing out q". Denote d_q as the dimension of q. Interpret q as two qudits q_1, q_2 w.r.t $\hat{\mathcal{H}}_1, \hat{\mathcal{H}}_2$, we have

$$\begin{aligned} & tr\left[(I - p_1)(I - p_2)(I - p_1')(I - p_2')\right] \\ =& tr\left[tr_q((I - p_1))/d_{q_2} \cdot tr_q(I - p_2)/d_{q_1} \cdot tr_q((I - p_1'))/d_q \cdot tr_q((I - p_2'))/d_q\right] \end{aligned} \tag{6.38}$$

Eqs. (6.34)-(6.38) illustrate how to project out a single removable qudit. Similarly, when calculating $tr(\prod_p (I - p))$ we can project out all the removable qudits. Specifically, we first perform Eq. (6.34)-Eq. (6.35) simultaneously for all removable qudits, and then perform Eq. (6.36)-(6.38) to project out all removable qudits. It is worth noting that we should be careful about the relative orders of each $(I - p)$ — To perform the calculations in Eq. (6.34)-Eq. (6.35), one requires that for each q, terms p_1, p_2 are put in the left, and p_1', p_2' in the right.

To perform such decomposition for all removable qudits simultaneously, it suffices to put all the black terms on the left and all the white terms on the right. Denote the

set of removable qudits as R, we have

$$tr\left[\prod_{p}(I-p)\right] = tr\left[\prod_{p\in\mathcal{P}_B}(I-p)\prod_{p'\in\mathcal{P}_W}(I-p')\right]$$
$$= \sum_{i_q,j_q;\ q\in R} tr\left[\prod_{q\in R}P^q_{i_q}\left(\prod_{p\in\mathcal{P}_B}(I-p)\right)\prod_{q\in R}P^q_{i_q}\prod_{q\in R}P'^q_{j_q}\left(\prod_{p\in\mathcal{P}_W}(I-p)\right)\prod_{q\in R}P'^q_{j_q}\right] \tag{6.39}$$

Then for each removable qudit q, we perform similar operations as Eq. (6.36)-(6.38) to project out q .

Finally for the quantity in Eq. (6.39), we project out all removable qudits for every $(I-p)$. All the remaining qudits originally correspond to type (b) in the Theorem description. By Lemma 174 we know the quantity in Eq. (6.39) can be computed in polynomial time. Note that all terms in $\prod_{p\in\mathcal{P}_B}(I-p)$ are commuting and positive, and similarly for $\prod_{p\in\mathcal{P}_W}(I-p)$. Thus

$$tr\left[\prod_{q\in R}P^q_{i_q}\left(\prod_{p\in\mathcal{P}_B}(I-p)\right)\prod_{q\in R}P^q_{i_q}\prod_{q\in R}P'^q_{j_q}\left(\prod_{p\in\mathcal{P}_W}(I-p)\right)\prod_{q\in R}P'^q_{j_q}\right] \geq 0.$$

In summary, proving $tr(\prod_p(I-p)) > 0$ is equivalent to proving that one of the quantity in Eq. (6.39) is > 0, where the quantity is tracing a product which has a one-dimensional structure, thus can be computed in classical polynomial time. Thus we conclude the qudit-CLHP-2D-projection which satisfies Theorem 175's conditions is in **NP**. ■

Qutrit-CLHP-2D is in NP

Finally, to prove that the qutrit-CLHP-2D is in **NP**, it suffices to notice that the qutrit-CLHP-2D-projection without semi-separable qudit satisfying conditions in Theorem 175.

Lemma 176 *For any qutrit-CLHP-2D-projection instance without semi-separable qudit, every qudit satisfies one of the two conditions in Theorem 175.*

Proof: Consider any qudit q. If $dim(q) = 1$, it is removable due to Definition 171 (i). W.l.o.g. assume $dim(q) \in \{2,3\}$. If p'_1, p'_2 act trivially on $\mathcal{H}^q$, since q is not semi-separable, we know p_1, p_2 must commute via (3)-way or (2)-way.

Furthermore, by the Structure Lemma, i.e. Corollary 157, we know there is a tensor product structure of $\mathcal{H}^q = \mathcal{H}^{q,1} \otimes \mathcal{H}^{q,2}$ such that

$$p_1 \in \mathcal{L}(\mathcal{H}^{q,1}) \otimes \mathcal{I}_{\mathcal{H}^{q,2}},$$
$$p_2 \in \mathcal{I}_{\mathcal{H}^{q,1}} \otimes \mathcal{L}(\mathcal{H}^{q,2}).$$

Thus q is removable due to Definition 171 (ii). A similar argument works when p_1, p_2 act trivially on $\mathcal{H}^q$. Thus w.l.o.g, we assume that at least one of p_1, p_2, and one of p'_1, p'_2 act non-trivially on q. Then

- When $dim(q) = 2$, by Lemma 161 we know either one of $\{p_1, p_2\}$ or $\{p'_1, p'_2\}$ commute in $(1,1)$-way, or both of them commute in (2)-way. For the first case, q is removable due to Lemma 173. For the second case, q satisfies Theorem 175 condition (b).
- When $dim(q) = 3$, since there is no semi-separable qudit, for any qudit q, by Lemma 170 and Lemma 160, we know either one of $\{p_1, p_2\}$ or $\{p'_1, p'_2\}$ commute in $(1,1,1)$-way, then q is removable by Lemma 172. Or both of them commute in (3)-way, then q satisfies Theorem 175 condition (b).

■

Combined with Corollary 192, Corollary 168, Lemma 176 and Theorem 175, we finally conclude that

Corollary 177 *The qutrit-CLHP-2D problem is in **NP**.*

6.5 Factorized commuting local Hamiltonian on 2D

In this section, we give a constructive proof to show that qudit-CLHP-2D-factorized is in **NP**, by proving that qudit-CLHP-2D-factorized is equivalent to a direct sum of qubit stabilizer Hamiltonian (see Theorem 189). Note that in this section we do **not** assume that $\{p\}_p$ are projections. The reason for this is that if we start with an arbitrary qudit-CLHP-2D-factorized Hamiltonian, such as the Toric code, and replace each term with a projection that preserve the kernel (as in Lemma 191), then the new Hamiltonian is not guaranteed to be a factorized projection. For example, if we take a Toric code term such as $h = XXXX$ and replace h with $(IIII - XXXX)/2$, the resulting term is no longer factorized. By contrast, for the qutrit-CLHP-2D, where we do not require that the terms be factorized, the assumption that the terms are projections does not weaken the results due to Corollary 192.

The structure of this section is as follows. In Sec. 6.5 we give notations and definitions. In Sec. 6.5 we prove that if there are no separable qudits, then the Hamiltonian is equivalent to a direct sum of qubit stabilizer Hamiltonian. Finally in Sec. 6.5 we prove Theorem 189.

Notations, definitions and lemmas

Notation. Let h be a Hermitian operator on Hilbert space $\mathcal{H}$. Let $\mathcal{H}'$ be a subspace of $\mathcal{H}$ and suppose h keeps $\mathcal{H}'$ invariant. We define

$$\ker_{\mathcal{H}}(h) := \{|\psi\rangle \in \mathcal{H} \mid h\,|\psi\rangle = 0\}$$

and

$$\ker_{\mathcal{H}'}(h) := \{|\psi\rangle \in \mathcal{H}' \mid h\,|\psi\rangle = 0\}.$$

For two orthogonal subspaces $V, W \subseteq \mathcal{H}$, their direct sum are defined as

$$V \bigoplus W = \{v + w \mid v \in V, w \in W\}.$$

For two Hermitian operators $h, h' \in \mathcal{L}(\mathcal{H})$, we write $hh' = \pm h'h$ if either $hh' = h'h$ or $hh' = -h'h$. We say an n-qudit Hermitian term h is factorized if $h = \otimes_q h^q$ where h^q is Hermitian, $\forall q$. When a factorized Hermitian $h = 0$, we always rewrite h to be tensor of zeros, i.e. $h^q = 0, \forall q$.

We say a Hilbert space $\mathcal{H}$ is equivalent to m-qubit space, denoted as $\mathcal{H} \sim (\mathbb{C}^2)^{\otimes m}$, if there exists tensor-product structure

$$\mathcal{H} = \mathcal{H}_1 \otimes ... \otimes \mathcal{H}_m,$$

where $dim(\mathcal{H}_i) = 2$. When considering Hilbert space $\mathcal{H} \sim (\mathbb{C}^2)^{\otimes n}$, we define Pauli groups as the group of operators generated by $\{I, X, Y, Z\}^{\otimes n}$, with respect the basis which makes $\mathcal{H}$ factorized as $(\mathbb{C}^2)^{\otimes n}$. We denote elements in the Pauli group as Pauli operators. First, we formally define what it means for a Hamiltonian to be equivalent to a qubit stabilizer Hamiltonian or a direct sum of qubit stabilizer Hamiltonians.

Definition 178 (Equivalence to qubit stabilizer) *Consider a commuting (but not necessarily local) Hamiltonian $H = \sum_i h_i$ acting on space $\mathcal{H}_* := \otimes_q \mathcal{H}_*^q$. We say H is equivalent to a qubit stabilizer Hamiltonian on $\mathcal{H}_*$ if the following hold:*

(1) For all q, $\mathcal{H}_*^q \sim (\mathbb{C}^2)^{\otimes m^q}$ for some integer m^q.

