On Near-Term Quantum Computation

Theoretical Aspects of Variational Quantum Algorithms and Quantum Computational Supremacy

by

John Christopher Napp

B.S., California Institute of Technology (2014)

Submitted to the Department of Physics
in partial fulfillment of the requirements for the degree of

Doctor of Philosophy in Physics

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

June 2021

Author .....
Department of Physics
May 21, 2021

Certified by .....
Aram W. Harrow
Associate Professor of Physics
Thesis Supervisor

Accepted by .....
Deepto Chakrabarty
Associate Department Head of Physics

On Near-Term Quantum Computation

Theoretical Aspects of Variational Quantum Algorithms and Quantum Computational Supremacy

by

John Christopher Napp

Submitted to the Department of Physics
on May 21, 2021, in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy in Physics

Abstract

In recent years, programmable quantum devices have reached sizes and complexities which put them outside the regime of simulation on modern supercomputers. However, since their computational power is not well understood, it's not obvious what to do with them! Of course, there are several ideas, and this thesis contributes to the theory underpinning some of these ideas. It has two parts, corresponding to two of the most natural directions to pursue in searching for applications of near-term quantum computers. The first part is concerned with obtaining a deeper understanding of heuristic, hybrid quantum-classical algorithms which are potentially implementable on near-term devices and are aimed at attaining quantum speedups for practical problems, but lack a strong theoretical foundation and provable guarantees on their performance. More precisely, we obtain new theoretical results on the convergence rates of *variational quantum algorithms*, and prove that certain optimization strategies in such algorithms can, in some settings, lead to substantially better performance than the originally proposed, simpler, and potentially easier-to-implement approach. The second part is concerned with better understanding the capabilities of near-term quantum computers for demonstrating evidence of *quantum computational supremacy* in the complexity-theoretic sense of violating the Extended Church-Turing Thesis: a superpolynomial quantum speedup for a well-defined computational problem, possibly of no practical use, over all classical algorithms. More precisely, we study the computational complexity of classically simulating random 2D quantum circuits. While the classical hardness of simulating random circuits forms the basis of one of the leading quantum supremacy proposals, we challenge some of the intuition and evidence underlying this belief by developing new classical simulation algorithms which are efficient (polynomial-time) for 2D random circuits of sufficiently low constant depth; interestingly, these algorithms appear to experience computational phase transitions into an inefficient, exponential-time regime when the depth or local Hilbert space dimension surpasses some critical value.

Thesis Supervisor: Aram W. Harrow
Title: Associate Professor of Physics

Acknowledgments

I'm grateful to my friends, family, teachers, mentors, and colleagues for making this thesis possible. At the top of my list of academic influences is my advisor Aram Harrow. Aram was invaluable in teaching me how to do research and think about quantum information, and is a seemly endless source of insights into quantum information and beyond; I left virtually every meeting with some nugget of wisdom or an interesting idea to explore. I'm grateful for his generosity of time and his encouragement to explore whatever research directions I most enjoyed. I've also been extremely fortunate to have Aram as well as Fernando Brandão, Alex Dalzell, and Rolando La Placa as coauthors during my time at MIT, as well as Matilde Marcolli and John Preskill during my prior undergraduate years at Caltech. All were great collaborators, and I became not only a stronger researcher but also a wider thinker in general by learning from the unique perspectives and approaches that each brought.

Of course, I've also benefited immensely from many others in the quantum information community (and adjacent), junior and senior, at MIT and elsewhere, from those with whom I've enjoyed fruitful research discussions, to fellow grad students with whom it's been a pleasure to share in the ups and downs of grad school. A partial list includes Nilin Abrahamsen, Eric Anschuetz, Srinivasan Arunachalam, Carina Belvin, Adam Bene Watts, Sergey Bravyi, Daniel Grier, Dhiraj Holden, Dax Koh, Linghang Kong, Tongyang Li, Yupan Liu, Zi-Wen Liu, Saeed Mehraban, Beatrice Nash, Anand Natarajan, Elina Sendonaris, Oles Shtanko, Mehdi Soleimanifar, Ryuji Takagi, Annie Wei, John Wright, Xiaodi Wu, Ted Yoder, Nicole Yunger Halpern, Elton Zhu, and my academic advisor Barton Zwiebach. Thank you to Aram, Peter Shor, and Vladan Vuletić for serving on my thesis committee.

I'm also grateful for the travel opportunities I've had throughout my PhD. For this, I'd like to thank Marco Cerezo, Lukasz Cincio, Andrew Sornborger, and especially Patrick Coles for hosting me during a visit to Los Alamos National Lab, where we had many illuminating discussions on variational quantum algorithms. I was also fortunate to attend the 2018 Boulder School for Condensed Matter Physics on Quantum Information, where I had the opportunity to meet many outstanding researchers in the field. Thanks also to IBM Research for hosting me for multiple four-week micro-internships.

Outside of quantum information, I'm incredibly grateful to my friends and family for their support along the way. A full accounting isn't feasible here, but I'd especially like to thank my parents, Tom, Iulia, Nicoleta, and of course my animal companions: Dolce, Jessie, and Luna.

Contents

1 Introduction **15**
- 1.1 Organization and bibliographical information . . . 19
- 1.2 Variational quantum algorithms (VQAs) . . . 19
- 1.3 Random quantum circuits and quantum computational supremacy . . 22
- 1.4 Overview of results . . . 26
  - 1.4.1 Variational algorithms . . . 27
  - 1.4.2 Random quantum circuits and their classical simulation . . . 30

2 Analytical Gradient Measurements Can Accelerate VQAs, I: Technical Exposition **37**
- 2.1 Introduction . . . 37
- 2.2 Prior work . . . 40
- 2.3 Black box model . . . 41
- 2.4 Lower bounds . . . 42
- 2.5 Upper bounds . . . 43
- 2.6 Conclusion . . . 45

3 Analytical Gradient Measurements Can Accelerate VQAs, II: Details and Derivations **47**
- 3.1 Technical preliminaries . . . 47
  - 3.1.1 Conventions, assumptions, and notation . . . 47
  - 3.1.2 Requisite results about stochastic convex optimization . . . 48
    - 3.1.2.1 Upper bounds for stochastic first-order optimization 49
    - 3.1.2.2 Upper bounds for stochastic zeroth-order (derivative-free) optimization . . . 50
- 3.2 Black-box formulation . . . 50
  - 3.2.1 Zeroth-order sampling . . . 51
  - 3.2.2 Review: analytic gradient measurements . . . 52
  - 3.2.3 First-order sampling . . . 54
  - 3.2.4 Higher-order sampling . . . 55
  - 3.2.5 Query complexity in the black-box formalism . . . 56
- 3.3 General upper bounds for variational algorithms in a convex region . 57
  - 3.3.1 Gradient estimators from oracle queries . . . 58
  - 3.3.2 Upper bounds . . . 59
  - 3.3.3 When is SMD superior to SGD? . . . 60

3.4 Oracle separation between zeroth-order and first-order optimization strategies for variational algorithms . . . 61
3.4.1 Defining $\mathcal{H}_n^\epsilon$ . . . 62
3.4.2 Proof of Theorem 9: zeroth-order lower bound for $\mathcal{H}_n^\epsilon$ in the vicinity of the optimum . . . 64
3.4.2.1 Choosing a well-separated subset $\mathcal{M}_n^\epsilon \subset \mathcal{H}_n^\epsilon$ . . . 64
3.4.2.2 Applying Fano's inequality . . . 68
3.4.2.3 Upper bounding $\max_{v,v'} \frac{1}{n} \sum_{j=1}^n D(\mathbb{P}^j_{v_j} \| \mathbb{P}^j_{v'_j})$. . . . 71
3.4.2.4 Completing the proof . . . 73
3.4.3 Proof of Theorem 10: upper bound for optimizing $\mathcal{H}_n^\epsilon$ . . . 74
3.4.4 Proof of Theorem 11: general query lower bound for optimizing $\mathcal{H}_n^\epsilon$ . . . 75
3.5 Further discussion and open questions . . . 78

4 Classical Algorithms for Random Shallow 2D Quantum Circuits, I: Technical Exposition 81
4.1 Introduction . . . 81
4.1.1 Our results . . . 84
4.1.2 Provable complexity separations . . . 85
4.1.3 Conjectures for uniform architectures . . . 86
4.2 Simulation by reduction to 1D dynamics . . . 88
4.2.1 Specification of algorithm . . . 90
4.2.2 Computing output probabilities with `SEBD` . . . 95
4.2.3 Example: `SEBD` applied to cluster state with Haar-random measurements (`CHR`) . . . 97
4.2.4 Conjectured entanglement spectrum of unitary-and-measurement dynamics in an area-law phase . . . 102
4.3 Rigorous analysis of SEBD for the "extended brickwork architecture" 104
4.4 Numerical results . . . 107
4.5 Analytical evidence for conjectures from statistical mechanics . . . 112
4.5.1 Overview . . . 112
4.5.2 Quasi-entropy . . . 113
4.5.3 Mapping . . . 114
4.5.4 Special case of $k = 2$ . . . 116
4.5.5 Mapping applied to general 2D circuits . . . 116
4.5.5.1 The classical stat mech model . . . 117
4.5.5.2 Eliminating negative weights via decimation when $k = 2$ . . . 118
4.5.5.3 Allowed domain wall configurations and disorder-order phase transitions . . . 118
4.5.5.4 Efficiency of `SEBD` algorithm from stat mech . . . 120
4.5.6 Depth-3 2D circuits with brickwork architecture . . . 122
4.5.6.1 Stat mech mapping for general k. . . . 123
4.5.6.2 Simplifications when $k = 2$. . . . 123
4.6 Future work and open questions . . . 125

5 Classical Algorithms for Random Shallow 2D Quantum Circuits, II: Details and Derivations **127**
5.1 General description and justification of the stat mech mapping 127
5.1.1 Generalized mapping procedure 127
5.1.1.1 Setup. 127
5.1.1.2 Goal. 128
5.1.1.3 Generalized interaction weights. 130
5.1.1.4 Justification of stat mech mapping 131
5.1.2 Mapping applied to 1D circuits with weak measurements . . . 134
5.1.2.1 Mapping to the honeycomb lattice. 134
5.1.2.2 Weak measurement and diagonal weights. 135
5.1.2.3 Eliminating negative weights via decimation when $k = 2$. 137
5.1.2.4 Phase diagram. 138
5.1.2.5 Connection between (dis)order and scaling of entanglement entropy. 139
5.1.2.6 Relationship to numerical simulation of `SEBD` on `CHR`. 140
5.1.2.7 Additional observations appearing in previous work. 140
5.1.3 `Patching` 141
5.2 Efficiency of `Patching` algorithm from stat mech 145
5.2.1 Disordered stat mech model suggests `Patching` is successful. . 146
5.2.2 Ordered stat mech model suggests `Patching` is unsuccessful. . 147
5.3 Relation to worst-to-average-case reductions based on truncated Taylor series 149
5.3.1 Implications for reductions based on truncated Taylor series . 150
5.3.1.1 Background: truncated Haar-random circuit ensembles and polynomial interpolation 151
5.3.1.2 Limitation of the interpolation argument 152
5.3.1.3 Inapplicability to `SEBD` and `Patching` 153
5.4 Deferred proofs 154

A Background on stochastic gradient and mirror descent **165**
A.1 Gradient descent 165
A.2 Mirror descent 167

List of Figures

4-1 Schematic depiction of SEBD simulating a shallow 2D circuit 91
4-2 Iteration of SEBD 92
4-3 Extended brickwork architecture with n qubits 104
4-4 Rényi half-chain entanglement entropies S_k versus sidelength L in the effective 1D dynamics for the CHR and brickwork models 108
4-5 Typical half-chain entanglement spectrum $\lambda_1 \geq \lambda_2 \geq \ldots$ observed during the effective 1D dynamics of CHR 109
4-6 Example of stat mech mapping applied to a circuit diagram with 4 qudits and 5 Haar-random gates 114
4-7 The graph produced by the stat mech mapping on shallow 2D circuits 117
4-8 The stat mech mapping yields nodes arranged within a roughly $\sqrt{n} \times t \times d$ prism 121
4-9 Result of stat mech mapping applied to brickwork architecture 122
4-10 Decimated stat mech model associated with brickwork architecture . 124

5-1 Graphical depiction of Haar integration formula given 132
5-2 Disjoint part that forms tensor network representation of $\mathbb{E}_U(Z_{k,\emptyset/A})$ after performing integrals over Haar-random gates 133
5-3 Summary of series of maps for Haar-random 1D circuits with weak measurements 135
5-4 Phase diagram showing for which values of q the anisotropic Ising model on the triangular lattice is ordered and disordered 139
5-5 Patching 142
5-6 Illustration for proof of exponential post-measurement entanglement decay in random depth-2 1D state 161

List of Tables

2.1 Upper bounds for the query complexity of optimizing $f(\theta)$ 44

3.1 Notation and parameters 48

Chapter 1

Introduction

In 1981, in a talk titled *Simulating Physics With Computers* [Fey82], delivered at the Conference on the Physics of Computation at MIT, Richard Feynman suggested that computers exploiting quantum effects may be capable of simulating physical systems that classical computers cannot, remarking that "nature isn't classical, dammit, and if you want to make a simulation of nature, you'd better make it quantum mechanical, and by golly it's a wonderful problem, because it doesn't look so easy."

With four decades of hindsight, this quote indeed seems prescient on all accounts: we still aren't able to efficiently simulate nature — that is, quantum mechanical systems — with classical computers except for in some (very important!) special cases, we do know of algorithms for *quantum* computers which could perform some quantum simulation tasks beyond the capabilities of known classical algorithms,[1] and we can certainly conclude that Feynman was right about quantum computers not being so easy to build.

Indeed, despite interest in the program of building a quantum computer being further spurred by Peter Shor's 1994 discovery of a quantum algorithm for efficiently factoring large integers [Sho97] — an algorithm which would break many modern cryptosystems that rely on an assumption that factoring is hard — cutting-edge programmable quantum devices today have just a few dozen noisy qubits. Perhaps this number will rise to a few hundred or a few thousand by the end of the decade. In comparison, factoring a 2048-bit RSA integer via Shor's algorithm would require thousands of logical qubits, which may translate into millions of physical qubits after accounting for overhead incurred by error correction. Estimates for the quantum resources needed to fulfill Feynman's original vision of simulating interesting quantum many-body systems beyond the reach of modern classical algorithms, via well-known quantum simulation approaches, are also pessimistic from the perspective of today's available hardware.

While implementations of Shor's algorithm or standard quantum simulation algorithms for problems outside the reach of classical algorithms remain in the distant future due to their significant overheads, it nevertheless is true that even some of the programmable quantum devices that exist today, consisting of tens of qubits rather

[1] For general background on quantum computing and quantum information more broadly, see for instance the classic reference [NC00].

than tens of thousands or millions, are already incapable of being efficiently simulated using modern supercomputers. Intuitively, this is due to the exponential blow-up in the Hilbert space dimension associated with an extensive quantum system; for example, $O(2^n)$ real numbers are needed to uniquely specify the state of an n-qubit system. Due to this fact, classical computers today are already incapable of even *storing* an arbitrary state of a quantum device consisting of more than approximately 50 qubits. This observation raises the tantalizing possibility that even though such small quantum computers — recently the phrase "noisy, intermediate-scale quantum" (NISQ) devices [Pre18] has caught on — consisting of a few dozen or a few thousand noisy qubits might not be capable of running quantum computing's flagship algorithms at scale, they might still be capable of doing *something* interesting beyond the capabilities of any existing classical computer. This thesis is concerned with developing the theory of near-term quantum computing, which along the way naturally necessitates a more fundamental exploration of when quantum circuits can be efficiently simulated on a classical computer.

Having realized that NISQ devices appear to be computationally far less powerful than scalable, error-corrected quantum computers, yet nonetheless hard to simulate classically — there are two natural ways to proceed. First, one might ask if the quantum computing community is *sure* that there are no practical applications of such NISQ devices; we're confident that we can't use them to break RSA cryptosystems with Shor's algorithm, but could there be other algorithms for these devices which are more tolerant of small qubit numbers and noise? Shor's algorithm and the well-known quantum simulation algorithms come with provable guarantees on their performance but also have heavy quantum resource requirements beyond the realm of NISQ devices. Could there be heuristic algorithms for these devices which perhaps lack provable guarantees and are hard to analyze theoretically, but nonetheless admit a quantum speedup for some practical problem? For inspiration, one could look to a field like deep learning in which there is a large gap between theory and practice; despite the algorithms often having only weak theoretical guarantees on their performance, and sometimes not even being well-understood theoretically, they nonetheless can perform very well in practice. One candidate class of quantum algorithms for filling this role, which will be discussed later in this chapter, is that of *variational quantum algorithms* (VQAs).

Second, one might concede that asking for a quantum speedup for a practical problem via a NISQ device is too much to ask for, but ask instead if a NISQ device might be used to solve a *useless* problem which we strongly believe cannot be solved efficiently (i.e. in polynomial time)[2] on a classical computer. That is, does there exist some well-defined computational problem which can be efficiently solved by a NISQ computer, but cannot be solved by a classical computer? If a quantum de-

[2]An algorithm runs in polynomial time if its asymptotic runtime is upper bounded by $\text{poly}(n)$, where n is the problem size (in bits) and $\text{poly}(n)$ represents any polynomial in n. In complexity theory, polynomial time is usually equated with "efficient." Superpolynomial time algorithms (algorithms with runtime $n^{\omega(1)}$) are usually considered "inefficient." For a primer on asymptotic notations like $\omega(1)$, see for example `https://en.wikipedia.org/wiki/Big_O_notation`.

vice accomplishes such a feat, we say that it has demonstrated *quantum supremacy*;[3] the research program of quantum supremacy involves proposing candidate computational problems for demonstrating supremacy, collecting strong complexity-theoretic evidence for classical intractability, executing the computational task on an actual quantum device, and verifying that the device is indeed successfully performing the task. Quantum supremacy researchers are indifferent to whether the computational problem in question is practically useful in any way, because even demonstrating a substantial quantum speedup for a "useless" problem would be an extremely noteworthy achievement for at least three related reasons. First, it would represent a landmark engineering feat on the way to scalable, fault-tolerant quantum computers. Second, it would help put to rest some arguments of quantum computing skeptics (e.g. [Lev03; Kal11]) which posit that superpolynomial quantum computational speedups over classical computers may be impossible due to insurmountable engineering limitations or even for more fundamental reasons. (For example, maybe there's some fundamental law of nature forbidding superpolynomial quantum speedups, which will only be discovered when we build a quantum computer and attempt to perform a classically intractable computation. Of course, such a shocking outcome might be even more interesting than demonstrating a quantum speedup!) Third, it would provide strong evidence against the *Extended Church-Turing Thesis*, which essentially posits that a probabilistic Turing machine can simulate any physically realistic model of computation with at worst polynomial overhead. Viewed through the lens of theoretical computer science, a resolution of the Extended Church-Turing Thesis would have profound implications for the nature of our universe; a positive resolution would essentially imply that *any* sort of computer we might build that's compatible with the laws of physics, including a quantum computer, could itself be simulated efficiently on classical computer. Clearly most quantum computing researchers aim to move in the direction of falsifying this conjecture.

Of course, the distinction drawn above between using a NISQ device to obtain a practical quantum speedup and using a NISQ device to demonstrate quantum supremacy is blurry. A NISQ device that can successfully factor huge integers would be both practically useful and would be demonstrating quantum supremacy, because the factoring problem appears to be classically hard.[4] However, one could imagine that a NISQ device running some heuristic algorithm might achieve a modest

[3]One can make several different definitions of what "quantum supremacy" really means. For instance, we alternatively don't need to require that the quantum speedup is asymptotically superpolynomial over any classical algorithm; maybe any speedup should be enough. We could also define quantum supremacy to be demonstrated if the quantum device outperforms any *known* classical algorithm for the task at hand, even if evidence that there does not exist *any* classical algorithm remains weak. In this exposition, we take the definition requiring evidence that the Extended Church-Turing Thesis is violated, which entails a superpolynomial speedup over all classical algorithms, under weak complexity-theoretic assumptions.

[4]However, the evidence for the classical hardness of factoring is not as strong as one might hope. To the best of my knowledge, the evidence is simply the fact that many researchers have looked for an efficient algorithm over the course of several decades without success (at least, that the public is aware of). In particular, the existence of a polynomial-time classical factoring algorithm would have no broader complexity-theoretic consequences of note. This is in contrast to, say, a polynomial-

computational edge over its classical competition for, say, some quantum many-body simulation problem. While this could be practically useful, it might not be a great demonstration of quantum supremacy without a strong theoretical underpinning. For one, we'd like a quantum supremacy claim to be asymptotic in nature, saying something of the form (as a completely arbitrary example): "For an input of length n, a quantum algorithm solves the task in time $O(n^2)$, while any classical algorithm has asymptotic runtime at least $\exp\big(\log^3 n\big)$." The factoring problem has a natural notion of input size: n corresponds to the bit-length of the number to be factored. However, if the problem of interest is to, say, compute the potential energy landscape of the water molecule, there is no immediate notion of problem size and therefore we could not straightforwardly use this problem to claim an asymptotic quantum speedup.

Furthermore, even if the problem does have a notion of input size, we might not have great confidence that the observed speedup truly represents the sort of asymptotically superpolynomial speedup which is often implicit in the phrase "quantum supremacy"; indeed, without a strong theoretical underpinning, it might not even be apparent that the observed speedup is due to quantum effects (as was the case with claims of quantum speedups made by D-Wave).

Finally, even if there does truly appear to be a superpolynomial quantum speedup over all *known* classical algorithms, the evidence that there does not exist *any* efficient classical algorithm could be far weaker. Even in the case of factoring, the evidence that there does not exist any classical algorithm is not very strong, consisting only of the fact that many researchers have tried to find one over the past few decades and have failed (as far as we know). In the context of quantum supremacy, one is interested in basing the proposal on the weakest possible assumptions on the hardness of classical simulation (i.e. on assumptions that are most strongly believed to be true); for a quantum supremacy researcher, a proposal whose classical hardness is implied by the (unproven but widely believed to be true) assumption that P $\neq$ NP would be superior to a proposal based on the assumption that the factoring problem is classically hard (which has evidence in favor of truth but for which the relevant research communities do not have great confidence in its truth), even if only a "useless" computational task is being solved in the former case and the useful task of factoring is being solved in the latter.[5]

To summarize, in looking for speedups for practical problems with NISQ devices, we don't mind exploring heuristic algorithms which may be lacking in provable guarantees on their performance, and we are not particularly concerned about the strength of the evidence that the quantum speedup achieved is provably asymptotically super-

time classical algorithm for exactly sampling from the output distribution of an arbitrary quantum circuit, whose existence would imply that the polynomial hierarchy collapses [TD04].

[5]Of course, it would be best to have an unconditional quantum supremacy proposal, not requiring *any* complexity-theoretic hardness assumptions. Unfortunately, such a proposal seems far to strong to hope for in the near future. This is simply a reflection of the fact that unconditional separations of complexity classes tend to be extremely difficult to prove. This is the same reason for why we don't expect that it will be so easy to formally prove that quantum computers can solve decision problems that classical computers cannot: such a proof would immediately imply that P $\neq$ PSPACE (since quantum circuits can be simulated in PSPACE) [NC00], and hence also solve a major open problem in classical complexity theory.

polynomial with respect to all classical algorithms. In contrast, in using the NISQ device for demonstrating quantum supremacy, we don't care about whether the device is solving a practically useful problem, but we do care about selecting a computational task which is believed to require superpolynomial time for any classical algorithm. This thesis explores both of these prongs of the overarching question of "What can we do with a near-term quantum computer?", investigating both the theory of hopefully practical NISQ algorithms, and the theory of quantum computational supremacy. To be more precise, in the former category it contributes new theoretical results on variational quantum algorithms (VQAs). In the latter category, it contributes new theoretical and numerical results on *random circuit sampling* (RCS), one of the leading proposals for demonstrating quantum supremacy with a near-term quantum device; along the way, it develops novel classical simulation algorithms for certain classes of short-time chaotic quantum dynamics, which may be of interest more broadly outside the context of quantum supremacy.

1.1 Organization and bibliographical information

The main body of this thesis is based on two papers. The first is [HN21] — joint work with Aram Harrow — which was published in PRL and presented at QIP'19 . The second is [Nap+19] — joint work with Rolando La Placa, Alex Dalzell, Fernando Brandão, and Aram Harrow — which is in submission and was presented at QIP'21.

The remainder of this chapter provides high-level background on variational quantum algorithms (in Section 1.2) and on demonstrating quantum supremacy via random quantum circuits (in Section 1.3), before summarizing the results and involved techniques (in Section 1.4). Chapter 2 and Chapter 3 reproduce the contents of [HN21] and contain the results relating to NISQ algorithms, while Chapter 4 and Chapter 5 reproduce the contents of [Nap+19] and contain the results related to random circuits and quantum supremacy. Chapter 2 and Chapter 4 provide technical expositions, while Chapter 3 and Chapter 5 contain auxiliary material and analysis, as well as proofs and derivations deferred from the former chapters. A reader interested only in a high-level and accessible overview of the work contained in the thesis need only read this chapter. A reader interested in a more detailed and technical exposition, but not necessarily in full proofs and auxiliary results, should additionally read the technical overview chapters. The chapters associated with [HN21] may be read independently from the chapters associated with [Nap+19].

1.2 Variational quantum algorithms (VQAs)

As discussed previously, NISQ computers will suffer from relatively small qubit counts as well as noise which will limit the coherence time of the quantum state. In addition to these limitations, these devices might furthermore be capable of implementing only certain classes of quantum gates, and might also have locality restrictions due to the underlying circuit architecture. For example, the qubits might be arranged in a square lattice, with only nearest-neighbor gates being feasible to apply.

A broad class of quantum algorithms which can be forgiving of the above limitations is that of variational quantum algorithms. To introduce the topic of VQAs, we start by motivating them by discussing a particular variety of VQA — also one of the most promising and popular to study — known in the literature as the variational quantum eigensolver (VQE) [Per+14].

To this end, suppose we have access to some quantum device capable of implementing some set of gates, and we may optionally assume that the device has some maximum coherence time. The particular specifications of the device regarding qubit count, architecture, available gate set, etc., are not important to the discussion. In this exposition, we'll assume for the sake of simplicity that the device is noiseless (at least, up until the maximum coherence time), although this assumption may be relaxed.

The computational problem we're trying to solve is as follows: given a description of some physical Hamiltonian H, use the quantum device to estimate the ground state energy of H. H could be, for example, a molecular Hamiltonian (in a quantum chemistry context), or could correspond to some other strongly-interacting quantum many-body model of interest which is hard to simulate classically (such as the Fermi-Hubbard model). The ground-state energy problem is of practical interest; for example, in the chemistry context this information could be used to predict molecular structure and chemical reaction rates.

The standard way of doing this with a quantum computer is via the quantum phase estimation algorithm in conjunction with a Hamiltonian simulation algorithm. However, this approach is unrealistic for a NISQ device; it involves the simulation of e^{-iHT}, where $T \propto \epsilon^{-1}$ if ϵ is the error in the final answer. This implies that the required coherence time for the simulation will generally scale inversely with the desired precision. If a precise answer is desired, long coherence times will be required, likely necessitating quantum error correction schemes and their daunting overheads.

Approaching this problem with a VQA allows one to potentially solve it with much shorter coherence times; notably, with coherence times that are independent of the desired precision. The idea is that, if one can use the quantum device to prepare some parameterized family of quantum states and perform simple quantum measurements, then one can optimize over this variational family to search for an approximate ground state and estimate its energy. More concretely, suppose that for some Hermitian operators A_i with $i \in [p]$[6], and for some easy-to-prepare n-qubit fiduciary state $|\Psi\rangle$, the quantum device can prepare the state

$$|\boldsymbol{\theta}\rangle := |\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_p\rangle = e^{-iA_p\boldsymbol{\theta}_p/2} \cdots e^{-iA_1\boldsymbol{\theta}_1/2} |\Psi\rangle \tag{1.1}$$

for any $\boldsymbol{\theta} \in \mathbb{R}^p$. ($\boldsymbol{\theta}$ is a p-dimensional real vector, and $\boldsymbol{\theta}_i$ denotes its i^{th} component.) The quantum device can be used to estimate $\langle\boldsymbol{\theta}|H|\boldsymbol{\theta}\rangle$ for any $\boldsymbol{\theta} \in \mathbb{R}^p$ under mild assumptions on H and on the set of measurements the quantum device is capable of making. For example, any Hermitian H can be written as a linear combination products of Pauli operators as $H = \sum_i \alpha_i P_i$, where $\alpha_i \in \mathbb{R}_+$ and P_i is a Pauli

[6]For $p \in \mathbb{Z}_+$, $[p] := \{i : i \in \mathbb{Z} \text{ and } 1 \leq i \leq p\}$.

string (that is, a tensor product of n Pauli operators acting on the n qubits). To estimate $f(\boldsymbol{\theta}) := \langle \boldsymbol{\theta}|H|\boldsymbol{\theta}\rangle$, one can use the quantum device to estimate each term $\langle \boldsymbol{\theta}|P_i|\boldsymbol{\theta}\rangle$, and then combine these estimates into an estimate of $f(\boldsymbol{\theta})$ via the relation $f(\boldsymbol{\theta}) = \sum_i \alpha_i \langle \boldsymbol{\theta}|P_i|\boldsymbol{\theta}\rangle$. To estimate a term such as $\langle \boldsymbol{\theta}|P_i|\boldsymbol{\theta}\rangle$, the quantum device can prepare many copies of $|\boldsymbol{\theta}\rangle$, measure P_i with respect to each of them independently (which can be done in a single timestep if the device can measure arbitrary single-qubit Pauli operators in parallel), and average the results. Since the eigenvalues of P_i are ± 1, by the Central Limit Theorem one can estimate $\langle \boldsymbol{\theta}|P_i|\boldsymbol{\theta}\rangle$ up to error ϵ using $O(\epsilon^{-2})$ samples (a "sample" consisting of a preparation of $|\boldsymbol{\theta}\rangle$ and a measurement of the observable P_i).