(2) *Each h_i acts as a Pauli operator up to phases on $\mathcal{H}_*$, with respect to the basis which makes $\mathcal{H}_* \sim (\mathbb{C}^2)^{\otimes(\sum_q m^q)}$.*

In the above definition, we allow $m_q = 0$, where $dim(\mathcal{H}_*^q) = 1$, and all h_i acts as I up to phases on $\mathcal{H}_*^q$.

Definition 179 (Simple subspace) *Consider an n-qudit space $\mathcal{H} = \otimes_q \mathcal{H}^q$. We say a subspace $\mathcal{H}_*$ of $\mathcal{H}$ is simple, if $\mathcal{H}_*$ is a tensor product of subspace of each qudit, i.e. $\mathcal{H}_* = \otimes_q \mathcal{H}_*^q$ where $\mathcal{H}_*^q$ is a subspace of $\mathcal{H}^q$.*

Definition 180 (Direct sum of qubit stabilizer) *Given a commuting Hamiltonian $H = \sum_i h_i$ acting on space $\mathcal{H} = \otimes_q \mathcal{H}^q$. We say H is equivalent to a direct sum of qubit stabilizer Hamiltonians, if there exists a set of simple subspace $\{\mathcal{H}_* := \otimes_q \mathcal{H}_*^q\}_{*\in P}$ such that*

(1) *$\{\mathcal{H}_*\}_{*\in P}$ are pairwise orthogonal, and $\mathcal{H} = \bigoplus_{*\in P} \mathcal{H}_*$;*

(2) *$\forall * \in P$, H keeps $\mathcal{H}_*$ invariant, $\{h_i\}_i$ keeps $\mathcal{H}_*$ invariant, and H is equivalent to qubit stabilizer Hamiltonian when restricted to $\mathcal{H}_*$.*

Remark 9 *(Terminology of "Equivalent to qubit stabilizer state" used in Theorem 146) Use notations in Definition 180, if an n-qudit Hamiltonian $H = \sum_i h_i$ is equivalent to a direct sum of qubit stabilizer Hamiltonians, there exists a simple subspace $\mathcal{H}_* = \otimes_q \mathcal{H}_*^q$ which contains a ground state of H, denoted as $|\psi_*\rangle$.*

*Since H is equivalent to qubit stabilizer Hamiltonian on $\mathcal{H}_*$, we know $|\psi_*\rangle$ can be chosen to be a qubit stabilizer state w.r.t to the basis which makes $\mathcal{H}_*^q \sim (\mathbb{C}^2)^{\otimes m_q}$ in Definition 178. In this sense we say H has a ground state which is equivalent to qubit stabilizer state. The notion of "equivalent to qubit stabilizer state" is only used for intuitively explaining how we prove that qudit-CLHP-2D-factorized is in **NP** in Theorem 146. To avoid ambiguity, we will **not** use this notion further in the following context.*

We now give the definitions and lemmas for commuting in a singular/regular way. For technical reasons, our definition is slightly different from [BV03] [10] .

Definition 181 *[Singular/regular way] Consider two factorized Hermitian terms $h, \hat{h}$ acting on qudits, with $[h, \hat{h}] = 0$.*

[10] [BV03] defines the case where $h\hat{h} = 0$ and $\forall q, h^q \hat{h}^q = \pm \hat{h}^q h^q$ as commuting in a singular way. We define this case as commuting in a regular way.

- *We say $h, \hat{h}$ commute in a regular way, if $\forall q$, $h^q \hat{h}^q = \pm \hat{h}^q h^q$.*
- *We say $h, \hat{h}$ commute in a singular way if there $\exists$ qudit q, $h^q \hat{h}^q \neq \pm \hat{h}^q h^q$.*

Recall that when $h = 0$, we always rewrite h to be tensor of zeros. Thus if $h, \hat{h}$ commute in a singular way, then $h \neq 0, \hat{h} \neq 0$. We say a set of factorized Hermitian terms $\{h_i\}_i$ commute in a regular way if $\forall i, j$, the h_i, h_j commute in a regular way. In the following, we introduce a lemma which states how factorized terms can commute with each other.

Lemma 182 (Rephrase of Lemma 9 in [BV03]) *Consider two factorized Hermitian terms $h, \hat{h}$ acting on n qudits, with $[h, \hat{h}] = 0$. If they only share one qudit q, then $[h^q, \hat{h}^q] = 0$. If they share two qudits q_1, q_2, then one of the following holds:*

(1) $h\hat{h} \neq 0$. In this case $h^{q_1} \hat{h}^{q_1} = \pm \hat{h}^{q_1} h^{q_1}$ and $h^{q_2} \hat{h}^{q_2} = \pm \hat{h}^{q_2} h^{q_2}$.

(2) $h\hat{h} = 0$. In this case one of $h^{q_1} \hat{h}^{q_1}$, or $h^{q_2} \hat{h}^{q_2}$ equals to 0.

Corollary 183 *Consider two factorized Hermitian terms $h, \hat{h}$ acting on n qudits, with $[h, \hat{h}] = 0$. If $h, \hat{h}$ share two qudits q_1, q_2, and commute in a singular way, then one of $h^{q_1} \hat{h}^{q_1}$, or $h^{q_2} \hat{h}^{q_2}$ equals to 0. For the other one qudit, denoted as q, we have $h^q \hat{h}^q \neq \pm \hat{h}^q h^q$.*

We also prove some useful lemmas.

Lemma 184 *Consider two Hermitian terms $h, \hat{h}$ acting on a Hilbert space $\mathcal{H}$. If $h\hat{h} = \alpha \hat{h} h$ for some $\alpha \in \mathbb{R}$, then h keeps $\ker_{\mathcal{H}}(\hat{h})$ invariant.*

Proof: It suffices to notice that $\forall \, |\psi\rangle \in \ker_{\mathcal{H}}(\hat{h})$, we have

$$\hat{h} h(|\psi\rangle) = \alpha h \hat{h} \, |\psi\rangle = 0.$$

This implies that $h \, |\psi\rangle \in \ker_{\mathcal{H}}(\hat{h})$. ∎

Lemma 185 *Consider a qudit q and Hermitian terms $\hat{h}^q \neq 0$ and $\{h_i^q\}_i$ acting on $\mathcal{H}^q$. Suppose that*

$$\forall i, \hat{h}^q h_i^q = \pm h_i^q \hat{h}^q,$$

and furthermore there exists i_0 such that $h_{i_0}^q \neq 0$, and $\hat{h}^q h_{i_0}^q = 0$. Then q is separable with respect to $\{\hat{h}^q\} \cup \{h_i^q\}_i$.

Proof: Define $\mathcal{H}_1 := \ker(\hat{h})$. Since both $\hat{h}, h_{i_0}$ are non-zero and $\hat{h}h_{i_0} = 0$, we know $\mathcal{H}_1 \neq \{0\}$ and $\mathcal{H}_1 \neq \mathcal{H}^q$. This implies the decomposition $\mathcal{H}^q = \mathcal{H}_1 \bigoplus \mathcal{H}_1^\perp$ is non-trivial. By lemma 184, we know that all operators in $\{\hat{h}\} \cup \{h_i\}_i$ keep $\mathcal{H}_1$ invariant. Since they are Hermitian, they also keep $\mathcal{H}_1^\perp$ invariant. Thus q is separable. ■

A weaker version without separable qudits

In this section we prove in Theorem 188 that if there are no separable qudits, then qudit-CLHP-2D-factorized is equivalent to qubit stabilizer Hamiltonian. In particular, we prove that in this situation, all the terms must commute in a regular way.

Lemma 186 *Consider qudit-CLHP-2D-factorized Hamiltonian $H = \sum_p p$ acting on n qudits. If there are no separable qudits, then all the terms $\{p\}_p$ commute in a regular way. Moreover, for any qudit q and any term $p, \hat{p} \in \{p_1, p_2, p_1', p_2'\}$ acting on q, as shown in Figure 6.3 (a), we have*

(1) $(p^q)^2 = c_{pq} I_q$ for some constant c_{pq}.

(2) $p^q \hat{p}^q = \pm \hat{p}^q p^q$.

(3) The C^*-algebra generated by $\{p^q\}_{p \in \{p_1, p_2, p_1', p_2'\}}$ is the whole algebra $\mathcal{L}(\mathcal{H}^q)$.

Proof: We first define some notations to help the illustration. To match the notations in Definition 178, we denote the Hilbert space of qudit q as $\mathcal{H}_*^q$, and the n-qudit space as $\mathcal{H}_* := \otimes_q \mathcal{H}_*^q$.

As shown in Figure 6.3 (a), Consider a qudit q, we denote the terms acting on q as p_1, p_2, p_1', p_2'. Recall that all the terms are factorized, thus we can use p_1^q to represent the factor of p_1 on q. Use similar notations for other terms. For any two terms

$$h, \hat{h} \in \{p_1, p_2, p_1', p_2'\},$$

we use symbols $-, \underline{0}, \times$ to represent the relationship between $h^q, \hat{h}^q$.