One important point to note here is that, in general, this process cannot be simulated efficiently using known classical algorithms. If $|\boldsymbol{\theta}\rangle$ is an n-qubit state, then simulating the process classically by directly manipulating the state vector requires $O(2^n)$ bits of memory which rapidly becomes intractable for reasonably large n. More clever classical algorithms that do not store the entire statevector and run in time $\text{poly}(n)$ are not known in general. Another important point is that, in sharp contrast to the approach to this problem via phase estimation, the required quantum coherence time is independent of the desired precision of the estimate of $f(\boldsymbol{\theta})$ for a fixed $\boldsymbol{\theta}$. If one desires a more precise estimate of $f(\boldsymbol{\theta})$, one will need more samples to increase the accuracy of the estimated $\langle \boldsymbol{\theta}|P_i|\boldsymbol{\theta}\rangle$ and therefore the accuracy of $f(\boldsymbol{\theta})$; this may increase the overall runtime of the procedure, but not the required quantum coherence time. It is this fact that makes the variational approach particularly appealing for near-term quantum devices.

We have discussed how to use the quantum device to estimate $\langle \boldsymbol{\theta}|H|\boldsymbol{\theta}\rangle$ for any $\boldsymbol{\theta} \in \mathbb{R}$. To see how to use the device to estimate the ground state energy of H, note that if the true ground state is contained in the variational family and corresponds to the parameter vector $\boldsymbol{\theta}^*$, then $\boldsymbol{\theta}^* \in \text{argmin}_{\boldsymbol{\theta}} \langle \boldsymbol{\theta}|H|\boldsymbol{\theta}\rangle = \text{argmin}_{\boldsymbol{\theta}} f(\boldsymbol{\theta})$. Hence, to approximately find the ground state energy, one may search for the minimum of $f(\boldsymbol{\theta})$, where the minimization over $\boldsymbol{\theta}$ is performed by a classical "outer loop". That is, a classical algorithm is used to search for a minimum of $f(\boldsymbol{\theta})$, with the quantum device being used as an "inner loop" to estimate $f(\boldsymbol{\theta})$.

This is the idea of VQAs, but there are many details one must address in implementing such an algorithm in practice for a particular problem. For instance, how should the variational ansatz in Equation (1.1) be chosen? There is much discussion of this question in the literature. Broadly speaking, there are two approaches to choosing the ansatz. One approach is to simply use an ansatz that is native to the available hardware. In this case, the operators $\{A_i\}_i$ in Equation (1.1) may correspond to interactions which may be easily applied in the given hardware. Some downsides to hardware-motivated ansätze include their potential difficulty of optimization and the fact that they do not well capture the true ground state. The other approach is to choose an ansatz which is theoretically well-motivated. In the quantum chemistry setting, one promising ansatz (as proposed in the original VQE paper [Per+14]) is the unitary coupled-cluster ansatz. Another well-motived ansatz is the *Hamiltonian variational ansatz* as proposed in [WHT15], in which the $\{A_i\}_i$ correspond to terms of the local Hamiltonian to be simulated. The downside to theoretically-motivated

ansätze is that, depending on the Hamiltonian and on the hardware, they may be significantly more difficult to implement. There is also the question of how the minimization of $f(\boldsymbol{\theta})$ should be performed; that is, what classical optimization algorithm should be employed, and how exactly a given $f(\boldsymbol{\theta})$ should be estimated. There are many proposals in the literature, and probing this question occupies a portion of this thesis and is discussed in more detail later in this chapter.

So far we have focused our discussion on using VQAs for quantum simulation, and in particular on estimating the ground-state energy of a many-body quantum system. VQAs for finding excited states and simulating quantum dynamics have also been proposed. While quantum simulation is the most well-motivated use case of VQAs and perhaps the most likely arena to see the first practical quantum speedup, a convenient feature of this class of algorithms is that they can be applied to many diverse types of computational problems, including combinatorial optimization problems and other classical problems. The most famous example is the Quantum Approximate Optimization Algorithm (QAOA) [FGG14], in which one chooses H to be a local Hamiltonian whose ground state exactly corresponds to the solution of a combinatorial optimization problem, and chooses the variational ansatz in a certain theoretically-motivated way. VQAs have additionally been proposed for machine learning, solving linear systems, factoring, quantum error correction, and quantum compiling. Currently, there is not strong evidence that variational algorithms for classical problems should admit quantum speedups over the best known classical algorithms for these tasks, but on the other hand it's not known how to classically simulate these VQAs. One may consult [Cer+20a] for a more detailed recent review of candidate applications of VQAs.

1.3 Random quantum circuits and quantum computational supremacy

We previously discussed the notion of quantum supremacy. To recap, the goal of a quantum supremacy experiment is to perform some computational task on a quantum device that challenges the Extended Church-Turing Thesis (ECT) — a conjecture that essentially states that any reasonable model of computation can be simulated on a classical computer with only polynomial overhead. To challenge the ECT, the computational task in question does not need to solve a practical problem, but there should be significant evidence that there does not exist an efficient (i.e. polynomial-time) classical algorithm for the task.

Modern proposals for quantum supremacy experiments usually involve *sampling* problems rather than *decision* problems[7], the latter being more intuitive and much more commonly studied in theoretical computer science.

[7]Decision problems ask for a yes/no answer (e.g., does this graph admit a 3-coloring?). In other words, for an input x, the goal is to output $f(x)$ where f is some boolean function. Note that the problem of factoring an integer can be equivalently posed as a decision problem: given N and k, does there exist an integer d such that $1 < d \leq k$ and d divides N?

A sampling problem asks, given an input string x, for one to generate a sample according to some probability distribution $\mathcal{D}_x$ defined as a function of x. As a natural example, suppose x encodes the description of a quantum circuit, and $\mathcal{D}_x$ is the *output distribution* associated with circuit x, defined to be the distribution over output strings induced by measuring each qubit in the computational basis after applying the quantum circuit described by x. Then the following is an example of a sampling problem: given a quantum circuit description x, generate a sample from $\mathcal{D}_x$. Note that there is a trivial quantum algorithm which executes this task: just implement the circuit x, measure all qubits, and report the measurement result. The task of sampling from $\mathcal{D}_x$ appears to be much more challenging for a classical computer, however, as a straightforward classical simulation of the circuit x would incur an exponential overhead. Indeed, it was shown by Terhal and DiVincenzo in 2002 [TD04] that there does not exist any efficient classical algorithm for sampling from $\mathcal{D}_x$ in general unless the polynomial hierarchy collapses to the third level, a consequence considered by most of the theoretical computer science community to be very unlikely. A stronger implication of their work, which is of direct relevance to this thesis, is that their hardness result applies even to the restricted class of 2D, depth-3 quantum circuits.[8]

Modern supremacy proposals typically involve sampling problems rather than decision problems for two reasons. The first is a matter of practicality: unlike candidates such as factoring that are based on decision problems, many candidates based on sampling problems have the potential to be implemented on NISQ devices in the near term, possibly without incurring the very substantial overhead associated with quantum error correction. The second reason is more theoretical: it appears likely that stronger complexity-theoretic evidence might be obtained for the classical hardness of proposals based on sampling. Considering again our examples mentioned above, the non-collapse of the polynomial hierarchy is considered to be a much weaker conjecture (that is, more likely to be true) than the conjecture that there does not exist a polynomial-time classical algorithm for factoring. Therefore, from the standpoint of demonstrating quantum supremacy, a sampling-based experiment whose classical hardness rests on the assumption that the polynomial hierarchy does not collapse could be superior to a demonstration based on factoring, despite the fact that the factoring problem is practical. [9]

While the 2002 result of Terhal and DiVincenzo implies that there cannot exist an efficient classical algorithm that samples from the output distribution of an arbitrary

[8] We refer to a quantum circuit as 2D if its qubits are arranged in a rectangular array, and every gate acts on either a single qubit, or on a pair of adjacent qubits. In general, the local dimension of each site could be greater than two, in which case it would be more accurate to refer to the sites as qudits (d-dimensional qubits). The *depth* of a quantum circuit is the maximum number of gates applied to any single qubit, not including measurements.

[9] A factoring-based quantum supremacy experiment does have at least one important advantage over sampling-based proposals: it is very easy to verify that a quantum factoring algorithm is working properly (just multiply together the prime factors that the quantum computer outputs and check that the result indeed is the original integer to be factored). On the other hand, verifying that a quantum device is actually generating samples from the distribution it's supposed to be generating samples from could be hard. Verifying the results of sampling-based supremacy experiments is an active area of research, and is outside the scope of this thesis.

depth-3 quantum circuit, there are obstacles in turning this result into a proposal for demonstrating quantum supremacy. One is that, while this result implies that there cannot exist a classical algorithm for simulating an *arbitrary* circuit instance, it cannot guarantee that a specific given instance is hard to simulate. In other words, the Terhal–DiVincenzo result is a result about worst-case complexity; any efficient classical algorithm must fail on *some* circuit instance, but in principle could succeed for the vast majority of instances. Another obstacle is that their result implies that it is hard to *exactly* sample from $\mathcal{D}_x$, but it is more natural to require the hardness of sampling from any *approximate* distribution $\widetilde{\mathcal{D}}_x$ which is "close" to the true distribution; more precisely, we would like it to be classically hard to sample from any $\widetilde{\mathcal{D}}_x$ satisfying $\|\widetilde{\mathcal{D}}_x - \mathcal{D}_x\|_1 \leq 1/\operatorname{poly}(n)$.[10][11] One reason for being more interested in the hardness of approximate rather than exact sampling is that, if the supremacy proposal is implemented on a NISQ device without error correction, the quantum computer itself will only sample from an approximate distribution. Another more fundamental reason is that a classical algorithm capable of efficiently generating samples from $\widetilde{\mathcal{D}}_x$ with $\|\widetilde{\mathcal{D}}_x - \mathcal{D}_x\|_1 \leq 1/p(n)$ for any polynomial $p(n)$ could fool any polynomial-time algorithm that attempts to distinguish samples generated by the simulator from samples generated by the quantum device.

To attempt to address these barriers to demonstrating quantum supremacy, proposals typically make additional classical hardness assumptions to go beyond the sort of worst-case hardness assumptions of the Terhal–DiVincenzo result. While that result is essentially of the form: "there does not exist any classical algorithm that exactly samples from the output distribution of an arbitrary quantum circuit drawn from the class $\mathcal{C}$, assuming the non-collapse of the polynomial hierarchy," many modern supremacy proposals are essentially of the form: "there does not exist any classical algorithm that approximately samples from the output distribution of most quantum circuits drawn from the class $\mathcal{C}$, assuming some strong complexity-theoretic conjecture." The latter sort of result would be much more suitable for demonstrating quantum supremacy than the former; to do so, one could use the quantum device to generate samples from the output distributions of randomly chosen circuit realizations from the class $\mathcal{C}$. The quantum device succeeds in the supremacy demonstration if, for a sufficiently large fraction of circuit instances, it successfully generates samples from the correct output distribution with sufficiently low error. (The precise meanings of "sufficiently large fraction" and "sufficiently low error" depend on the details of the supremacy proposal.)

The three most well-known supremacy proposals of the above form are based on linear-optical networks [AA11], IQP circuits [SB09; BJS10; BMS16], and random

[10] $\|\widetilde{\mathcal{D}}_x - \mathcal{D}_x\|_1 := \sum_\alpha |\widetilde{\mathcal{D}}_x(\alpha) - \mathcal{D}_x(\alpha)|$, where the sum runs over the 2^n possible output strings. This 1-norm distance is equivalent (up to a factor of 2) to the *total variation distance* between probability measures, a very natural distance measure for probability distributions.

[11] [TD04] furthermore implies that it remains hard to sample from an approximate distribution with small "multiplicative error," but we ultimately desire to show hardness of sampling with small "additive error" — equivalently "total variation distance error" or "1-norm distance error" — as defined here.

circuits [Boi+18].[12] In this thesis we study random quantum circuits.[13] While we primarily motivate their study by applications to quantum supremacy experiments, random quantum circuits also find a plethora of applications in quantum information (e.g. quantum pseudo-randomness, scrambling, decoupling) and physics (e.g. operator spreading and entanglement growth under chaotic quantum dynamics). The quantum supremacy proposal based on random circuits is known as *Random Circuit Sampling* (RCS), the task of approximately sampling from the output distribution of a random quantum circuit, with high probability of success over circuit instance. The idea for a supremacy proposal based on RCS originated from an email thread between several quantum computing researchers in 2015, in which the participants came to the conclusion that, for the purpose of performing a supremacy experiment on a very near-term quantum device, an experiment based on random circuits was better than the alternative possibilities from an engineering standpoint [Aar19].

Indeed, RCS was an extremely natural candidate for a supremacy proposal. Experimentally, it was amenable to implementation in the very near-term. Theoretically, there was intuition supporting the belief that random quantum circuits should be hard for classical computers to simulate. The Terhal–DiVincenzo result already implied that exactly simulating random quantum circuits should be classically hard in the worst case — no classical algorithm should work on *all* instances. Essentially, their result needed to be extended from an "exact, worst case" result to an "approximate, average case result." This extension indeed felt plausible, because (in the noiseless setting) efficient classical simulation algorithms are generally only known for classes of quantum circuits that are highly structured; a few examples include Clifford circuits, circuits consisting of matchgates, and 1D circuits which only generate very low entanglement. On the other hand, one would expect that randomizing the gates of a quantum circuit might wash out any structure that would allow for an efficient classical simulation.