- "$-$": If $h^q \hat{h}^q = \pm \hat{h}^q h^q$, and $h, \hat{h}$ commute in a regular way.
- "$\underline{0}$": If $h^q \hat{h}^q = 0$, and $h, \hat{h}$ commute in a singular way.
- "$\times$": If $h^q \hat{h}^q \neq \pm \hat{h}^q h^q$, and $h, \hat{h}$ commute in a singular way.

Using the above symbols, we will draw a graph to represent the relationship of $\{p_1, p_2, p_1', p_2'\}$ on qudit q. For example, the graph in Figure 6.3 (f) means: The relationship between p_1^q and p_2^q is "$-$". That is

$$p_1^q p_2^q = \pm p_2^q p_1^q,$$

and p_1, p_2 commute in a regular way. Similar for $p_1^{'q}$ and $p_2^{'q}$. The relationship between p_1^q and $p_1^{'q}$ is "$\underline{0}$". That is

$$p_1^q p_1^{'q} = 0,$$

and p_1, p_1' commute in a singular way. Similar for $p_2^{'q}$ and p_2^q. The relationship for $p_1^q, p_2^{'q}$ is "$\times$". That is

$$p_1^q p_2^{'q} \neq \pm p_2^{'q} p_1^q,$$

and p_1, p_2' commute in a singular way. Similar for $p_1^{'q}$ and p_2^q. We use $\#_q\underline{0}$ to represent the number of $\underline{0}$ in the graph for q, and similar for symbol "$\times$". For example, in Figure 6.3 (f), we have

$$\#_q\underline{0} = \#_q\times = 2.$$

For each q, we draw such a graph and assign $-, \underline{0}, \times$ to each pair

$$h, \hat{h} \in \{p_1^q, p_2^q, p_1^{'q}, p_2^{'q}\}.$$

For q in the boundary [11] of the lattice, some terms might be missing. We draw the graph for the existing terms similarly.

We use $\#\underline{0}$ to represent the total number of $\underline{0}$ when considering all qudits, that is

$$\#\underline{0} = \sum_q \#_q\underline{0}.$$

Similarly for $\#\times$. Note that by Corollary 183, we have

$$\#\underline{0} = \#\times.$$

Now we are prepared to prove Lemma 186. We first prove the following.

Claim 1: There is no qudit q such that two of the terms acting on q, i.e. two of p_1, p_2, p_1', p_2' in Figure 6.3(a), can commute in a singular way.

[11] Here we mean the physical boundary of the lattice, not the (virtual) boundary defined in [AKV18].

With contradiction suppose there are such qudits, then

$$\#\underline{0} = \#\times \geq 1.$$

Then there must be a qudit q such that

$$\#_q\underline{0} \geq \#_q\times \geq 1.$$

We will prove that q is separable thus leading to a contradiction. Here we suppose q is in not on the boundary, thus there are four terms p_1, p_2, p_1', p_2' acting on q. The case where q is on the boundary and some terms are missing can be analyzed similarly.

Note that we always have

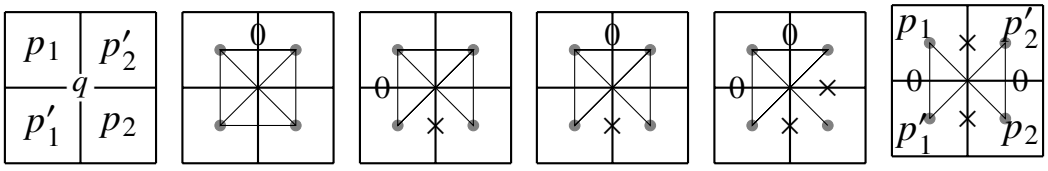
$$p_1^q p_2^q = p_2^q p_1^q$$

since p_1, p_2 only shares one qudit. Similar for $p_1^{'q}$ and $p_2^{'q}$. One can check that up to rotating the lattice, the relationships of terms on q must be related to one of the graphs in Figure 6.3 (b-f). Please read the captions of Figure 6.3 for a precise description of the classification. Note that if two terms $h, \hat{h}$ on q satisfy $\underline{0}$, by definition of commuting in a singular way, we have $p \neq 0, \hat{p} \neq 0$, and thus $p^q \neq 0, \hat{p}^q \neq 0$.

For Figure 6.3 (b-e), let

$$\hat{h}^q := p_1^q, \{h_i^q\}_i := \{p_2^q, p_1^{'q}, p_2^{'q}\}.$$

By Lemma 185, we know q is separable. For Figure 6.3 (f), let $\mathcal{A}$ to be the C^*-algebra generated by $p_1^q, p_2^{'q}$, $\mathcal{A}'$ to be the C^*-algebra generated by $p_1^{'q}, p_2^q$. By Lemma 158 we know q is separable.



(a) (b) (c) (d) (e) (f)

Figure 6.3: Relationships of factors on q. The classification is organized in the increasing order of $\#_q\times$. The (b) case corresponds to when $\#_q\times = 0, \#_q\underline{0} = 1$. Other cases when $\#_q\times = 0, \#_q\underline{0} \geq 1$ can be handled in the same way. The (c)(d) cases correspond to when $\#_q\times = 1, \#_q\underline{0} = 1$. Other cases when $\#_q\times = 1, \#_q\underline{0} \geq 1$ can be handled in the same way. The (e)(f) cases correspond to when $\#_q\times = 2, \#_q\underline{0} = 2$.

Claim 2: The conditions (1)(2)(3) in Lemma 186 hold. Consider arbitrary qudit q, let $\mathcal{A}$ be the C^*-algebra generated by $\{p^q\}_{p\in\{p_1,p_2,p_1',p_2'\}}$. Since q is not separable, by lemma 155, we know the the center of $\mathcal{A}$, i.e.$\mathcal{Z}(\mathcal{A})$, must be trivial. Besides, lemma 155 also implies

$$\mathcal{H}^q = \mathcal{H}_1 \otimes \mathcal{H}_2, \mathcal{A} = \mathcal{L}(\mathcal{H}_1) \otimes \mathcal{I}_{\mathcal{H}_2}.$$

Since q is not separable, we must further have $\mathcal{H}_2$ is of dimension 1, thus $\mathcal{A} = \mathcal{L}(\mathcal{H}^q)$ (condition (3)). Otherwise q is again separable by considering

$$\mathcal{H}^q = \bigoplus_i \mathcal{H}_1 \otimes |\phi_i\rangle \langle\phi_i|,$$

where $\{\phi_i\}$ is a basis of $\mathcal{H}_2$.

Since Claim 1 is true, all terms are commuting in a regular way. This means all factors are either anti-commuting or commuting (condition (2)), and thus $\forall$ term p, $(p^q)^2$ commute with every factor, i.e. $(p^q)^2 \in \mathcal{Z}(\mathcal{A})$. Since we already argued that $\mathcal{Z}(\mathcal{A})$ must be trivial, we have $(p^q)^2 = c_{pq} I$ for some constant c_{pq} (condition (1)). ■

[BV03] showed that for qudit-factorized-CHP [12] , if all terms commute in a regular way, then one can transform the Hamiltonian into a qubit stabilizer Hamiltonian. In particular, they proved the following lemma.

Lemma 187 (Lemma 12 of [BV03]) *Let $\mathcal{H}_*^q$ be a Hilbert space, $G_1, ..., G_r \in \mathcal{L}(\mathcal{H}_*^q)$ be Hermitian operators such that*

$$G_a^2 = I, G_a G_b = \pm G_b G_a, \forall a, b \in \{1, ..., r\}$$

and such that the algebra generated by $G_1, ..., G_r$ coincides with $\mathcal{L}(\mathcal{H}_*^q)$. Then there exists an integer n, a tensor product structure $\mathcal{H}_*^q = (\mathbb{C}^2)^{\otimes n}$ and a unitary operator $U^q \in \mathcal{L}(\mathcal{H}_*^q)$ such that $U^q G_a U^{q\dagger}$ is a tensor of Pauli operators and the identity (up to sign) for all $a \in \{1, ..., r\}$. Here n may be equal to 0.

Finally, we are prepared to prove the main theorem in this section.

Theorem 188 *Consider a qudit-CLHP-2D-factorized Hamiltonian $H = \sum_p p$ on n-qudit space. If there are no separable qudits, then H is equivalent to qubit stabilizer Hamiltonian on $\mathcal{H}$.*

[12] Recall that CHP represents commuting Hamiltonian problem where the Hamiltonian might not be local.

Proof: To match the notations in Definition 178, we denote the Hilbert space of qudit q as $\mathcal{H}_*^q$, the n-qudit space as $\mathcal{H}_* := \otimes_q \mathcal{H}_*^q$. Consider a qudit-CLHP-2D-factorized $H = \sum_p p$. W.o.l.g assume that all p are non-zero. Note that since p is Hermitian, if $p^2 = 0$, then $p = 0$.

Consider arbitrary qudit q, use notations in Lemma 186 we know $c_{pq} \neq 0, \forall p, q$. For every p, define $\widetilde{p}$ be a normalized version of p, that is

$$\widetilde{p} = \otimes_q \widetilde{p}^q \text{ where } \widetilde{p}^q = p^q / c_{pq}.$$

For any qudit q, view $..., \widetilde{p}^q, ...$ as $..., G_a, ...$ By Lemma 186, one can check that $\{\widetilde{p}^q\}_p$ satisfies the condition of Lemma 187. Thus by choosing appropriate basis of each qudit, $\mathcal{H}_*^q$ will have a factorized structure as $(\mathbb{C}^2)^{\otimes m^q}$ for integer m^q, and $\{\widetilde{p}^q\}$ are tensor of Pauli operators up to sign. Thus p^q acts as a Pauli operator on this basis, up to phases. Thus H is equivalent to qubit stabilizer Hamiltonian H by definition.

■

The full version

Finally, we remove the constraints of no separable qudits in Theorem 188 and prove the following.

Theorem 189 *Any qudit-CLHP-2D-factorized Hamiltonian $H = \sum_p p$ is equivalent to a direct sum of qubit stabilizer Hamiltonian.*

Proof: Denote the space that H acting on as $\mathcal{H} = \bigotimes_q \mathcal{H}^q$. Recall that if a qudit q is separable with respect to decomposition $\mathcal{H}^q = \bigoplus_i \mathcal{H}_i^q$, for any chosen index i we can restrict all terms on this subspace, and get a new instance of qudit-CLHP-2D-factorized. If we repetitively perform this restriction whenever this is a separable qudit, after polytime we will reach the case with no separable qudits.

To prove theorem 189, it suffices to imagine the restricting process as a decision tree. Specifically, we write down a root node and define the space of the root node as $\mathcal{H}$, and repeat the following process: Transverse all the leaf nodes. Denote the leaf node considered currently as $*$, and its space as $\mathcal{H}_*$. If H restricting on $\mathcal{H}_*$ has separable qudits, choose an arbitrary such separable qudit. Denote this qudit as q and the corresponding decomposition as $\mathcal{H}^q = \bigoplus_i \mathcal{H}_i^q$. For every i, we build a child node w_i to $*$, and define the space of w_i as restricting $\mathcal{H}^q$ to $\mathcal{H}_i^q$ in $\mathcal{H}_*$. We repeat this process until for every leaf node, there are no separable qudits. In the

final tree, every leaf node $*$ corresponds to a simple subspace $\mathcal{H}_*$. By the definition of the tree, we know $\{\mathcal{H}_*\}_*$ are orthogonal to each other and $\mathcal{H} = \bigoplus_* \mathcal{H}_*$. By Theorem 188, $\mathcal{H}$ is equivalent to qubit stabilizer Hamiltonian on $\mathcal{H}_*$. Thus we prove that H is equivalent to a direct sum of qubit stabilizer Hamiltonian. ■

Corollary 190 *Qudit-CLHP-2D-factorized is in* ***NP****.*

Proof: Consider a qudit-CLHP-2D-factorized problem with Hamiltonian $H = \sum_p p$ and parameters a, b. By Theorem 189 we know there exists a $*$, where

$$\mathcal{H}_* = \bigotimes_q \mathcal{H}_*^q$$

is a simple subspace, such that there is a ground state lies in $\mathcal{H}_*$, and H is equivalent to qubit stabilizer Hamiltonian on $\mathcal{H}_*$. The **NP** prover is supposed to provide the subspace $\{\mathcal{H}_*^q\}_q$, and provide the qudit unitary U^q in Theorem 187 for each q. Using that information, the verifier firstly checks that all terms $\{p\}_p$ keep the subspaces $\{\mathcal{H}_*^q\}_q$ invariant. Then the verifier uses polynomial time to transform $\mathcal{H}_*^q$ to be tensor of qubit space, i.e.$(\mathbb{C}^2)^{\otimes m_q}$, and transform H on $\mathcal{H}_*$ to be a summation of Pauli operators up to phases, denoted as

$$H|_* = \sum_h a_h h.$$

Here h is a Pauli operator.

Then the verifier is going to verify $\lambda(H|_*) \leq a$. Since $\{a_h h\}_h$ are commuting, there is a ground state which is the common eigenstate of every h. Denote the corresponding eigenvalue as λ_h. The prover is supposed to provide such $\{\lambda_h\}_h$. The verifier verifies that

$$\sum_h a_h \lambda_h \leq a,$$

and verifies there is a state which is the common 1-eigenstate of commuting Pauli operators $\{h/\lambda_h\}$. The common 1-eigenstate verification can be done in polynomial time by standard stabilizer formalism. Note that although we describe the prover in an interactive way, they can in fact send all the witnesses at the same time. ■

Acknowledgement

We thank Thomas Vidick and Daniel Ranard for their helpful discussions. Jiaqing Jiang is supported by MURI Grant FA9550-18-1-0161 and the IQIM, an NSF Physics Frontiers Center (NSF Grant PHY-1125565).

6.6 Appendix:Relationship between general case and projection case

Lemma 191 *If k-qudit-CLHP-projection is in **NP**, then k-qudit-CLHP is in **NP**.*

Proof: Consider a k-qudit-CLHP(H, a, b), where

$$H = \sum_i h_i; a, b \in \mathbb{R}$$

and $b - a \geq 1/poly(n)$. Denote the ground energy $\lambda := \lambda(H)$. Since $\{h_i\}_i$ are commuting with each other, there exists a ground state $|\psi\rangle$ and $\{\lambda_i \in \mathbb{R}\}_i$ such that

$$h_i |\psi\rangle = \lambda_i |\psi\rangle, \forall i$$

and $\sum_i \lambda_i = \lambda$. Let Π_i be the projection onto the λ_i-eigenspace of h_i. Let

$$\hat{h}_i = I - \Pi_i,$$
$$\hat{H} = \sum_i \hat{h}_i.$$

Since $\{h_i\}_i$ are commuting, we know that $\{\hat{h}_i\}_i$ are also commuting.

The prover is supposed to list such $\{\lambda_i\}_i$, then the verifier can check $\sum_i \lambda_i < a$ and compute Π_i, $\hat{h}_i$ and $\hat{H}$. Then the prover is supposed to prove that qudit-CLHP-2D-projection$(\hat{H})$ is a Yes instance — that is, proving there exists $|\psi\rangle$ such that

$$\hat{h}_i |\psi\rangle = 0, \forall i.$$

Thus if k-qudit-CLHP-projection is in **NP**, then k-qudit-CLHP is in **NP**. ∎

Corollary 192 *If the qutrit-CLHP-2D-projection is in **NP**, then the qutrit-CLHP-2D is in **NP**.*

6.7 Appendix:Qudits on the vertexs or on the edges

In this manuscript, we consider commuting local Hamiltonian on a 2D square lattice, where qudits are on the vertices and Hermitian terms are on the plaquettes. There is another setting that put qudits on the edges, and Hermitian terms on "plaquettes" and "stars". Specifically, as shown in Figure 6.4 for each plaquette p, there is a Hermitian term B_p acting on the qudits on its edges, i.e. q_1, q_2, q_3, q_4. For each vertex v, consider the star consisting of v and edges adjacent to v, there is a Hermitian term A_v acting on qudits on its edges, i.e. q_3, q_4, q_5, q_6. The Hamiltonian is

$$H = \sum_p B_p + \sum_v A_v.$$

We abbreviate this setting as "qudits on 2D edges" and the setting in Sec. 6.2 as "qudits on 2D vertices". In the following we will show that the two settings are equivalent.

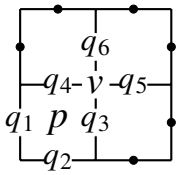


Figure 6.4: Qudits on edges to qudits on vertices

(1) "qudits on 2D edges" ⇒ "qudits on 2D vertices". Begin from "qudits on 2D edges", as shown in Figure 6.5, the qudits on the edges can in fact be viewed as qudits on the vertices of another 2D square lattice defined by the dashed lines. The terms B_p and A_s will correspond to plaquette terms in the dashed lattice. Thus our techniques directly apply to the setting for qudits on the edges.

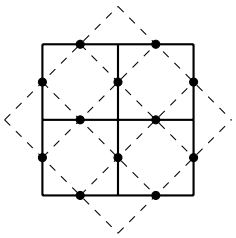


Figure 6.5: Qudits on 2D edges to qudits on 2D vertices

(1) "qudits on 2D vertices" ⇒ "qudits on 2D edges". Begin from "qudits on 2D vertices", as shown in Figure 6.6, the qudits on the vertices can in fact be viewed as qudits on the edges of another 2D square lattice defined by the dashed lines. The plaquette terms will correspond to plaquette and star terms in the dashed lattice.

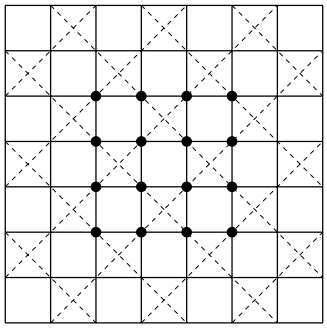


Figure 6.6: Qudits on 2D edges to qudits on 2D vertices

BIBLIOGRAPHY

[AAG22a] Anurag Anshu, Itai Arad, and David Gosset. "An area law for 2d frustration-free spin systems". In: *Proceedings of the 54th Annual ACM SIGACT Symposium on Theory of Computing*. 2022, pp. 12–18.