After the idea of demonstrating quantum supremacy via RCS was introduced, a number of groups studied the computational complexity of RCS, collecting evidence that RCS should be hard for a classical computer. In [Boi+18; AC17], it was essentially argued that simulating random quantum circuits is hard assuming very strong

[12] There is also a notable class of supremacy proposals based on *single-instance hardness* rather than *average-case hardness* as described above (beginning with [GWD17]); however, the former sort of proposal can be directly related to the latter sort. In this type of proposal, the goal is to sample from the output of a *fixed* (usually 2D, constant-depth) circuit instance, rather than a random instance. The idea is that, in the measurement-based quantum computing picture, performing measurements induces an effective deep, random 1D circuit. Since the effective 1D circuit that is applied is random, similar hardness arguments can be made for this sort of proposal.

[13] "Random quantum circuit" here is vaguely defined, but refers in general to circuits in which at least some subset of the gates are chosen randomly. In this thesis, for we primarily focus on random circuits consisting of Haar-random, 2-local gates acting on qudits with local dimension q which need not in general be 2. That is, suppose we have fixed a 2-local circuit architecture, in the sense that we have specified a pattern of gates, each of which is applied to two qubits. Then, we obtain a random circuit instance with this architecture by drawing each of the gates independently from the Haar measure on the unitary group $\mathrm{U}(q^2)$ (which may informally be thought of as the uniform distribution over this group). In practice the random gates might not be chosen this way, but this model is theoretically natural and is expected to capture the essence of random circuits.

but plausible complexity-theoretic conjectures. In [HM18] it was shown that sufficiently deep random quantum circuits exhibit a property known as *anti-concentration* (which essentially states that the output distribution is close to uniform), which could be viewed as an additional piece of evidence for the classical hardness of simulation.

But, perhaps the piece of theoretical evidence for the classical hardness of RCS that has been held up most frequently is due to Bouland et. al [Bou+19], with a subsequent technical improvement due to Movassagh [Mov19]. The main result of this line of work is essentially that it is classically hard to precisely compute specific output probabilities of random circuits, under only weak complexity-theoretic assumptions. While the computational task of computing output probabilities rather than sampling is unnatural (as even a quantum computer can't straightforwardly do this), this line of work was nonetheless viewed as evidence that simulating random circuits should be essentially as hard as the worst case. One consequence of this thesis, discussed later in the chapter, is that this line of work and much of the intuition underlying the belief that random circuits should be among the hardest possible to simulate is deeply flawed as evidence for the classical hardness of RCS. Indeed, this thesis develops novel classical algorithms capable of efficiently simulating certain classes of random quantum circuits for which much of this "evidence of classical hardness" applies, cautioning against boldly conjecturing that "exact, worst-case" hardness results for classical simulation extend to (ultimately desirable) "approximate, average-case" hardness results.

In 2019, Google claimed to have achieved quantum supremacy via a random circuit experiment performed using a 2D array of 53 superconducting qubits with nearest-neighbor interactions [Aru+19]. While it was true (and remains true) that there are no known classical algorithms capable of simulating their experiment in a reasonable amount of time, the evidence that their experiment directly challenges the Extended Church-Turing Thesis arguably remains weak. This is because the classical hardness of their experiment rests on the sort of strong conjectures mentioned above which have not received widespread scrutiny. In other words, while there is no *known* polynomial-time classical algorithm for simulating Google's experiment, the evidence that no such algorithm exists is weak. One of the goals of this thesis is to obtain a deeper understanding of the asymptotic classical hardness of RCS, and by association, this supremacy experiment and others based on random circuits.

1.4 Overview of results

In this section, we give a brief, high-level overview of the main results of this thesis. More technical summaries of results and techniques may be found in the respective technical overview chapters: Chapter 2 for VQA results, and Chapter 4 for random circuit results.

1.4.1 Variational algorithms

The exposition on variational quantum algorithms above was fairly vague. Assuming one can prepare some variational family of states $\{|\boldsymbol{\theta}\rangle\}_{\boldsymbol{\theta}}$, the goal is to minimize some objective function $f(\boldsymbol{\theta}) := \langle\boldsymbol{\theta}|H|\boldsymbol{\theta}\rangle$ as well as possible. The quantum device is simply used to estimate $\langle\boldsymbol{\theta}|H|\boldsymbol{\theta}\rangle$, while a classical "outer loop" performs an optimization in parameter space to search for a minimum. Little was said about how this minimization should be actually be performed, and what sort of guarantees are known on the performance of such algorithms.

In fact, relatively little is known theoretically or numerically about the performance of such variational algorithms. For example, a major question in designing a variational quantum algorithm is that of which (classical) optimization procedure should be used to perform the optimization; that is, while the quantum device can be used to generate a noisy estimate of $f(\boldsymbol{\theta})$, how should the classical outer loop use this information to actually perform the optimization? As an even more granular question, for a fixed point in parameter space $\boldsymbol{\theta}$, how many state preparations and measurements should be used to estimate $f(\boldsymbol{\theta})$? From the Central Limit Theorem, we expect that the number of quantum state preparations and measurements required to estimate $f(\boldsymbol{\theta})$ with precision ϵ will scale like ϵ^{-2}, so there is a time/precision tradeoff that the algorithm designer must make a decision about. Furthermore, there is the question of what overarching classical strategy should be used. For example, should gradient descent be used to search for a local optimum? Or what about a "derivative-free" method such as the well-known Nelder–Mead algorithm? Making things even more complicated is the fact that the resulting optimization problem of trying to minimize $f(\boldsymbol{\theta})$ is a *stochastic* optimization problem, meaning the optimization algorithm (i.e. the classical "outer loop") cannot directly access $f(\boldsymbol{\theta})$, but rather can only access noisy estimates of $f(\boldsymbol{\theta})$. In the setting of variational algorithms, the source of this noise is the inherent randomness of quantum measurements and is hence unavoidable. It is also crucial to take into account, as the presence of noise can greatly impact convergence rates of optimization algorithms.

Consider again the question of whether gradient descent or Nelder–Mead is a better choice of optimization algorithm for use as an outer loop in a VQA. The details of the Nelder–Mead algorithm are not important to the discussion, but what is important is the fact that Nelder–Mead is an example of a *derivative-free* optimization algorithm. A derivative-free optimization algorithm attempts to minimize the objective function f without utilizing any information about the gradient, or about higher-order derivatives, of the function.

In contrast, gradient descent is the canonical example of a gradient-based optimization algorithm; if we desire to minimize $f(\boldsymbol{\theta})$, and we are at the point $\boldsymbol{\theta}_t$ at timestep t, then we update the current parameter via the rule

$$\boldsymbol{\theta}_{t+1} = \boldsymbol{\theta}_t - \eta_t \nabla f(\boldsymbol{\theta}_t), \tag{1.2}$$

for some $\eta_t \in \mathbb{R}_+$ which controls the stepsize. In other words, in each timestep we take a step in the direction opposite to the gradient at the current point. The parameters

η_t could be independent of t, or could vary with t according to a schedule. Of course, if gradient descent is employed in a VQA, due to the randomness inherent in quantum measurements, it will only be possible to obtain a noisy estimate of $\nabla f(\boldsymbol{\theta}_t)$; in this case it is more accurate to say that the outer loop is implementing *stochastic gradient descent* (SGD) by moving in the direction opposite to that of the noisy, estimated gradient at each time step.

We discussed in Section 1.2 how the quantum device can be used to estimate $f(\boldsymbol{\theta})$. How can the quantum device be used to estimate $\nabla f(\boldsymbol{\theta})$? One method is via finite-differences. In this approach, one writes $\partial_i f(\boldsymbol{\theta}) = \frac{f(\boldsymbol{\theta}+\epsilon\mathbf{e}_i)-f(\boldsymbol{\theta}-\epsilon\mathbf{e}_i)}{2\epsilon} + O(\epsilon^2)$, where $\mathbf{e}_i$ is the unit vector along the i^{th} coordinate axis, and then uses the quantum device to separately estimate $f(\boldsymbol{\theta} + \epsilon\mathbf{e}_i)$ and $f(\boldsymbol{\theta} - \epsilon\mathbf{e}_i)$ via (for example) the method described previously. Note that this method entails a bias–variance tradeoff. The finite difference estimate of the derivative will inherently be a *biased* estimate, due to higher-order derivatives of f. The bias may be arbitrarily reduced by choosing a smaller ϵ; however, reducing ϵ will blow up the variance of the estimate of $\frac{f(\boldsymbol{\theta}+\epsilon\mathbf{e}_i)-f(\boldsymbol{\theta}-\epsilon\mathbf{e}_i)}{2\epsilon}$ for a fixed number of samples used to estimate $f(\boldsymbol{\theta} \pm \epsilon\mathbf{e}_i)$.

A second approach for estimating $\nabla f(\boldsymbol{\theta})$ using a quantum device involves exploiting the analytical form of $\nabla f(\boldsymbol{\theta})$. Recalling that $f(\boldsymbol{\theta}) = e^{-iA_p\boldsymbol{\theta}_p/2} \cdots e^{-iA_1\boldsymbol{\theta}_1/2} |\Psi\rangle$, we may write $\nabla_i f(\boldsymbol{\theta}) = \langle\boldsymbol{\theta}|G_i|\boldsymbol{\theta}\rangle$, where

$$G_i = \frac{i}{2}[U_{(i+1):p} A_i U^\dagger_{(i+1):p}, H];$$

here, $U_{i:k}$ is shorthand for $e^{-iA_k\theta_k/2} \cdots e^{-iA_i\theta_i/2}$. One could therefore estimate $\nabla_i f(\boldsymbol{\theta})$ by using the quantum device to directly measure the observables $\{G_i\}_i$; this differs from the original formulation of VQAs (as introduced in [Per+14]) in which one uses the quantum device to measure H.

One advantage of the second approach to estimating the gradient over the first is that, in using the second approach, one can obtain an unbiased estimate; we discussed previously how a finite-difference approach to estimating the gradient generally yields a biased estimator. On the other hand, estimating the gradient via "analytical gradient measurements" as described above has its own set of downsides. For one, in contrast to the approach based on finite differences, the quantum circuit involved for performing an analytical gradient measurement may require controlled unitary gates to be applied. Controlled unitaries may be challenging to implement on a NISQ device. Furthermore, the estimator for $\nabla f(\boldsymbol{\theta})$ obtained via the analytical gradient approach can have significantly larger variance than that obtained via a finite difference approach in many settings; perhaps this increased variance cancels out any benefit stemming from the fact that this estimator is unbiased.

An important question of both theoretical and practical interest in the design of NISQ algorithms is then: Is performing analytical gradient measurements beneficial in VQAs? A priori, it is not clear that it is, despite the fact that a plethora of works [Li+17; Rom+18; Mit+18; Sch+20; LW18; Ben+19; Kha+19; Sch+19; RA19; Liu+19; ZLW19; Xu+19; Bra+19; Küb+20; Swe+19; LDD19; Cer+20b; Arr+20] have proposed using gradient measurements in VQAs for various purposes. There

exist derivative-free optimization algorithms such as Nelder–Mead which do not use any gradient information whatsoever to perform the optimization. Furthermore, even for gradient-based approaches such as SGD, we've seen that it's possible to estimate the gradient without performing analytical gradient measurements. In fact, one work [GS17] numerically studied whether gradient measurements lead to an improvement in solving MaxCut with a VQA, and found that they did not in the particular numerical experiments which were performed.

We call VQAs that only perform "energy measurements" (measurements of H) zeroth-order algorithms, and VQAs that may additionally perform "gradient measurements" (measurements of $\{G_i\}_i$) first-order algorithms. It is straightforward to define a notion of a k^{th}-order algorithm, in which derivatives of order up to k are measured by exploiting the analytical form (which more generally will involve nested commutators). Optimization approaches based on Nelder–Mead or SGD in which gradients are estimated via finite differences are examples of zeroth-order algorithms, while an approach based on SGD in which gradients are estimated by directly measuring $\{G_i\}_i$ is an example of a first-order algorithm.

We study this question in Chapter 2 and Chapter 3, affirmatively answering the question of whether implementing gradient measurements can significantly improve the convergence of a VQA. To do this, we rigorously analyze the performance of various VQAs used to solve a simple optimization problem. Intuitively, the optimization problem in question is to find an approximate ground state of a non-interacting spin Hamiltonian. That is, given some non-interacting Hamiltonian $H = \sum_i h_i$ where h_i acts on qubit i, the goal is to find a description of a state $|\psi\rangle$ such that $\langle\psi|H|\psi\rangle - E_{g.s.} \leq \epsilon$ where $E_{g.s.}$ denotes the actual ground state energy.[14]

One theorem provides a lower bound on the number of samples needed for *any* zeroth-order algorithm to solve this problem with high probability, showing that the required number of samples is proportional to ϵ^{-2}. This result is proven via information-theoretic techniques, using ideas from statistical learning theory.

Another theorem shows that, on the other hand, if one makes a prudent choice of variational ansatz for the problem at hand then there exists a first-order algorithm capable of solving the same problem with a substantially smaller number of samples, proportional to ϵ^{-1}. In fact, a first-order algorithm accomplishing this is based on a simple stochastic gradient descent. This theorem is proven by utilizing previously-known bounds on the convergence rate of SGD in the purely classical setting, for various stepsize schedules. A third theorem implies that this first-order algorithm based on SGD is in fact optimal for this problem among all possible k^{th}-order algorithms.

Taken together, these theorems demonstrate how analytical gradient measurements in VQAs can, in principle, substantially improve the convergence rate over the "zeroth-order" approach described in the original VQE paper; they also provide the first formal theoretical justification of the use of gradient measurements in the

[14] Of course, the problem as specified may be easily solved with a classical computer since H is non-interacting. To force the problem to be solved with a VQA rather than with a "cheating" brute-force classical algorithm, we introduce a black-box formalism for VQAs (somewhat analogous to the black-box formalism of classical optimization), and prove our results in this setting.

plethora of aformentioned VQA proposals which utilize them. While the results are proved with respect to the simple, non-interacting ground state problem described above, they demonstrate that gradient measurements can be fundamentally more advantageous than energy measurements. Intuitively, gradient measurements provide more information about the location of the true ground state than energy measurements; stochastic gradient descent is capable of utilizing this information gap to converge faster than any zeroth-order approach. Prior to these results, it was plausible that gradient measurements could *never* lead to an improvement over algorithms based on energy measurements. For example, maybe for any first-order algorithm $\mathcal{A}$, there is some zeroth-order algorithm $\mathcal{A}'$, perhaps which approximates the gradients via finite differences, which achieves the same performance. These theorems rule out this possibility.

These results may be especially relevant for problems (such as those encountered in quantum chemistry) in which one desires a very precise answer; that is, problems for which the error ϵ is very small. In this case, an ϵ^{-1} convergence rate could be far faster than an ϵ^{-2} rate. The results suggest that implementing gradient measurements may be important for the convergence rate, and that VQAs may benefit from using this gradient information as part of a gradient-based optimization approach.