[AAG22b] Anurag Anshu, Itai Arad, and David Gosset. "An area law for 2d frustration-free spin systems". In: *Proceedings of the 54th Annual ACM SIGACT Symposium on Theory of Computing*. ACM, June 2022. DOI: 10.1145/3519935.3519962. URL: https://doi.org/10.1145%2F3519935.3519962.

[AAV13] Dorit Aharonov, Itai Arad, and Thomas Vidick. "Guest column: the quantum PCP conjecture". In: *Acm sigact news* 44.2 (2013), pp. 47–79.

[AB09] Sanjeev Arora and Boaz Barak. *Computational complexity: a modern approach*. Cambridge University Press, 2009.

[ABN23] Anurag Anshu, Nikolas P Breuckmann, and Chinmay Nirkhe. "NLTS Hamiltonians from good quantum codes". In: *Proceedings of the 55th Annual ACM Symposium on Theory of Computing*. 2023, pp. 1090–1096.

[AE11] Dorit Aharonov and Lior Eldar. "On the complexity of commuting local Hamiltonians, and tight conditions for topological order in such systems". In: *2011 IEEE 52nd Annual Symposium on Foundations of Computer Science*. IEEE. 2011, pp. 334–343.

[AE13] Dorit Aharonov and Lior Eldar. "The commuting local Hamiltonian on locally-expanding graphs is in NP". In: *arXiv preprint arXiv:1311.7378* (2013).

[AFH09] Robert Alicki, Mark Fannes, and Michal Horodecki. "On thermalization in Kitaev's 2D model". In: *Journal of Physics A: Mathematical and Theoretical* 42.6 (2009), p. 065303.

[AGM20] Anurag Anshu, David Gosset, and Karen Morenz. "Beyond product state approximations for a quantum analogue of max cut". In: *arXiv preprint arXiv:2003.14394* (2020).

[AKV18] Dorit Aharonov, Oded Kenneth, and Itamar Vigdorovich. "On the Complexity of Two Dimensional Commuting Local Hamiltonians". In: *13th Conference on the Theory of Quantum Computation, Communication and Cryptography (TQC 2018)*. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik. 2018.

[AL10] Itai Arad and Zeph Landau. "Quantum computation and the evaluation of tensor networks". In: *SIAM Journal on Computing* 39.7 (2010), pp. 3089–3121. DOI: https://doi.org/10.1137/080739379.

[Alh23] Álvaro M Alhambra. "Quantum many-body systems in thermal equilibrium". In: *PRX Quantum* 4.4 (2023), p. 040201.

[Ami+18] Mohammad H Amin et al. "Quantum boltzmann machine". In: *Physical Review X* 8.2 (2018), p. 021050.

[And] David F Anderson. "Lecture Notes on Stochastic Processes with Applications in Biology". In: ().

[Ara+17] Itai Arad et al. "Rigorous RG algorithms and area laws for low energy eigenstates in 1D". In: *Communications in Mathematical Physics* 356 (2017), pp. 65–105. DOI: https://doi.org/10.1007/s00220-017-2973-z.

[Bak+24] Ainesh Bakshi et al. "High-temperature Gibbs states are unentangled and efficiently preparable". In: *arXiv preprint arXiv:2403.16850* (2024).

[Ban23] Mari Carmen Banuls. "Tensor Network Algorithms: A Route Map". In: *Annual Review of Condensed Matter Physics* 14.1 (Mar. 2023), pp. 173–191. ISSN: 1947-5462. DOI: 10.1146/annurev-conmatphys-040721-022705. URL: http://dx.doi.org/10.1146/annurev-conmatphys-040721-022705.

[Bañ23] Mari Carmen Bañuls. "Tensor network algorithms: A route map". In: *Annual Review of Condensed Matter Physics* 14.1 (2023), pp. 173–191.

[Bar+23] Ivan Bardet et al. "Rapid thermalization of spin chain commuting Hamiltonians". In: *Physical Review Letters* 130.6 (2023), p. 060401.

[Bar14] Alexander Barvinok. "Computing the partition function for cliques in a graph". In: *arXiv preprint arXiv:1405.1974* (2014). DOI: DOI: 10.4086/toc.2015.v011a013.

[Bar16a] Alexander Barvinok. "Approximating permanents and hafnians". In: *arXiv preprint arXiv:1601.07518* (2016).

[Bar16b] Alexander Barvinok. "Computing the permanent of (some) complex matrices". In: *Foundations of Computational Mathematics* 16 (2016), pp. 329–342. DOI: https://doi.org/10.1007/s10208-014-9243-7.

[Bau+20] Bela Bauer et al. "Quantum algorithms for quantum chemistry and quantum materials science". In: *Chemical Reviews* 120.22 (2020), pp. 12685–12717.

[BBA24] Yimu Bao, Maxwell Block, and Ehud Altman. "Finite-time teleportation phase transition in random quantum circuits". In: *Physical Review Letters* 132.3 (2024), p. 030401. DOI: 10.1103/physrevlett.132.030401.

[BBT06] Sergey Bravyi, Arvid J Bessen, and Barbara M Terhal. "Merlin-Arthur games and stoquastic complexity". In: *arXiv preprint quant-ph/0611021* (2006).

[BCA20] Yimu Bao, Soonwon Choi, and Ehud Altman. "Theory of the phase transition in random unitary circuits with measurements". In: *Physical Review B* 101.10 (2020), p. 104301. DOI: https://doi.org/10.1103/PhysRevB.101.104301.

[BCK15] Dominic W Berry, Andrew M Childs, and Robin Kothari. "Hamiltonian simulation with nearly optimal dependence on all parameters". In: *2015 IEEE 56th annual symposium on foundations of computer science*. IEEE. 2015, pp. 792–809.

[BCL24] Thiago Bergamaschi, Chi-Fang Chen, and Yunchao Liu. "Quantum computational advantage with constant-temperature Gibbs sampling". In: *arXiv preprint arXiv:2404.14639* (2024).

[Ber+14] Dominic W Berry et al. "Exponential improvement in precision for simulating sparse Hamiltonians". In: *Proceedings of the forty-sixth annual ACM symposium on Theory of computing*. 2014, pp. 283–292.

[Ber+15] Dominic W Berry et al. "Simulating Hamiltonian dynamics with a truncated Taylor series". In: *Physical review letters* 114.9 (2015), p. 090502.

[BH18] Seymour Michael Blinder and James E House. *Mathematical physics in theoretical chemistry*. Elsevier, 2018.

[Bor+20] Christian Borgs et al. "Efficient sampling and counting algorithms for the Potts model on Z^d at all temperatures". In: *Proceedings of the 52nd Annual ACM SIGACT Symposium on Theory of Computing*. 2020, pp. 738–751.

[Bor+99] Christian Borgs et al. "Torpid mixing of some Monte Carlo Markov chain algorithms in statistical physics". In: *40th Annual Symposium on Foundations of Computer Science (Cat. No. 99CB37039)*. IEEE. 1999, pp. 218–229.

[Bra+08] Sergey Bravyi et al. "The complexity of stoquastic local Hamiltonian problems". In: *Quantum Information and Computation* 8.5 (2008), pp. 361–385.

[Bra+19] Fernando GSL Brandão et al. "Quantum SDP solvers: Large speed-ups, optimality, and applications to quantum learning". In: *46th International Colloquium on Automata, Languages, and Programming (ICALP 2019)*. Schloss-Dagstuhl-Leibniz Zentrum für Informatik. 2019.

[Bra+21] Sergey Bravyi et al. "On the complexity of quantum partition functions". In: *arXiv preprint arXiv:2110.15466* (2021).

[Bra+23a] Sergey Bravyi et al. “A rapidly mixing Markov chain from any gapped quantum many-body system”. In: *Quantum* 7 (2023), p. 1173.

[Bra+23b] Sergey Bravyi et al. “A rapidly mixing Markov chain from any gapped quantum many-body system”. In: *Quantum* 7 (2023), p. 1173.

[Bra14] Sergey Bravyi. “Monte Carlo simulation of stoquastic Hamiltonians”. In: *arXiv preprint arXiv:1402.2295* (2014).

[BŠ06] Harry Buhrman and Robert Špalek. “Quantum verification of matrix products”. In: *Proceedings of the seventeenth annual ACM-SIAM symposium on Discrete algorithm*. 2006, pp. 880–889.

[BSV14] Sergey Bravyi, Martin Suchara, and Alexander Vargo. “Efficient algorithms for maximum likelihood decoding in the surface code”. In: *Physical Review A* 90.3 (2014), p. 032326. DOI: https://doi.org/10.1103/PhysRevA.90.032326.

[BT10] Sergey Bravyi and Barbara Terhal. “Complexity of stoquastic frustration-free Hamiltonians”. In: *Siam journal on computing* 39.4 (2010), pp. 1462–1485.

[BV03] Sergey Bravyi and Mikhail Vyalyi. “Commutative version of the k-local Hamiltonian problem and common eigenspace problem”. In: *arXiv preprint quant-ph/0308021* (2003).

[Cao+19] Yudong Cao et al. “Quantum chemistry in the age of quantum computing”. In: *Chemical reviews* 119.19 (2019), pp. 10856–10915.

[Car+19] Juan Carrasquilla et al. “Reconstructing quantum states with generative models”. In: *Nature Machine Intelligence* 1.3 (2019), pp. 155–161.

[CCS87] JT Chayes, L Chayes, and Roberto Henrique Schonmann. “Exponential decay of connectivities in the two-dimensional Ising model”. In: *Journal of Statistical Physics* 49 (1987), pp. 433–445.

[Ces+96] Filippo Cesi et al. “On the two-dimensional stochastic Ising model in the phase coexistence region near the critical point”. In: *Journal of statistical physics* 85 (1996), pp. 55–102.

[Ces01] Filippo Cesi. “Quasi-factorization of the entropy and logarithmic Sobolev inequalities for Gibbs random fields”. In: *Probability Theory and Related Fields* 120 (2001), pp. 569–584.

[Che+23] Chi-Fang Chen et al. “Quantum thermal state preparation”. In: *arXiv preprint arXiv:2303.18224* (2023).

[Che+25] Jielun Chen et al. “Sign Problem in Tensor-Network Contraction”. In: *PRX Quantum* 6 (2025), p. 010312. DOI: 10.1103/prxquantum.6.010312. eprint: arXiv:2404.19023.

[Chi+20] Nai-Hui Chia et al. “Quantum-inspired algorithms for solving low-rank linear equation systems with logarithmic dependence on the dimension”. In: *31st International Symposium on Algorithms and Computation (ISAAC 2020)*. Schloss Dagstuhl-Leibniz-Zentrum für Informatik. 2020.

[Chi+22] Nai-Hui Chia et al. “Sampling-based sublinear low-rank matrix arithmetic framework for dequantizing quantum machine learning”. In: *Journal of the ACM* 69.5 (2022), pp. 1–72.

[CKG23] Chi-Fang Chen, Michael J Kastoryano, and András Gilyén. “An efficient and exact noncommutative quantum Gibbs sampler”. In: *arXiv preprint arXiv:2311.09207* (2023).

[CL16] Andrew M Childs and Tongyang Li. “Efficient simulation of sparse Markovian quantum dynamics”. In: *arXiv preprint arXiv:1611.05543* (2016).

[Con+23] Mirko Consiglio et al. “Variational Gibbs State Preparation on NISQ devices”. In: *arXiv preprint arXiv:2303.11276* (2023).

[Cor16] Philippe Corboz. “Variational optimization with infinite projected entangled-pair states”. In: *Physical Review B* 94.3 (2016), p. 035133.

[CRF20] Ángela Capel, Cambyse Rouzé, and Daniel Stilck França. “The modified logarithmic Sobolev inequality for quantum spin systems: classical and commuting nearest neighbour interactions”. In: *arXiv preprint arXiv:2009.11817* (2020).

[CS16] Anirban Narayan Chowdhury and Rolando D Somma. “Quantum algorithms for Gibbs sampling and hitting-time estimation”. In: *arXiv preprint arXiv:1603.02940* (2016).

[CT17] Giuseppe Carleo and Matthias Troyer. “Solving the quantum many-body problem with artificial neural networks”. In: *Science* 355.6325 (2017), pp. 602–606.

[CW16] Richard Cleve and Chunhao Wang. “Efficient quantum algorithms for simulating Lindblad evolution”. In: *arXiv preprint arXiv:1612.09512* (2016).

[Dav76] Edward Brian Davies. “Quantum theory of open systems”. In: *(No Title)* (1976).

[Dav79] E Brian Davies. “Generators of dynamical semigroups”. In: *Journal of Functional Analysis* 34.3 (1979), pp. 421–432.

[Din+24] Zhiyan Ding et al. “Polynomial-Time Preparation of Low-Temperature Gibbs States for 2D Toric Code”. In: 2024. URL: https://api.semanticscholar.org/CorpusID:273025835.

[DLL24] Zhiyan Ding, Bowen Li, and Lin Lin. "Efficient quantum Gibbs samplers with Kubo–Martin–Schwinger detailed balance condition". In: *arXiv preprint arXiv:2404.05998* (2024).

[Dye+04] Martin Dyer et al. "Mixing in time and space for lattice spin systems: A combinatorial view". In: *Random Structures & Algorithms* 24.4 (2004), pp. 461–479.

[EM18] Lior Eldar and Saeed Mehraban. "Approximating the permanent of a random matrix with vanishing mean". In: *2018 IEEE 59th Annual Symposium on Foundations of Computer Science (FOCS)*. IEEE. 2018, pp. 23–34.

[EV15] Glen Evenbly and Guifre Vidal. "Tensor network renormalization". In: *Physical review letters* 115.18 (2015), p. 180405. DOI: https://doi.org/10.1103/PhysRevLett.115.180405.

[Fey18] Richard P Feynman. "Simulating physics with computers". In: *Feynman and computation*. cRc Press, 2018, pp. 133–153.

[FGW23] Weiming Feng, Heng Guo, and Jiaheng Wang. "Swendsen-Wang dynamics for the ferromagnetic Ising model with external fields". In: *Information and Computation* 294 (2023), p. 105066.

[Fis87] Charlotte Froese Fischer. "General hartree-fock program". In: *Computer physics communications* 43.3 (1987), pp. 355–365.

[FP14] Andrew J Ferris and David Poulin. "Tensor networks and quantum error correction". In: *Physical review letters* 113.3 (2014), p. 030501. DOI: https://doi.org/10.1103/PhysRevLett.113.030501.

[Fre79] Rūsiņš Freivalds. "Fast probabilistic algorithms". In: *International Symposium on Mathematical Foundations of Computer Science*. Springer. 1979, pp. 57–69.

[FS97] Gerald B Folland and Alladi Sitaram. "The uncertainty principle: a mathematical survey". In: *Journal of Fourier analysis and applications* 3 (1997), pp. 207–238.

[GC24] Johnnie Gray and Garnet Kin-Lic Chan. "Hyperoptimized approximate contraction of tensor networks with arbitrary geometry". In: *Physical Review X* 14.1 (2024), p. 011009. DOI: https://doi.org/10.1103/PhysRevX.14.011009.

[GD17] Xun Gao and Lu-Ming Duan. "Efficient representation of quantum many-body states with deep neural networks". In: *Nature communications* 8.1 (2017), p. 662.

[Gha+15] Sevag Gharibian et al. "Quantum hamiltonian complexity". In: *Foundations and Trends® in Theoretical Computer Science* 10.3 (2015), pp. 159–282.

[Gil+19] András Gilyén et al. “Quantum singular value transformation and beyond: exponential improvements for quantum matrix arithmetics”. In: *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing*. 2019, pp. 193–204.

[Gil+24a] András Gilyén et al. “Quantum generalizations of Glauber and Metropolis dynamics”. In: *arXiv preprint arXiv:2405.20322* (2024).

[Gil+24b] András Gilyén et al. “Quantum generalizations of Glauber and Metropolis dynamics”. In: *arXiv preprint arXiv:2405.20322* (2024).

[GKW16] James Gubernatis, Naoki Kawashima, and Philipp Werner. *Quantum Monte Carlo Methods*. Cambridge University Press, 2016.

[GL16] Reza Gheissari and Eyal Lubetzky. “Mixing times of critical 2D Potts models”. In: *arXiv preprint arXiv:1607.02182* (2016).

[GL22] Sevag Gharibian and François Le Gall. “Dequantizing the quantum singular value transformation: hardness and applications to quantum chemistry and the quantum PCP conjecture”. In: *Proceedings of the 54th Annual ACM SIGACT Symposium on Theory of Computing*. 2022, pp. 19–32.

[GMV17] David Gosset, Jenish C Mehta, and Thomas Vidick. “QCMA hardness of ground space connectivity for commuting Hamiltonians”. In: *Quantum* 1 (2017), p. 16.

[GOL24] Andi Gu, Salvatore FE Oliviero, and Lorenzo Leone. “Doped stabilizer states in many-body physics and where to find them”. In: *arXiv preprint arXiv:2403.14912* (2024).

[Got97] Daniel Gottesman. *Stabilizer codes and quantum error correction*. California Institute of Technology, 1997.

[Guo+19] Chu Guo et al. “General-purpose quantum circuit simulator with projected entangled-pair states and the quantum supremacy frontier”. In: *Physical review letters* 123.19 (2019), p. 190501. DOI: https://doi.org/10.1103/PhysRevLett.123.190501.

[GZ03] Alice Guionnet and Bogusław Zegarlinksi. “Lectures on logarithmic Sobolev inequalities”. In: *Séminaire de probabilités XXXVI* (2003), pp. 1–134.

[Haf+20] Jonas Haferkamp et al. “Contracting projected entangled pair states is average-case hard”. In: *Physical Review Research* 2.1 (2020), p. 013010. DOI: https://doi.org/10.1103/PhysRevResearch.2.013010.

[Han+20] Dominik Hangleiter et al. “Easing the Monte Carlo sign problem”. In: *Science advances* 6.33 (2020), eabb8341.

[Has07] Matthew B Hastings. "An area law for one-dimensional quantum systems". In: *Journal of statistical mechanics: theory and experiment* 2007.08 (2007), P08024. DOI: 10.1088/1742-5468/2007/08/P08024.

[Has11] Matthew B Hastings. "Topological order at nonzero temperature". In: *Physical review letters* 107.21 (2011), p. 210501.

[Has12] Matthew B Hastings. "Matrix product operators and central elements: Classical description of a quantum state". In: *Geometry & Topology Monographs* 18.115-160 (2012), p. 276.