As an auxiliary set of results, we derive rigorous upper bounds on the number of quantum state preparations and measurements needed for a VQA to converge within the vicinity of a local optimum. These formal bounds, which we hope will help guide expectations on the convergence rates of VQAs in practice, are obtained by combining known convergence guarantees from classical optimization with different methods of constructing an unbiased estimator for $f(\boldsymbol{\theta})$ or $\nabla f(\boldsymbol{\theta})$ via state preparations and measurements. The bounds depend on various parameters describing the shape of f in the vicinity of the local minimum; an interesting problem for future study is to understand how these various parameters scale with system size for various classes of problems of practical interest.

1.4.2 Random quantum circuits and their classical simulation

In Section 1.3 we discussed how Random Circuit Sampling (RCS) — sampling from the output distribution of a random quantum circuit — is a leading proposal for demonstrating quantum computational supremacy, and also the proposal on which Google's 2019 claims to have demonstrated quantum supremacy are based. We saw that there is some formal complexity theoretic evidence in support of the classical hardness of RCS, but the basic intuition many researchers had for why RCS should be hard for classical computers is easy to state. For every known class of non-trivial noiseless quantum circuits which admits efficient classical simulation, the classical simulation algorithm works by exploiting some special structure. For a given quantum circuit architecture, choosing the gates randomly is intuitively the "most unstructured" thing one could do. Furthermore, it is known that entanglement spreads rapidly in random circuits, and entanglement is intuitively associated with difficulty of classical simulation. For these reasons, coupled with the knowledge that it is hard

to classically simulate *arbitrary* circuit instances [TD04], one might expect random circuits to be among the hardest possible to simulate.

In Chapter 4 and Chapter 5 we show that, surprisingly, this intuition is deeply flawed. In fact, we introduce two novel classical algorithms for RCS, and analyze their efficiency analytically and numerically to conclude that 2D random circuits of sufficiently shallow, constant depth admit efficient classical simulation[15]. This is true despite the fact that the Terhal–DiVincenzo result applies to this family of circuits, implying that there cannot an efficient classical algorithm capable of simulating *all* random circuit instances.

The results of Bouland et al. [Bou+19] and Movassagh [Mov19] discussed previously, showing that it is classically hard to efficiently compute output probabilities of random quantum circuits, also apply to all of the classes of circuits we study. These results had previously been held up as some of the strongest pieces of formal evidence known for the hardness of simulating random circuits. In particular, it had been argued that (1) by proving a worst-case-to-average-case reduction for computing output probabilities, their results confirm the intuition that randomly chosen gates should be essentially as hard as arbitrary gates to simulate, and (2) a technical improvement to their proof technique would enable their proof to extend to the hardness of sampling. In this thesis, we extinguish any hope of extending these particular techniques to proving the classical hardness of RCS. Arguably, in demonstrating that there exist classes of random circuits for which RCS is classically tractable yet these hardness results apply, the previous intuition and theoretical underpinning for the classical hardness of RCS — and by extension, Google's quantum supremacy claim — is significantly weakened.

Interestingly, despite weakening several of the previous reasons in believing in the classical hardness of RCS, these chapters may be viewed as also contributing new evidence in *support* of the hardness of simulating sufficiently *deep* 2D random quantum circuits. This is because the two novel classical simulation algorithms that we introduce appear to experience computational phase transitions as the circuit depth is increased. Roughly speaking, for a particular 2D circuit architecture, we find evidence that there is a critical constant depth d^* such that, in performing random circuit sampling on a $\sqrt{n} \times \sqrt{n}$ square array of qubits of depth d, our algorithms run in time $\mathrm{poly}(n)$ if $d < d^*$, and run in time $\sim \exp\left(n^{\Theta(1)}\right)$ if $d > d^*$.

To give a sense of the source of this computational phase transition, it's necessary to give an overview of how one of the classical simulation algorithms works and its analysis, which makes novel connections between the complexity of quantum random circuit sampling, classical statistical mechanics, and the subject of "unitary-and-measurement" dynamics which has seen an explosion of interest in the condensed matter physics community over the past 2.5 years [LCF18; Cha+19; SRN19; LCF19;

[15] In other words: Take a $\sqrt{n} \times \sqrt{n}$ array of qubits. Imagine that we apply a constant, d, layers of nearest-neighbor, Haar-random gates. (A single layer consists of $O(n)$ random gates applied in parallel with a layout depending on the architecture.) For some constant d^*, independent of n but depending on the structure of the circuit architecture, if $d < d^*$ then our algorithms are conjectured to be efficient, running in time $n^{1+o(1)}$. If $d > d^*$, we expect the algorithms to be inefficient, running in time $2^{n^{\Theta(1)}}$.

SRS19; Cho+20; GH20a; BCA20; Jia+20; GH20b; Zab+20; TZ20; NS20; AB20; Fan+20; Li+20; LAB21; SH20; Ipp+21; FA20; SRS20; Vij20; LP20; LF20; TFD20; FHH21; Nah+21; IK21]. In fact, one of our simulation algorithms can be viewed as an application of this recent line of work.

We'll give an overview of this algorithm here. The idea is to convert the problem of simulating a 2D random circuit of constant depth to a problem of simulating a 1D random quantum circuit of extensive depth. In other words, one of the spatial dimensions of the original circuit to be simulated is converted into a "time" dimension, and the problem of classically simulating a random, $\sqrt{n} \times \sqrt{n}$, 2D, constant-depth (i.e. constant-time) quantum circuit is reduced to an equivalent problem of simulating a 1D quantum circuit of length $\sqrt{n}$ evolving for $\sqrt{n}$ timesteps.[16]

The effective 1D process obtained after performing this reduction looks like a random local 1D circuit with weak measurements interspersed. Fortunately, as mentioned above, very similar sorts of random-unitary-and-measurement have been studied in a plethora of very recent works, which find that there generally appears to be an entanglement phase transition in such processes driven by measurement strength. In particular, if the measurement strength is above some threshold, the dynamics equilibrates to an "area-law" regime in which (with high probability) the state obeys an area law for its entanglement entropy. Meanwhile, if the measurement strength is below the threshold, the dynamics equilibrates to a "volume-law" regime. [17] Empirically, it is often true that 1D dynamics in which the entanglement obeys an area law may be efficiently classically simulated via Matrix Product States. On the other hand, it is not generally known how to efficiently simulate 1D dynamics in a volume law regime (Matrix Product States become inefficient in this regime).

We find a heuristic correspondence between the depth of the 2D circuit to be simulated and the measurement strength in the associated 1D effective dynamics, raising the possibility of a computational phase transition for this simulation algorithm as a function of the 2D circuit depth. For sufficiently low depth, the strength of the measurements in the associated effective 1D dynamics may be sufficiently high that the effective 1D dynamics is in an area-law phase for the entanglement, making efficient simulation possible via MPS. As the depth is increased, the strength of the measurements in the effective 1D dynamics is decreased, eventually decreasing past the critical measurement strength which pushes the dynamics into a volume-law phase, making classical simulation inefficient.

[16] The intuition behind this 2D-to-1+1D reduction is quite similar to that behind measurement-based quantum computing (MBQC). In fact, there is a sense in which this reduction can be understood as a form of noisy, non-adaptive MBQC. To elaborate, there are two ways in which our reduction is different than standard MBQC. First, in MBQC, the gates are generally chosen such that unitary dynamics are induced in the corresponding effective 1D dynamics; in our case, the fact that the gates are random corresponds to the effective 1D dynamics looking like alternating rounds of unitaries and weak measurements. Second, in MBQC, measurements are performed adaptively (i.e. the choice of measurement depends on the previously observed measurement results), while in our case the measurements are non-adaptive.

[17] A 1D quantum state is said to obey an area law if, for any contiguous subregion A, $S(A) \leq O(1)$ where $S(A)$ denotes the entanglement entropy of subregion A. It is said to obey a volume law if $S(A) \geq \Omega(|A|)$, where $|A|$ denotes the number of spins in A.

While this picture has strong intuitive support, rigorously proving that this algorithm is efficient when the 2D circuit depth is sufficiently low is challenging. For one, the entanglement phase transition as a function of measurement strength in 1D unitary-and-measurement circuits remains unproven, despite very strong numerical evidence and significant research efforts from both the condensed matter physics and the quantum information communities. Nonetheless, we make the following contributions.

1. We rigorously prove that this algorithm is efficient, running in time $O(n)$, for some special case families of $\sqrt{n} \times \sqrt{n}$ 2D random circuits for which the hardness results of Terhal–DiVincenzo and Bouland et al./Movassagh apply. Intuitively, these special-case architectures have the property that the measurement strength in the associated effective 1D dynamics is actually increasing as a function of n, rather than remaining constant. A key technical result involved in this proof, which may be of independent interest, is that a random 1D quantum state produced by a constant-depth,[18] local, random circuit obeys a certain post-measurement exponential decay of entanglement property. More specifically, if we perform computational basis measurements on a contiguous block of k qubits and ask how much entanglement there is between the remaining "left" and "right" blocks of the resulting post-measurement pure state, we find that the expected entanglement entropy $\mathbb{E}\, S$ obeys $\mathbb{E}\, S \leq c^k$, for some constant $c < 1$.

2. Implementing the algorithm on a laptop, we find that it is efficient in practice for non-trivial families of random circuits which are highly intractable for all previously known classical simulation algorithms we are aware of, and which go beyond the regime in which we can rigorously prove efficiency. For example, we used the algorithm to approximately sample from the output distributions of a certain depth-3 random circuit family defined on a 400×400 square lattice. More precisely, a laptop running non-optimized code was able to sample from the output distribution up to a total variation distance error of 0.01[19], requiring on the order of one minute per sample. In contrast, the previously best-known classical algorithms for this task are hopelessly intractable for this same problem.[20]

[18] We prove this for depth two, but expect it to hold for any constant depth.

[19] The algorithm has the property of being *self-certifying* in the sense that it can numerically bound its own error, allowing us to be confident that the variation distance error incurred is less than 0.01.

[20] The previously best-known algorithms are exact simulations based on tensor network contraction and have asymptotic runtime $2^{\Theta(\sqrt{n})}$. Meanwhile, we conjecture that the asymptotic runtime of our new algorithm for this same task is $n^{1+o(1)}$. To be more fine-grained, we estimate that for previously known tensor-network based algorithms, this simulation task is roughly equivalent to simulating a depth-40 circuit on a 20×20 lattice with the architecture considered in [Vil+20], where the entangling gates are CZ gates. The task of simulating a depth-40 circuit on a 7×7 lattice was reported to require more than two hours using tensor network contraction on the 281 petaflop supercomputer Summit [Vil+20], and the exponentiality of the runtime suggests scaling this to 20×20 would take many orders of magnitude longer, a task that is decidedly intractable.

3. We numerically study how the performance of the algorithm scales asymptotically with n, and collect strong numerical evidence that the algorithm is not only efficient in practice for non-trivial random circuit families, but is asymptotically efficient.

4. We analytically study a toy model for a 1D random-unitary-and-measurement process which models the effective 1D dynamics associated with a sufficiently shallow 2D random circuit. In particular, we study how the entanglement spectrum equilibrates in such a process. We anticipate that this analysis may be of interest to researchers outside the field of quantum computing who are studying unitary-and-measurement processes for independent reasons. The toy model allows us to conjecture that the asymptotic runtime of our algorithm is $n^{1+o(1)} \cdot \exp\Big(O(\sqrt{\log(1/\varepsilon\delta)})\Big)$, where ϵ (resp. δ) is the maximum total variation distance error permitted (resp. the maximum probability of failure permitted).

5. We propose a second classical simulation algorithm based on a completely different idea. We don't elaborate upon the algorithm in detail here, but the idea is to start by first exactly generating samples from marginals of the output distribution, restricted to spatially disconnected "patches" of the 2D lattice. Since the circuit depth is constant, this can be done efficiently by restricting the circuit to the relevant lightcone for each "patch" to be simulated. These patches are subsequently "stitched together" via recovery maps to generate a global sample.

6. We additionally analyze the performance of both algorithms by utilizing a correspondence between random quantum circuits and classical statistical mechanical models. Previously, this correspondence had been used in other contexts to study random tensor networks [Hay+16] and various properties of random quantum circuits [NVH18]. We employ this correspondence to study the efficiency of our algorithm, and find evidence of a correspondence between the runtimes of our algorithms and the phase (i.e. ordered/disordered) of the associated classical stat mech models. We find that the stat mech models associated with random shallow 2D quantum circuits resemble Ising models, in which the interaction strength is related to the circuit depth and to the local qudit dimension; increasing the circuit depth or the qudit dimension effectively corresponds to an increase in interaction strength (i.e. decrease in temperature) in the corresponding classical stat mech model. The existence of a phase transition in the classical stat mech models provides additional evidence for a computational phase transition for the algorithm between an efficient (polynomial-time) and inefficient (exponential-time) regime.

Studying the classical simulability of random shallow 2D quantum circuits reveals intimate connections between multiple disparate subjects (quantum computing, complexity theory, information theory, statistical mechanics, unitary-and-measurement dynamics), and it also opens up a number of interesting directions for future work. We list some of the major possible directions below.

1. Perhaps the most obvious open question is whether there exists an efficient classical algorithm to simulate random circuits more generally, as our algorithms are only efficient for sufficiently shallow 2D random circuits. One lesson from this thesis is that the randomness of the quantum gates — previously expected for the most part to make the simulation more difficult by eliminating structure — can itself by exploited as a form of structure by a clever classical algorithm. Currently, there is not a particularly strong reason to believe that no such classical algorithm can exist.

2. Another interesting question is whether some version of our simulation algorithms might work for sufficiently shallow D-dimensional circuits for $d > 2$. We expect that the algorithm based on a 2D-to-1+1D reduction will not naturally extend to higher dimensions, but there does not appear to be a fundamental barrier against the other algorithm (based on simulating disconnected "patches" and then stitching them together) from extending to higher dimensions.[21]

3. This thesis is only concerned with *noiseless* circuits. We expect that the simulation might become easier in the presence of noise. That is, we expect that in the presence of some varieties of noise, the critical depth d^* separating the polynomial-time and exponential-time regimes might increase. It would be interesting to better understand this effect, and also investigate whether there are variants of the algorithms we propose which are capable of exploiting noise to make the simulation even easier.

[21] However, if it did extend to higher dimensions, it would run in quasi-polynomial time rather than polynomial time. For a constant-depth circuit with D spatial dimensions, its scaling with n would be $\exp\left(\log^{(D-1)} n\right)$.

Chapter 2

Analytical Gradient Measurements Can Accelerate VQAs, I: Technical Exposition

2.1 Introduction

In recent years, an array of *variational hybrid quantum-classical algorithms* have been widely studied as leading candidates for near-term quantum computers, due to their relatively modest quantum resource requirements and potential of scalability. Variational algorithms have been proposed in the context of quantum simulation (e.g. variational quantum eigensolvers [Per+14; WHT15]), combinatorial optimization (e.g. QAOA [FGG14]), and machine learning (e.g. quantum classifiers [FN18; Mit+18; SK19; Sch+20; Hav+19]).

In the variational setting, one can prepare states belonging to some parameterized family $\{|\boldsymbol{\theta}\rangle\}_{\boldsymbol{\theta}}$ for $\boldsymbol{\theta} \in \mathcal{X} \subset \mathbb{R}^p$, where p is the number of variational parameters. The set of parameterized states which may be prepared will depend on the specifications of the quantum device. We consider parameterizations consisting of p "pulses" applied to some easy-to-prepare starting state $|\Psi\rangle$,

$$|\boldsymbol{\theta}\rangle := |\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_p\rangle = e^{-iA_p\boldsymbol{\theta}_p/2} \cdots e^{-iA_1\boldsymbol{\theta}_1/2} |\Psi\rangle ,$$

where A_j is the Hermitian operator which generates pulse j. This form of parameterization is well-motivated theoretically [McC+16; Yan+17; BJ19] and is widely considered in the literature.

A classical "outer loop" controls the quantum device, which is used only for preparing variational states and making simple measurements. The classical outer loop uses this measurement information to perform a *classical* optimization of some objective function $f(\boldsymbol{\theta})$ over the feasible set $\mathcal{X}$, where $f(\boldsymbol{\theta})$ is induced by some Hermitian *objective observable* H, via the definition $f(\boldsymbol{\theta}) := \langle\boldsymbol{\theta}|H|\boldsymbol{\theta}\rangle$.

Given the ability to prepare variational states $|\boldsymbol{\theta}\rangle$, there remain the questions of what observables should be measured, and how the measurement outcomes should be used by the classical outer loop to find an approximate minimizer for $f(\boldsymbol{\theta})$. Typically,

the objective observable H is decomposed as a linear combination of observables which each can be efficiently measured in low depth. For instance, we may always write a Pauli decomposition $H = \sum_i \alpha_i P_i$, where $\alpha_i > 0$ and P_i are tensor products of Pauli operators. By linearity, it is possible to construct an estimator for $\langle\boldsymbol{\theta}|H|\boldsymbol{\theta}\rangle$ via measurements of the Pauli strings $\{P_i\}_i$.

In this work, we will find it convenient to take a novel but natural approach for estimating the objective function or its derivatives via sampling terms of the Pauli decomposition to measure according to an appropriate distribution; a similar sampling strategy was previously employed in the context of random compiling for Hamiltonian simulation [Cam19]. To this end, we express H as an expectation value, $H = E\,\mathbb{E}_X\,P_X$, where $E := \sum_i \alpha_i$ and the random variable X is distributed as $p_X(x) := \alpha_x/E$. For a given point $\boldsymbol{\theta}$ in parameter space, by linearity we have $f(\boldsymbol{\theta}) = \langle\boldsymbol{\theta}|H|\boldsymbol{\theta}\rangle = E\,\mathbb{E}_X\,\langle\boldsymbol{\theta}|P_X|\boldsymbol{\theta}\rangle$. Hence, an unbiased $\pm E$-valued estimator for $f(\boldsymbol{\theta})$ may be obtained by sampling x from the distribution p_X, measuring P_x w.r.t. $|\boldsymbol{\theta}\rangle$, and then scaling the output by E. With estimates of f obtained in this way, the classical outer loop performs a stochastic zeroth-order (i.e. derivative-free) optimization of the function $f(\boldsymbol{\theta})$; 'stochastic' because of the randomness of the measurement outcomes when estimating $f(\boldsymbol{\theta})$, and 'derivative-free' because the outer loop receives estimates of $f(\boldsymbol{\theta})$ rather than estimates of its gradient $\nabla f(\boldsymbol{\theta})$ or of higher-order derivatives.

However, it is not apparent that such a zeroth-order strategy is best. Indeed, as observed in a number of works (listed in the subsequent section), by performing a slightly more complicated measurement it is possible to directly estimate $\nabla f(\boldsymbol{\theta})$; this estimate can then be used with a first-order (i.e. gradient-based) optimization algorithm. To this end, we may express the j^{th} component of the gradient as an expectation value, $\nabla_j f(\boldsymbol{\theta}) = \langle\boldsymbol{\theta}|G_j|\boldsymbol{\theta}\rangle$, where

$$G_j = \frac{i}{2}[U_{(j+1):p} A_j U_{(j+1):p}^\dagger, H]$$

(see Section 3.2 for a derivation). Here, $U_{j:k}$ is shorthand for $e^{-iA_k\theta_k/2}\cdots e^{-iA_j\theta_j/2}$. To measure G_j with a low-depth circuit, we may expand A_j and H as linear combinations of products of Pauli operators with positive coefficients, obtaining

$$\nabla_j f(\boldsymbol{\theta}) = \Gamma_j \mathop{\mathbb{E}}_{K,L} \langle\boldsymbol{\theta}|\frac{i}{2}[U_{(j+1):p} Q_K^{(j)} U_{(j+1):p}^\dagger, P_L]|\boldsymbol{\theta}\rangle\,,$$

where $Q_k^{(j)}$ are Pauli operators appearing in the expansion of A_j, Γ_j is the sum of coefficients appearing in the resulting expansion, and the joint probability of $(K = k, L = l)$, denoted $q_{KL}(k,l)$, is proportional to the coefficient associated with the term in the expansion including $Q_k^{(j)}$ and P_l. A ± 1-valued unbiased estimator for $\langle\boldsymbol{\theta}|\frac{i}{2}[U_{(j+1):p} Q_k^{(j)} U_{(j+1):p}^\dagger, P_l]|\boldsymbol{\theta}\rangle$ can be obtained with a single measurement via a simple Hadamard-test circuit (as described in [LB17; GS17; Rom+18]; see also Section 3.2). Now, we may construct a $\pm\Gamma_j$-valued unbiased estimator for $\nabla_j f(\boldsymbol{\theta})$ with a single measurement by sampling (k,l) from q_{KL}, measuring the corresponding ob-

servable as described above, and scaling the output by Γ_j. Generalizations of this strategy permit the measurement of higher-order derivatives as well.

Finally, an unbiased estimator $\hat{\mathbf{g}}(\boldsymbol{\theta})$ for the full gradient may be constructed with one measurement by choosing component j with probability $\Gamma_j / \|\vec{\Gamma}\|_1$, estimating $\nabla_j f(\boldsymbol{\theta})$ using the method described above, and then scaling the output by $(\|\vec{\Gamma}\|_1 / \Gamma_j)\hat{e}_j$ where $\hat{e}_j$ denotes the unit vector in direction j. Here we have defined the vector $\vec{\Gamma} := (\Gamma_1, \ldots, \Gamma_p)^\top$. It may be verified (in Chapter 3) that $\hat{\mathbf{g}}$ is $\pm\|\vec{\Gamma}\|_1$-valued, and that $\mathbb{E}\,\hat{\mathbf{g}} = \nabla f$. Note that the choice to sample j with probability proportional to Γ_j is optimal for minimizing $\mathbb{E}\,\|\hat{\mathbf{g}}\|^2$ among all choices of sampling weights (as may be verified via a Lagrange multiplier), and furthermore results in this quantity having no explicit dependence on p.

Our method for constructing unbiased estimators for f and its gradient is effectively a form of importance sampling which assigns higher weight to larger terms in the sum; this is reflected in the fact that the magnitude of an estimator depends on an appropriate *sum* of coefficients, but carries no explicit dependence on the number of terms in the decomposition (or on the number of variational parameters for the gradient estimator). This is especially relevant for applications (such as quantum chemistry) for which many terms of the sum may have small weight. After a preprint of this paper was made public, subsequent works [Swe+19; Arr+20] have numerically studied similar estimators and have furthermore proposed methods of adaptively setting the sampling weights associated with each observable in the expansion [Küb+20; Arr+20].

A fundamental question is now whether, within the vicinity of a local optimum, "first-order" variational algorithms which perform measurements to construct gradient estimators can converge faster than algorithms which use the simpler, "zeroth-order" strategy of estimating only the objective function itself. This question may be especially important in the context of quantum simulation, in which a precise solution is often desired. Within a natural black-box setting, we answer this question affirmatively by exhibiting an optimization problem for which performing gradient measurements, and using these gradient estimates in conjunction with stochastic gradient descent (SGD) [Bub15], converges to an optimum asymptotically faster than any strategy based on measuring the objective function.

The optimization problem we analyze to demonstrate this separation is quite simple: it is essentially the problem of learning the ground state of a 1-local (non-interacting) spin Hamiltonian. While an analytic solution to this problem may be readily derived, the black-box model ensures the variational algorithm behaves in a generic way, rather than merely solving the problem analytically (as this would be computationally infeasible for more complicated problems). This simple problem provides a counterexample to the proposition that, within the natural black box setting defined below, the convergence rate of gradient-based variational algorithms can be generically matched by that of zeroth-order algorithms. In particular, this rules out the possibility that gradient measurements can always be replaced by gradient estimates obtained by finite-differencing energy measurements without a loss of performance. This observation may be of interest in the design of practical NISQ al-

gorithms, in which gradient measurements could be more difficult to implement than energy measurements. Our results demonstrate that one cannot hope to generically simulate gradient measurements while maintaining equivalent performance; hence, incurring extra overhead for measuring gradients could be worthwhile.

The speedup we obtain for gradient-based algorithms crucially relies on using an appropriate choice of variational ansatz for the problem at hand, making our toy model setting more similar to that of variational algorithms with theoretically motivated ansätze rather than those which use a "hardware-efficient ansatz" [Kan+17]. Indeed, the setting of "barren plateaus" [McC+18] in which the ansatz looks random and gradient-based optimization fails may be viewed as the opposite situation to that studied in this work.

While our analysis of a non-interacting system is sufficient to rule out the existence of zeroth-order algorithms which *generically* match the performance of first-order algorithms, we cannot rule out the possibility that certain classes of problems contain additional structure which allows zeroth-order algorithms to match the convergence rate of first-order algorithms. However, we might expect the non-interacting model to exhibit qualitatively similar behavior to that of general models in a disordered phase which flow under RG to non-interacting systems. Furthermore, in the toy model settings we study, the algorithms are constrained to remain within the vicinity of the optimum. Hence, our zeroth-order bounds do not apply to algorithms which may operate far from the vicinity of the optimum to which they are trying to converge. Indeed, in some cases analytic gradient measurements may be performed by performing multiple "non-local" energy measurements and combining the results [Li+17; Mit+18; Sch+19].

2.2 Prior work

Prior works had considered gradient measurements in variational algorithms, but it remained unclear whether they could confer an advantage. It was first observed that gradients could be directly measured in the context of hybrid quantum-classical algorithms in [WHT15], but the authors pointed out that "it is not clear whether or not access to the derivative would improve the optimization". Many subsequent works [Li+17; Rom+18; Mit+18; Sch+20; LW18; Ben+19; Kha+19; Sch+19; RA19; Liu+19; ZLW19; Xu+19; Bra+19; Küb+20; Swe+19; LDD19; Cer+20b; Arr+20] have proposed using gradient measurements in variational algorithms for specific applications, but lacked concrete theoretical evidence for an advantage over zeroth-order algorithms; our work complements these proposals by providing such evidence. In [GS17], algorithms based on gradient measurements were numerically compared against zeroth-order algorithms for the combinatorial optimization problem MaxCut. Interestingly, the authors found no advantage for gradient-measurement-based algorithms for this problem. The discrepancy between our results and theirs could be explained by the possibility that their simulations were dominated by the cost of *finding* good local optima rather than *converging* to a specific local optima (as is our focus in this Letter). Similar questions about the benefit of noisy gradients for

optimization had previously been studied in the purely classical context [Aga+09; JNR12], but fundamental differences between the classical and quantum variational settings prevented these results from being directly applicable in the present setting. Nonetheless, our strategy for proving an advantage for gradient-based variational algorithms adapts some techniques developed in these works, which in turn are inspired by methods from statistical minimax and learning theory.

2.3 Black box model

We now discuss our rigorous separation between the performance of zeroth-order and first-order variational algorithms. As a prerequisite, we first introduce a black-box model for variational algorithms. To see why such a setting is useful, note that a classical computer with unbounded computational resources is capable of simulating any hybrid quantum-classical algorithm; in this sense, no quantum measurements are required. Of course, this simulation will generally require exponential space and runtime, and therefore be intractable. Since one of our goals is to prove lower bounds on the number of quantum measurements required by a variational algorithm, we must impose extra constraints on the classical component of the algorithm to rule out such brute-forcing behavior. A natural way to do this which is also amenable to theoretical analysis is via a model in which the 'quantum' component of the algorithm is only accessible via queries to a black box. Note that our motivation for a black-box model is analogous to the motivation for a black-box model in the context of purely classical optimization, where it has been found to be a highly useful and insightful framework [Bub15].

We now describe the black box setting. We assume the classical outer loop is not given an explicit description of the objective observable H, but rather has access to an oracle $\mathcal{O}_H$ encoding H. Suppose $H = E\,\mathbb{E}_L\,P_L$ as above. The classical outer loop may query $\mathcal{O}_H$ with a variational state ansatz description Θ, a parameter $\boldsymbol{\theta} \in \mathbb{R}^p$, and optionally an index $j \in [p]$. Upon querying $\mathcal{O}_H$ without the optional index, which we call a *zeroth-order query*, the oracle prepares the variational state $|\boldsymbol{\theta}\rangle$ according to the ansatz described by Θ and outputs an unbiased $\pm E$-valued estimate of $f(\boldsymbol{\theta})$ following the sampling approach described above. Similarly, upon querying $\mathcal{O}_H$ with index j, which we call a *first-order query*, the oracle outputs an unbiased $\pm\Gamma_j$-valued estimate of $\nabla_j f(\boldsymbol{\theta})$ following the sampling approach. Higher-order queries to the oracle may be defined analogously, as described explicitly in Chapter 3. In the black-box setting, we say an algorithm is k^{th}*-order* if it only makes queries of order k or lower.

In this oracle model, following the classical optimization literature [Bub15], the classical outer loop is given black-box access to $\mathcal{O}_H$ and may be promised that H belongs to some family $\mathcal{H}$, but is not given explicit knowledge of H. The relevant performance metric of an optimization algorithm in this setting is the query complexity, that is, the number of oracle calls made by the classical algorithm. If the oracles are implemented physically via the observable sampling procedures described above, the query complexity exactly corresponds to the number of quantum state preparations and measurements performed.