[Has13] Matthew B. Hastings. "Trivial Low Energy States for Commuting Hamiltonians, and the Quantum PCP Conjecture". In: *Quantum Info. Comput.* 13.5–6 (May 2013), pp. 393–429. ISSN: 1533-7146.

[Hay+16] Patrick Hayden et al. "Holographic duality from random tensor networks". In: *Journal of High Energy Physics* 2016.11 (2016), pp. 1–56. DOI: https://doi.org/10.1007/JHEP11(2016)009.

[Hol85] Richard Holley. "Rapid convergence to equilibrium in one dimensional stochastic Ising models". In: *The Annals of Probability* (1985), pp. 72–89.

[HPR19] Tyler Helmuth, Will Perkins, and Guus Regts. "Algorithmic pirogovsinai theory". In: *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing*. 2019, pp. 1009–1020.

[HS83] Geoffrey E Hinton and Terrence J Sejnowski. "Optimal perceptual inference". In: *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*. Vol. 448. Citeseer. 1983, pp. 448–453.

[HS89] Richard A Holley and Daniel W Stroock. "Uniform and L 2 convergence in one dimensional stochastic Ising models". In: *Communications in mathematical physics* 123.1 (1989), pp. 85–93.

[HST01] Heikki Haario, Eero Saksman, and Johanna Tamminen. "An adaptive Metropolis algorithm". In: (2001).

[Hua+20] Cupjin Huang et al. "Classical simulation of quantum supremacy circuits". In: *arXiv preprint arXiv:2005.06787* (2020).

[IJ23] Sandy Irani and Jiaqing Jiang. "Commuting Local Hamiltonian Problem on 2D beyond qubits". In: *arXiv preprint arXiv:2309.04910* (2023).

[Ioa+20] Marios Ioannou et al. "Termwise versus globally stoquastic local Hamiltonians: questions of complexity and sign-curing". In: *arXiv preprint arXiv:2007.11964* (2020).

[JI24] Jiaqing Jiang and Sandy Irani. "Quantum Metropolis Sampling via Weak Measurement". In: *arXiv preprint arXiv:2406.16023* (2024).

[Jia+24] Jiaqing Jiang et al. “Positive bias makes tensor-network contraction tractable”. In: *arXiv preprint arXiv:2410.05414* (2024).

[Jia25] Jiaqing Jiang. “Local Hamiltonian problem with succinct ground state is MA-complete”. In: *PRX Quantum* 6.2 (2025), p. 020312.

[Kau49] Bruria Kaufman. “Crystal statistics. II. Partition function evaluated by spinor analysis”. In: *Physical Review* 76.8 (1949), p. 1232. DOI: https://doi.org/10.1103/PhysRev.76.1232.

[KB16] Michael J Kastoryano and Fernando GSL Brandao. “Quantum Gibbs samplers: The commuting case”. In: *Communications in Mathematical Physics* 344 (2016), pp. 915–957.

[Kit03a] A Yu Kitaev. “Fault-tolerant quantum computation by anyons”. In: *Annals of physics* 303.1 (2003), pp. 2–30.

[Kit03b] A.Yu. Kitaev. “Fault-tolerant quantum computation by anyons”. In: *Annals of Physics* 303.1 (Jan. 2003), pp. 2–30. DOI: 10.1016/s0003-4916(02)00018-0. URL: https://doi.org/10.1016%2Fs0003-4916%2802%2900018-0.

[KKR06] Julia Kempe, Alexei Kitaev, and Oded Regev. “The complexity of the local Hamiltonian problem”. In: *Siam journal on computing* 35.5 (2006), pp. 1070–1097.

[Koc+24] Jan Kochanowski et al. “Rapid thermalization of dissipative many-body dynamics of commuting Hamiltonians”. In: *arXiv preprint arXiv:2404.16780* (2024).

[KSV02] Alexei Yu Kitaev, Alexander Shen, and Mikhail N Vyalyi. *Classical and quantum computation*. 47. American Mathematical Soc., 2002.

[KT19] Joel Klassen and Barbara M Terhal. “Two-local qubit Hamiltonians: when are they stoquastic?” In: *Quantum* 3 (2019), p. 139.

[KZW06] SC Kou, Qing Zhou, and Wing Hung Wong. “Equi-energy sampler with applications in statistical inference and statistical mechanics”. In: (2006).

[LC19] Guang Hao Low and Isaac L Chuang. “Hamiltonian simulation by qubitization”. In: *Quantum* 3 (2019), p. 163.

[LC21] Ryan Levy and Bryan K Clark. “Entanglement entropy transitions with random tensor networks”. In: *arXiv preprint arXiv:2108.02225* (2021).

[Lee+22] Chee Kong Lee et al. “Variational Quantum Simulations of Finite-Temperature Dynamical Properties via Thermofield Dynamics”. In: *arXiv preprint arXiv:2206.05571* (2022).

[Lee+23] Seunghoon Lee et al. “Evaluating the evidence for exponential quantum advantage in ground-state quantum chemistry”. In: *Nature Communications* 14.1 (2023), p. 1952.

[Liu21] Yupan Liu. "StoqMA Meets Distribution Testing". In: *16th Conference on the Theory of Quantum Computation, Communication and Cryptography*. 2021.

[LP13] Olivier Landon-Cardinal and David Poulin. "Local topological order inhibits thermal stability in 2D". In: *Physical review letters* 110.9 (2013), p. 090502.

[LVV15] Zeph Landau, Umesh Vazirani, and Thomas Vidick. "A polynomial time algorithm for the ground state of one-dimensional gapped local Hamiltonians". In: *Nature Physics* 11.7 (2015), pp. 566–569.

[LW22] Xiantao Li and Chunhao Wang. "Simulating Markovian open quantum systems using higher-order series expansion". In: *arXiv preprint arXiv:2212.02051* (2022).

[Maj66] Chanchal K Majumdar. "Analytic properties of the onsager solution of the Ising model". In: *Physical Review* 145.1 (1966), p. 158.

[McA+20] Sam McArdle et al. "Quantum computational chemistry". In: *Reviews of Modern Physics* 92.1 (2020), p. 015003.

[Met+20] Mekena Metcalf et al. "Engineered thermalization and cooling of quantum many-body systems". In: *Physical Review Research* 2.2 (2020), p. 023214.

[Met+53] Nicholas Metropolis et al. "Equation of state calculations by fast computing machines". In: *The journal of chemical physics* 21.6 (1953), pp. 1087–1092.

[MH21] Ryan L Mann and Tyler Helmuth. "Efficient algorithms for approximating quantum partition functions". In: *Journal of Mathematical Physics* 62.2 (2021). DOI: https://doi.org/10.1063/5.0013689.

[MLH19] Milad Marvian, Daniel A Lidar, and Itay Hen. "On the computational complexity of curing non-stoquastic Hamiltonians". In: *Nature communications* 10.1 (2019), p. 1571.

[MO94] Fabio Martinelli and Enzo Olivieri. "Approach to equilibrium of Glauber dynamics in the one phase region: II. The general case". In: *Communications in Mathematical Physics* 161.3 (1994), pp. 487–514.

[Mot+20] Mario Motta et al. "Determining eigenstates and thermal states on a quantum computer using quantum imaginary time evolution". In: *Nature Physics* 16.2 (2020), pp. 205–210.

[Mov18] Ramis Movassagh. "Efficient unitary paths and quantum computational supremacy: A proof of average-case hardness of Random Circuit Sampling". In: *arXiv preprint arXiv:1810.04681* (2018).

[MS08] Igor L Markov and Yaoyun Shi. "Simulating quantum computation by contracting tensor networks". In: *SIAM Journal on Computing* 38.3 (2008), pp. 963–981. DOI: https://doi.org/10.1137/050644756.

[Mül+15] Markus P Müller et al. "Thermalization and canonical typicality in translation-invariant quantum lattice systems". In: *Communications in Mathematical Physics* 340 (2015), pp. 499–561.

[MVC07] Valentin Murg, Frank Verstraete, and J Ignacio Cirac. "Variational study of hard-core bosons in a two-dimensional optical lattice using projected entangled pair states". In: *Physical Review A* 75.3 (2007), p. 033605. DOI: https://doi.org/10.1103/PhysRevA.75.033605.

[MW05] Chris Marriott and John Watrous. "Quantum arthur–merlin games". In: *computational complexity* 14.2 (2005), pp. 122–152.

[MZ18] Mario Motta and Shiwei Zhang. "Ab initio computations of molecular systems by the auxiliary-field quantum Monte Carlo method". In: *Wiley Interdisciplinary Reviews: Computational Molecular Science* 8.5 (2018), e1364.

[NC10] Michael A Nielsen and Isaac L Chuang. *Quantum computation and quantum information*. Cambridge university press, 2010.

[Nor98] James R Norris. *Markov chains*. 2. Cambridge university press, 1998.

[Ons44] Lars Onsager. "Crystal statistics. I. A two-dimensional model with an order-disorder transition". In: *Physical Review* 65.3-4 (1944), p. 117. DOI: https://doi.org/10.1103/PhysRev.65.117.