To formally state our separation between zeroth- and first-order variational algorithms, it will be necessary to make some additional definitions. Let $\mathcal{H}$ be some fixed set of objective observables, and suppose $\mathcal{A}$ is a (possibly randomized) classical algorithm which has oracle access to $H \in \mathcal{H}$ and outputs a (generally random) description of a quantum state $|\psi\rangle$ from some distribution $\mathcal{D}_H$ which may depend on H. Then the optimization error of $\mathcal{A}$ with respect to $\mathcal{H}$, $\mathrm{Err}(\mathcal{A}, \mathcal{H})$, is defined as

$$\mathrm{Err}(\mathcal{A}, \mathcal{H}) := \sup_{H \in \mathcal{H}} \mathop{\mathbb{E}}_{\phi \sim \mathcal{D}_H} [\, \langle \phi | H | \phi \rangle - \lambda_{\min}(H)],$$

where $\lambda_{\min}(H)$ is the smallest eigenvalue of H, and the expectation is over the possible randomness of the output state $|\phi\rangle$. That is, $\mathrm{Err}(\mathcal{A}, \mathcal{H})$ quantifies the worst-case (over $H \in \mathcal{H}$) expected optimization error of $\mathcal{A}$. In some cases, we will be particularly interested in the setting in which the variational algorithm is close to an optimum and is trying to converge. To this end, it is helpful to define $\mathcal{A}$ to be a *δ-vicinity algorithm* with respect to $\mathcal{H}$ if $\mathcal{A}$ only queries the oracle with descriptions of variational states in the *δ-optimum* of $\mathcal{H}$; this defined to be the set of states $|\boldsymbol{\theta}\rangle$ such that $\langle \boldsymbol{\theta} | H | \boldsymbol{\theta} \rangle - \lambda_{\min}(H) \leq \delta$ for some $H \in \mathcal{H}$.

We now introduce the parameterized family of objective observables which we use to prove our sample complexity separation. First, for any $\delta \in \mathbb{R}$ and $v \in \{-1, 1\}^n$, define the n-qubit observable

$$H_v^\delta := -\sum_{i=1}^{n} \Big[\sin\Big(\frac{\pi}{4} + v_i \delta\Big) X_i + \cos\Big(\frac{\pi}{4} + v_i \delta\Big) Z_i \Big],$$

where X_i (Z_i) denotes the Pauli X (Z) operator acting on qubit i. Now, for a fixed parameter $\epsilon > 0$ we define $\delta(\epsilon) := \sqrt{\frac{45\epsilon}{n}}$ and

$$\mathcal{H}_n^\epsilon := \{ H_v^{\delta(\epsilon)} \ : \ \forall v \in \{-1, 1\}^n \}.$$

We prove lower and upper bounds on the query cost of finding a low-energy state w.r.t. observables in of the family $\mathcal{H}_n^\epsilon$.

2.4 Lower bounds

We now state our lower bound for zeroth-order variational algorithms. (The numerical constants are chosen for ease of proof and have not been carefully optimized.)

Theorem 1 (Lower bound for zeroth-order methods)**.** *For any $n > 15$ and $\epsilon < 0.01n$, let $\mathcal{A}$ be any zeroth-order 100ϵ-vicinity algorithm for the family $\mathcal{H}_n^\epsilon$ that makes T queries to the oracle. Then, if $\mathrm{Err}(\mathcal{A}, \mathcal{H}_n^\epsilon) \leq \epsilon$, it must hold that $T \geq \Omega(\frac{n^3}{\epsilon^2})$ where the implicit factor is some fixed constant.*

The proof of Theorem 1 is information-theoretic, and may be found in Section 3.4. We choose a set $\mathcal{M} \subset \mathcal{H}_n^\epsilon$ that is both large and has well-separated points, then run $\mathcal{A}$ on a randomly chosen $H \in \mathcal{M}$. Since the points in $\mathcal{M}$ are sufficiently well separated,

if $\mathrm{Err}(\mathcal{A}, \mathcal{H}_n^\epsilon) \leq \epsilon$, we can unambiguously distinguish which $H \in \mathcal{M}$ we are given. On the other hand, if $\mathcal{M}$ is large then learning this information means that the oracle outputs must have large mutual information with the identity of H (via Fano's inequality [CT91]; indeed our strategy is also known as Fano's method). Finally, in the vicinity of the ground state the output distributions produced by zeroth-order queries to $\mathcal{O}_H$ and $\mathcal{O}_{H'}$ for any $H, H' \in \mathcal{M}$ have small relative entropy, which implies an upper bound on the amount of mutual information obtained by each oracle query. Putting this together yields a lower bound on the number of queries needed to optimize $\mathcal{H}_n^\epsilon$ with error ϵ.

Theorem 1 gives a lower bound for *zeroth-order* variational algorithms *restricted to the vicinity* of the optimum. Upon lifting these two restrictions, we obtain a more general lower bound following a similar proof strategy. The primary difference is that now, for this unrestricted case, the oracle output distributions associated with two different $H, H' \in \mathcal{M}$ may be more distinguishable, yielding a weaker lower bound.

Theorem 2 (General lower bound)**.** *For any $n > 15$, $\epsilon < 0.01n$, and $k \in \mathbb{Z}_+$, suppose $\mathcal{A}$ is a k^{th}-order algorithm that makes T queries and satisfies* $\mathrm{Err}(\mathcal{A}, \mathcal{H}_n^\epsilon) \leq \epsilon$*. Then* $T \geq \Omega(\frac{n^2}{\epsilon})$.

2.5 Upper bounds

The arguments above indicate that zeroth-order measurements taken in the vicinity of the optimum may be less informative in some sense than more general measurements. *A priori*, it is unclear if this observation translates into an algorithmic advantage for variational algorithms making gradient measurements. To this end, we show that a first-order algorithm based on SGD can attain an upper bound which matches the lower bound of Theorem 2, even when restricted to the vicinity of an optimum. Hence, not only does this show that a first-order algorithm can converge faster than any zeroth-order algorithm in the vicinity of the optimum, but it also shows that for the specific problem under consideration, the first-order SGD-based algorithm is in fact essentially optimal among all k^{th}-order algorithms for any k. This result is stated as the following theorem.

Theorem 3 (Upper bound for first-order methods)**.** *For any $\epsilon < 0.01n$, there exists a first-order, 100ϵ-vicinity algorithm $\mathcal{A}$ based on SGD that makes $O(\frac{n^2}{\epsilon})$ queries and achieves an error* $\mathrm{Err}(\mathcal{A}, \mathcal{H}_n^\epsilon) \leq \epsilon$.

En route to showing this theorem, we first obtain general upper bounds on the query cost of variational algorithms in the vicinity of a local minimum, reported in Table 2.1. More precisely, the bounds are applicable when the induced objective function f is known to be convex within some fixed convex feasible set. They are obtained by combining objective function or gradient estimators with known convergence results [Bub15] from the theory of stochastic optimization. In particular, the SGD bounds utilize the estimator $\hat{\mathbf{g}}(\boldsymbol{\theta})$ defined previously (note that $\hat{\mathbf{g}}(\boldsymbol{\theta})$ can be constructed from a single first-order oracle query). While Theorem 10 will only require

Convexity of $f(\boldsymbol{\theta})$	Zeroth-order	SGD	SMD
Convex	$\min\left(\frac{p^{7.5}E^2}{\epsilon^2}, \frac{p^2E^4(R_2/r_2)^2}{\epsilon^4}\right)$	$\frac{R_2^2\|\vec{\Gamma}\|_1^2}{\epsilon^2}$	$\frac{R_1^2\|\vec{\Gamma}\|_2^2}{\epsilon^2}$
λ_2-strongly convex w.r.t. $\|\cdot\|_2$	$\min\left(\frac{p^{7.5}E^2}{\epsilon^2}, \frac{p^2E^4(R_2/r_2)^2}{\epsilon^4}\right)$	$\frac{\|\vec{\Gamma}\|_1^2}{\lambda_2\epsilon}$	$\frac{p\|\vec{\Gamma}\|_2^2}{\lambda_2\epsilon}$
λ_1-strongly convex w.r.t. $\|\cdot\|_1$	$\min\left(\frac{p^{7.5}E^2}{\epsilon^2}, \frac{p^2E^4(R_2/r_2)^2}{\epsilon^4}\right)$	$\frac{\|\vec{\Gamma}\|_1^2}{\lambda_1\epsilon}$	$\frac{\|\vec{\Gamma}\|_2^2}{\lambda_1\epsilon}$

Table 2.1: Rigorous upper bounds for the query complexity of optimizing $f(\boldsymbol{\theta})$ to precision ϵ in a convex region $\mathcal{X} \subset \mathbb{R}^p$ contained in a 2-ball of radius R_2, contained in an 1-ball of radius R_1, and containing a 2-ball of radius r_2, using zeroth-order strategies or gradient measurements in conjunction with SGD or SMD with an l_1 setup. Constants, logarithmic factors, and some Lipschitz constants are hidden for clarity. For the family $\mathcal{H}_n^\epsilon$, and with respect to the variational ansatz used in our proof of Theorem 10, we have $p = n$, $E = \Theta(n)$, $\Gamma_i = \Theta(1)$, $R_2 = \Theta(\sqrt{\epsilon})$, $R_1 = \Theta(\sqrt{\epsilon n})$, $r_2 = O(\sqrt{\epsilon/n})$, $\lambda_2 = \Theta(1)$, and $\lambda_1 = \Theta(1/n)$.

an SGD bound, we also report bounds based on stochastic mirror descent (SMD), as well as (for comparison) zeroth-order algorithms. The zeroth-order bounds, based on [FKM05; Bel+15], are the best rigorous bounds we are aware of, but may likely be outperformed in practice. SMD [Bub15] is a non-Euclidean generalization of SGD; the SMD bounds we report are based on taking the norm in parameter space to be the 1-norm rather than the Euclidean 2-norm, as is the case for SGD. Further background on these algorithms, motivation for considering SMD, and full derivation of the bounds in Table 2.1 may be found in Section 3.3.

We now describe an algorithm $\mathcal{A}$ attaining the upper bound in Theorem 10. We refer the reader to Section 3.4 for full technical details of the argument. Start by fixing the following n-parameter variational ansatz Θ:

$$|\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_n\rangle := \exp\left(-i \sum_{j=1}^{n} (\boldsymbol{\theta}_j + \pi/4) Y_j/2\right) |0\rangle^{\otimes n}.$$

This parameterization has a simple geometric interpretation: $|\boldsymbol{\theta}\rangle$ is the product state on n qubits for which the polarization of qubit j is $\sin(\pi/4 + \boldsymbol{\theta}_j)\hat{x} + \cos(\pi/4 + \boldsymbol{\theta}_j)\hat{z}$.

Now, consider some objective observable $H_v^\delta \in \mathcal{H}_n^\epsilon$. The induced objective function $f(\boldsymbol{\theta})$ is found to be $f(\boldsymbol{\theta}) = \langle\boldsymbol{\theta}|H_v^\delta|\boldsymbol{\theta}\rangle = -\sum_{i=1}^n \cos(\boldsymbol{\theta}_i - \delta v_i)$. Let $\mathcal{B}_\infty(\delta) \subset \mathbb{R}^n$ denote the ∞-ball of radius δ centered at the origin. That is, $\mathcal{B}_\infty(\delta) = \{\boldsymbol{\theta} : \max(\boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \ldots, \boldsymbol{\theta}_n) \leq \delta\}$. Note that the ground state of H_v^δ is the state $|\delta v_1, \delta v_2, \ldots, \delta v_n\rangle$, and hence corresponds to a parameter inside the set $\mathcal{B}_\infty(\delta)$ for any choice of v. Furthermore, the set of states associated with $\mathcal{B}_\infty(\delta)$ is contained in the 100ϵ-optimum of $\mathcal{H}_n^\epsilon$, and the induced objective function $f(\boldsymbol{\theta})$ is 0.01-strongly convex w.r.t. the 2-norm (strong convexity is reviewed in Chapter 3). It is also straightforward to show that, for this problem, $\|\vec{\Gamma}\|_1 = O(n)$. Theorem 10 now follows from the SGD upper

bound for strongly convex functions in Table 2.1, taking $\mathcal{B}_\infty(\delta)$ as the feasible set. We note that SMD with a 1-norm setup achieves an identical performance for this toy problem, up to logarithmic factors.

2.6 Conclusion

Our results provide theoretical evidence that taking analytic gradient measurements in variational algorithms can be advantageous, supporting recent gradient-based proposals. We expect the rigorous upper bounds we report in Table 2.1 may be helpful in guiding expectations on the performance of gradient-based variational algorithms for particular classes of problems, even if more heuristic algorithms may be used in practice. To this end, an interesting direction for future work is to understand how the parameters appearing in Table 2.1 behave for various problems of practical interest. Further discussion, open questions, and comparison with the literature may be found in Section 3.5.

Chapter 3

Analytical Gradient Measurements Can Accelerate VQAs, II: Details and Derivations

3.1 Technical preliminaries

3.1.1 Conventions, assumptions, and notation

We will assume throughout that variational states are parameterized according to an ansatz of the form

$$|\boldsymbol{\theta}\rangle := e^{-iA_p\boldsymbol{\theta}_p/2}\cdots e^{-iA_1\boldsymbol{\theta}_1/2}\,|\Psi\rangle\,,$$

where $|\Psi\rangle$ is assumed to be some easy-to-prepare starting state, $\boldsymbol{\theta} := (\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_p)^\top \in \mathcal{X} \subset \mathbb{R}^p$, and A_i are Hermitian operators. We will refer to $\mathcal{X}$ as the *feasible set*. We refer to an individual factor $e^{-iA_j\boldsymbol{\theta}_j/2}$ in the ansatz as a *pulse*, and an A_i as a *pulse generator*. We will occasionally need to refer to a specific variational parameterization, often labeled by the character Θ. When we refer to a parameterization Θ, we assume that Θ collects information about the starting state $|\Psi\rangle$ and the pulse generators A_i.

Given a parameterization Θ and a feasible set $\mathcal{X}$, the classical objective function to be minimized is induced by some Hermitian operator H, which we refer to as the *objective observable*. In particular, the objective function is given by

$$f(\boldsymbol{\theta}) = \langle\boldsymbol{\theta}|H|\boldsymbol{\theta}\rangle\,,\ \boldsymbol{\theta} \in \mathcal{X}.$$

In the context of variational algorithms, it is assumed that the quantum device is capable of measuring some subset of quantum observables. We assume that the set of observables which may be measured is the set of all tensor products of Pauli operators.