[Orú19] Román Orús. "Tensor networks for complex quantum systems". In: *Nature Reviews Physics* 1.9 (2019), pp. 538–550. DOI: https://doi.org/10.1038/s42254-019-0086-7.

[Ped+17] Edwin Pednault et al. "Breaking the 49-qubit barrier in the simulation of quantum circuits". In: *arXiv preprint arXiv:1710.05867* 15 (2017).

[PR17] Viresh Patel and Guus Regts. "Deterministic polynomial-time approximation algorithms for partition functions and graph polynomials". In: *SIAM Journal on Computing* 46.6 (2017), pp. 1893–1919.

[PW09] David Poulin and Pawel Wocjan. "Sampling from the thermal quantum Gibbs state and evaluating partition functions with a quantum computer". In: *arXiv preprint arXiv:0905.2199* (2009).

[Qin+22] Mingpu Qin et al. "The Hubbard model: A computational perspective". In: *Annual Review of Condensed Matter Physics* 13 (2022), pp. 275–302.

[RFA24] Cambyse Rouzé, Daniel Stilck França, and Álvaro M Alhambra. "Efficient thermalization and universal quantum computing with quantum Gibbs samplers". In: *arXiv preprint arXiv:2403.12691* (2024).

[RGE12] Arnau Riera, Christian Gogolin, and Jens Eisert. "Thermalization in nature and on a quantum computer". In: *Physical review letters* 108.8 (2012), p. 080402.

[RW24] Joel Rajakumar and James D Watson. "Gibbs Sampling gives Quantum Advantage at Constant Temperatures with $O(1)$-Local Hamiltonians". In: *arXiv preprint arXiv:2408.01516* (2024).

[RWW23] Patrick Rall, Chunhao Wang, and Pawel Wocjan. "Thermal state preparation via rounding promises". In: *Quantum* 7 (2023), p. 1132.

[SBE17] Martin Schwarz, Olivier Buerschaper, and Jens Eisert. "Approximating local observables on projected entangled pair states". In: *Physical Review A* 95.6 (2017), p. 060102. DOI: https://doi.org/10.1103/PhysRevA.95.060102.

[Sch+07] Norbert Schuch et al. "Computational complexity of projected entangled pair states". In: *Physical review letters* 98.14 (2007), p. 140506. DOI: https://doi.org/10.1103/PhysRevLett.98.140506.

[Sch05] Ulrich Schollwöck. "The density-matrix renormalization group". In: *Reviews of modern physics* 77.1 (2005), p. 259.

[Sch11] Norbert Schuch. "Complexity of commuting Hamiltonians on a square lattice of qubits". In: *Quantum Information & Computation* 11.11-12 (2011), pp. 901–912.

[Sch87] Roberto H Schonmann. "Second order large deviation estimates for ferromagnetic systems in the phase coexistence region". In: *Communications in mathematical physics* 112.3 (1987), pp. 409–422.

[SN16] Alireza Shabani and Hartmut Neven. "Artificial quantum thermal bath: Engineering temperature for a many-body quantum system". In: *Physical review A* 94.5 (2016), p. 052301.

[SRN19] Brian Skinner, Jonathan Ruhman, and Adam Nahum. "Measurement-induced phase transitions in the dynamics of entanglement". In: *Physical Review X* 9.3 (2019), p. 031009. DOI: https://doi.org/10.1103/PhysRevX.9.031009.

[SSC22] Or Sharir, Amnon Shashua, and Giuseppe Carleo. "Neural tensor contractions and the expressive power of deep neural quantum states". In: *Physical Review B* 106.20 (2022), p. 205136.

[Sto83] Larry Stockmeyer. "The complexity of approximate counting". In: *Proceedings of the fifteenth annual ACM symposium on Theory of computing*. 1983, pp. 118–126.

[SW87] Robert H Swendsen and Jian-Sheng Wang. "Nonuniversal critical dynamics in Monte Carlo simulations". In: *Physical review letters* 58.2 (1987), p. 86.

[SZ92a] Daniel W Stroock and Boguslaw Zegarlinski. "The equivalence of the logarithmic Sobolev inequality and the Dobrushin-Shlosman mixing condition". In: *Communications in mathematical physics* 144 (1992), pp. 303–323.

[SZ92b] Daniel W Stroock and Boguslaw Zegarlinski. "The logarithmic Sobolev inequality for discrete spin systems on a lattice". In: *Communications in Mathematical Physics* 149 (1992), pp. 175–193.

[Tag23] Hamed Taghavian. *A fast algorithm for computing Bell polynomials based on index break-downs using prime factorization*. 2023. arXiv: 2004.09283 [math.CA]. URL: https://arxiv.org/abs/2004.09283.

[Tak+03] Masamichi Takesaki et al. *Theory of operator algebras II*. Vol. 125. Springer, 2003.

[Tan19] Ewin Tang. "A quantum-inspired classical algorithm for recommendation systems". In: *Proceedings of the 51st annual ACM SIGACT symposium on theory of computing*. 2019, pp. 217–228.

[TD00] Barbara M Terhal and David P DiVincenzo. "Problem of equilibration and the computation of correlation functions on a quantum computer". In: *Physical Review A* 61.2 (2000), p. 022301.

[Tem+11] Kristan Temme et al. "Quantum metropolis sampling". In: *Nature* 471.7336 (2011), pp. 87–90.

[Ten+95] DFB Ten Haaf et al. "Proof for an upper bound in fixed-node Monte Carlo for lattice fermions". In: *Physical Review B* 51.19 (1995), p. 13039.

[Til+22] Jules Tilly et al. "The variational quantum eigensolver: a review of methods and best practices". In: *Physics Reports* 986 (2022), pp. 1–128.

[Ull12] Mario Ullrich. "Rapid mixing of Swendsen-Wang dynamics in two dimensions". In: *arXiv preprint arXiv:1212.4908* (2012).

[Van+16] Laurens Vanderstraeten et al. "Gradient methods for variational optimization of projected entangled-pair states". In: *Physical Review B* 94.15 (2016), p. 155123.

[Van+17] Joran Van Apeldoorn et al. "Quantum SDP-solvers: Better upper and lower bounds". In: *2017 IEEE 58th Annual Symposium on Foundations of Computer Science (FOCS)*. IEEE. 2017, pp. 403–414.

[VC21] Patrick CG Vlaar and Philippe Corboz. "Simulation of three-dimensional quantum systems with projected entangled-pair states". In: *Physical Review B* 103.20 (2021), p. 205137. DOI: https://doi.org/10.1103/PhysRevB.103.205137.

[Vid03] Guifré Vidal. "Efficient classical simulation of slightly entangled quantum computations". In: *Physical review letters* 91.14 (2003), p. 147902.

[VMC08] Frank Verstraete, Valentin Murg, and J Ignacio Cirac. "Matrix product states, projected entangled pair states, and variational renormalization group methods for quantum spin systems". In: *Advances in physics* 57.2 (2008), pp. 143–224.

[WFC23] Jordi Weggemans, Marten Folkertsma, and Chris Cade. "Guidable Local Hamiltonian Problems with Implications to Heuristic Ansatze State Preparation and the Quantum PCP Conjecture". In: *arXiv preprint arXiv:2302.11578* (2023).

[Whi92] Steven R White. "Density matrix formulation for quantum renormalization groups". In: *Physical review letters* 69.19 (1992), p. 2863.

[Whi93] Steven R White. "Density-matrix algorithms for quantum renormalization groups". In: *Physical review b* 48.14 (1993), p. 10345. DOI: https://doi.org/10.1103/PhysRevB.48.10345.

[WJB03] Pawel Wocjan, Dominik Janzing, and Thomas Beth. "Two QCMA-complete problems". In: *Quantum Information & Computation* 3.6 (2003), pp. 635–643.

[WLW21] Youle Wang, Guangxi Li, and Xin Wang. "Variational quantum Gibbs state preparation with a truncated Taylor series". In: *Physical Review Applied* 16.5 (2021), p. 054035.

[Wol12] Michael M Wolf. "Quantum channels and operations-guided tour". In: (2012).

[WT23] Pawel Wocjan and Kristan Temme. "Szegedy walk unitaries for quantum maps". In: *Communications in Mathematical Physics* 402.3 (2023), pp. 3201–3231.

[Yan+22] Zhi-Cheng Yang et al. "Entanglement phase transitions in random stabilizer tensor networks". In: *Physical Review B* 105.10 (2022), p. 104306. DOI: https://doi.org/10.1103/PhysRevB.105.104306.

[ZBC23] Daniel Zhang, Jan Lukas Bosse, and Toby Cubitt. "Dissipative quantum Gibbs sampling". In: *arXiv preprint arXiv:2304.04526* (2023).

[Zeg90] Bogusław Zegarlinski. "Log-Sobolev inequalities for infinite one-dimensional lattice systems". In: *Commun. Math. Phys* 133.1 (1990), pp. 147–162.

[Zha+10] Hui-Hai Zhao et al. “Renormalization of tensor-network states”. In: *Physical Review B* 81.17 (2010), p. 174411. DOI: https://doi.org/10.1103/PhysRevB.81.174411.

[ZK11] Konstantin M Zuev and Lambros S Katafygiotis. “Modified Metropolis–Hastings algorithm with delayed rejection”. In: *Probabilistic Engineering Mechanics* 26.3 (2011), pp. 405–412.