When we refer to a qubit with polarization $\vec{r}$, we mean the state specified by the density matrix $\frac{1}{2}(I + \vec{r}\cdot\vec{\sigma})$, where $\vec{\sigma} := (X, Y, Z)^\top$ is the vector of Pauli operators. For some vector $\mathbf{x} \in \mathbb{R}^p$, $\mathbf{x}_i$ denotes the i^{th} component of $\mathbf{x}$. Vectors should be considered column vectors by default. Logarithms are assumed to be base 2 unless otherwise

Notation	Meaning
H	Objective observable.
$\lambda_{\min}(H)$	Smallest eigenvalue of H.
Θ	State parameterization of form $e^{-iA_p\theta_p/2}\cdots e^{-iA_1\theta_1/2}\|\Psi\rangle$.
p	Number of parameters/pulses. Dimension of the optimization problem.
A_j	Generator of pulse j.
$\|\Psi\rangle$	Starting state.
$\boldsymbol{\theta}\in\mathbb{R}^p$	Parameter.
$\|\boldsymbol{\theta}\rangle$	State corresponding to parameter $\boldsymbol{\theta}$ (and implicit parameterization Θ).
$\mathcal{X}\subset\mathbb{R}^p$	Feasible set.
R_1, R_2	Smallest radius of a 1-ball or Euclidean ball, respectively, containing $\mathcal{X}$.
r_2	Largest radius of a Euclidean ball contained in $\mathcal{X}$.
$f(\boldsymbol{\theta})$	Induced objective function. Equal to $\langle\boldsymbol{\theta}\|H\|\boldsymbol{\theta}\rangle$.
α_i, m, P_i	$H=\sum_{i=1}^m \alpha_i P_i$ where the r.h.s. is the Pauli decomposition of H, and $\alpha_i>0$.
$\beta_k^{(j)}$, n_j, $Q_k^{(j)}$	$A_j=\sum_{k=1}^{n_j}\beta_k^{(j)}Q_k^{(j)}$ where the r.h.s. is the Pauli decomposition of A_j, and $\beta_k^{(j)}>0$.
E	$\sum_{i=1}^m \alpha_i$. Upper bounds the operator norm of H.
$\gamma_{kl}^{(j)}$	0 or $\beta_k^{(j)}\alpha_l$ (see Section 3.2.3).
Γ_j	$\sum_{k=1}^{n_j}\sum_{l=1}^m \gamma_{kl}^{(j)}$.
$\vec{\Gamma}$	$(\Gamma_1,\Gamma_2,\ldots,\Gamma_p)^\top$.
λ_1,λ_2	Strong convexity parameter w.r.t. 1-norm or 2-norm, respectively.
$\boldsymbol{\theta}^*$	Minimizer of $f(\boldsymbol{\theta})$ on the feasible set.
$\vec{r}_i$	Polarization (Bloch vector) of the reduced state on qubit i.

Table 3.1: Notation and parameters.

specified. The notation $[p]$ for $p\in\mathbb{Z}_+$ denotes the set $\{1,2,\ldots,p\}$. The q-norm of a vector $\mathbf{x}\in\mathbb{R}^p$ for $q\geq 1$ is defined as $\|\mathbf{x}\|_q := (|\mathbf{x}_1|^q+\cdots+|\mathbf{x}_p|^q)^{1/q}$. The ∞-norm is defined as $\|\mathbf{x}\|_\infty := \max\{|\mathbf{x}_1|,\ldots,|\mathbf{x}_p|\}$. If $\|\cdot\|$ is an arbitrary norm, the dual norm $\|\cdot\|_*$ is defined as $\|\mathbf{g}\|_* := \sup_{\mathbf{x}:\,\|\mathbf{x}\|=1}\mathbf{g}^\top\mathbf{x}$. The notation $\widetilde{O}(\cdot)$ hides polylogarithmic factors. The notation $\mathbb{E}$ denotes an expectation value. $\hat{e}_j$ denotes the unit vector along coordinate j. We let $\lambda_{\min}(H)$ denote the smallest eigenvalue of Hermitian matrix H.

We collect notation and parameters in Table 3.1.

3.1.2 Requisite results about stochastic convex optimization

We will obtain upper bounds for variational algorithms in convex regions by combining well known classical convergence results with sampling strategies for estimating the gradient. Here, we record the classical optimization results we will need. Background on stochastic gradient descent and stochastic mirror descent may be found in Appendix A (see e.g. [Nem+09; JN11; Bub15] for more thorough reviews). First, we define Lipschitz continuity and strong convexity.

Definition 1 (Lipschitz continuity)**.** *For $L>0$, the real-valued function f is L-Lipschitz with respect to norm $\|\cdot\|$ on some convex domain $\mathcal{X}$ if $\forall\mathbf{x},\mathbf{y}\in\mathcal{X}$,*

$$|f(\mathbf{x})-f(\mathbf{y})|\leq L\|\mathbf{x}-\mathbf{y}\|.$$

Definition 2 (Strong convexity)**.** *For $\lambda > 0$, the real-valued function f is λ-strongly convex with respect to norm $\|\cdot\|$ on some convex domain $\mathcal{X}$ if $\forall \mathbf{x}, \mathbf{y} \in \mathcal{X}$,*

$$f(\mathbf{y}) \geq f(\mathbf{x}) + \nabla f(\mathbf{x})^\top (\mathbf{y} - \mathbf{x}) + \frac{\lambda}{2}\|\mathbf{x} - \mathbf{y}\|^2.$$

Note that a twice-differentiable function is λ-strongly convex with respect to the 2-norm if all of the eigenvalues of the Hessian matrix at each point in the domain are at least λ. More generally, f is λ-strongly convex at $\mathbf{x}$ w.r.t. an arbitrary norm $\|\cdot\|$ if $\mathbf{h}^\top \nabla^2 f(\mathbf{x}) \mathbf{h} \geq \lambda \|\mathbf{h}\|^2$ for all $\mathbf{h}$, where $\nabla^2 f(\mathbf{x})$ is the Hessian of f at $\mathbf{x}$. In contrast, f is convex if the Hessians are merely positive semidefinite. Intuitively, if f is strongly convex, then it is lower bounded by a quadratic function.

3.1.2.1 Upper bounds for stochastic first-order optimization

In this section, we record known upper bounds for optimizing convex functions given access to noisy, unbiased gradient information. For the first two results below, we follow the presentation of the review on algorithms for convex optimization [Bub15].

Assume we have access to a stochastic gradient oracle, which upon input of $\mathbf{x} \in \mathcal{X}$, returns a random vector $\hat{\mathbf{g}}(\mathbf{x})$ such that $\mathbb{E}\,\hat{\mathbf{g}}(\mathbf{x}) = \nabla f(\mathbf{x})$, $\mathbb{E}\,\|\hat{\mathbf{g}}(\mathbf{x})\|_2^2 \leq G_2^2$, and $\mathbb{E}\,\|\hat{\mathbf{g}}(\mathbf{x})\|_\infty^2 \leq G_\infty^2$. Assume $\mathcal{X} \subset \mathbb{R}^p$ is a closed convex set. Let $\mathbf{x}^*$ denote a minimizer of f on $\mathcal{X}$.

Theorem 4 (SGD)**.** *Assume $\mathcal{X}$ is contained in a Euclidean ball of radius R_2 and f is convex on $\mathcal{X}$. Then projected SGD with fixed step size $\eta = \frac{R_2}{G_2}\sqrt{\frac{2}{T}}$ satisfies*

$$\mathbb{E}\, f\left(\frac{1}{T}\sum_{s=1}^{T} \mathbf{x}_s\right) - f(\mathbf{x}^*) \leq R_2 G_2 \sqrt{\frac{2}{T}}.$$

where $\mathbf{x}_1$ is the starting point, and the algorithm visits points $\mathbf{x}_1, \ldots, \mathbf{x}_T$.

Theorem 5 (SGD for strongly convex functions)**.** *Assume f is λ_2-strongly convex on $\mathcal{X}$ with respect to $\|\cdot\|_2$. Then SGD with step size $\eta_s = \frac{2}{\lambda_2(s+1)}$ at iteration s satisfies*

$$\mathbb{E}\, f\left(\sum_{s=1}^{T} \frac{2s}{T(T+1)} \mathbf{x}_s\right) - f(\mathbf{x}^*) \leq \frac{2G_2^2}{\lambda_2(T+1)}.$$

where $\mathbf{x}_1$ is the starting point, and the algorithm visits points $\mathbf{x}_1, \ldots, \mathbf{x}_T$.

Theorem 6 (SMD with l_1 setup [Nem+09])**.** *Assume $\mathcal{X}$ is contained in a 1-ball of radius R_1, and f is convex on $\mathcal{X}$. Then stochastic mirror descent with an appropriate l_1 setup and and step size $\eta = \frac{R_1}{G_\infty}\sqrt{\frac{2}{T}}$, satisfies*

$$\mathbb{E}\, f\left(\frac{1}{T}\sum_{s=1}^{T} \mathbf{x}_s\right) - f(\mathbf{x}^*) \leq R_1 G_\infty \sqrt{\frac{2e \ln p}{T}}.$$

where $\mathbf{x}_1$ *is the starting point, and the algorithm queries points* $\mathbf{x}_1, \ldots, \mathbf{x}_T$.

Theorem 7 (SMD with l_1 setup for strongly convex functions [HK14])**.** *Assume* f *is* λ_1*-strongly convex on* $\mathcal{X}$ *with respect to norm* $\|\cdot\|_1$*. Then a variant of SMD [HK14] running for* T *iterations outputs a (random) vector* $\bar{\mathbf{x}}$ *such that*

$$\mathbb{E}\, f(\bar{\mathbf{x}}) - f(\mathbf{x}^*) \leq \frac{16 G_\infty^2}{\lambda_1 T}.$$

3.1.2.2 Upper bounds for stochastic zeroth-order (derivative-free) optimization

Compared to stochastic first-order optimization, less is known about rigorous upper bounds for stochastic zeroth-order optimization. The bounds we use in this paper are from [FKM05] and [Bel+15], which give algorithms for which the expected error in objective function value converges to zero like $\sqrt[4]{\frac{p^2}{T}}$ and $\sqrt{\frac{p^{7.5}}{T}}$, respectively, where T is the number of iterations. These are the strongest stochastic zeroth-order convergence guarantees known to the authors which are applicable to the settings considered in this work. We record their results below, adapted for our purposes.

Theorem 8 (Adapted [FKM05] and [Bel+15])**.** *Let the convex feasible set* $\mathcal{X} \subset \mathbb{R}^p$ *be contained in a Euclidean ball of radius* R_2*, and contain a Euclidean ball of radius* r_2*. Assume access to a stochastic oracle which, upon input* $\mathbf{x} \in \mathcal{X}$*, outputs a real-valued random variable* $\hat{k}(\mathbf{x})$ *such that* $\mathbb{E}\, \hat{k}(\mathbf{x}) = f(\mathbf{x})$ *and* $|\hat{k}(\mathbf{x})| \leq E$*. If* f *is convex and* L*-Lipschitz w.r.t.* $\|\cdot\|_2$*, there exists an algorithm [FKM05] that makes* T *queries and outputs a (random) vector* $\bar{\mathbf{x}}$ *such that*

$$\mathbb{E}\, f(\bar{\mathbf{x}}) - f(\mathbf{x}^*) \leq O\left(\frac{p^2 E^2 R_2^2 (L + E/r_2)^2}{T}\right)^{1/4}.$$

There also exists an algorithm [Bel+15] that makes T *queries and, with high probability, outputs* $\bar{\mathbf{x}}$ *such that*

$$\mathbb{E}\, f(\bar{\mathbf{x}}) - f(\mathbf{x}^*) \leq \widetilde{O}\left(\frac{p^{7.5} E^2}{T}\right)^{1/2}.$$

Note that this implies that $\min\left(O\left(\frac{p^2 E^2 R_2^2 (L+E/r_2)^2}{\epsilon^4}\right), \widetilde{O}(\frac{p^{7.5} E^2}{\epsilon^2})\right)$ queries are needed to optimize to expected precision ϵ.

3.2 Black-box formulation

In this section, we more precisely define our black-box formulation of variational algorithms. Recall that, in the black-box setting, for a given objective observable H the classical algorithm is allowed access to a "sampling oracle" $\mathcal{O}_H$ which, upon being given a description of a variational ansatz and a list of indices $s_1, s_2, \ldots, s_k$,

prepares the appropriate quantum state and outputs an unbiased estimator for the k^{th}-order derivative $\frac{\partial^k f}{\partial \boldsymbol{\theta}_{s_1} \cdots \partial \boldsymbol{\theta}_{s_k}}$. In the zeroth-order case, the output is an unbiased estimator for $f(\boldsymbol{\theta})$. The estimator is constructed by expanding $\partial_{s_1} \cdots \partial_{s_k} f$ into a linear combination of terms, each of which can be measured with a low-depth circuit, and sampling one of the terms in the expansion to measure with weight proportional to that term's coefficient. We make this more precise below.

Definition 3 (Sampling oracle)**.** *Let H be an objective observable. The the sampling oracle $\mathcal{O}_H$ encoding H is defined as follows. It receives as input a description of a p-parameter parameterization Θ, a parameter $\boldsymbol{\theta} \in \mathbb{R}^p$, and a multiset $S = \{s_1, s_2, \ldots, s_k\}$ with each $s_i \in [p]$. The oracle then follows the procedure for k^{th}-order sampling defined below, which returns some random variable X such that $\mathbb{E}\, X = f(\boldsymbol{\theta}) := \langle \boldsymbol{\theta}|H|\boldsymbol{\theta}\rangle$ for zeroth-order sampling, or $\mathbb{E}\, X = \frac{\partial^k f}{\partial \boldsymbol{\theta}_{s_1} \cdots \partial \boldsymbol{\theta}_{s_k}}(\boldsymbol{\theta})$ for k^{th}-order sampling if $k \geq 1$.*

If the parameterization Θ is clear from context, we may not explicitly note that Θ is provided as input to the oracle. If we speak of querying the oracle with a state $|\psi\rangle$, we mean querying the oracle with a parameterization and parameter vector that describe the state $|\psi\rangle$. In the remainder of this section, we first define how the oracle behaves for zeroth-order queries. We then briefly review one way of analytically measuring gradients (and higher-order derivatives) with shallow circuits, and then define the behavior of the oracle upon first- and higher-order queries.

3.2.1 Zeroth-order sampling

Let H be some objective observable. Decompose H into a linear combination of m products of Pauli operators as $H = \sum_{i=1}^m \alpha_i P_i$ where $\alpha_i > 0$. (The coefficients may all be assumed to be positive by absorbing the phase into the operator.) Now, defining the normalization factor $E := \sum_i \alpha_i$ and the probability distribution $p_i := \alpha_i / E$, we may write

$$H = E \sum_{i=1}^m p_i P_i = E \mathop{\mathbb{E}}_{i \sim p_i} P_i.$$

By linearity,

$$f(\boldsymbol{\theta}) := \langle \boldsymbol{\theta}|H|\boldsymbol{\theta}\rangle = E \mathop{\mathbb{E}}_{i \sim p_i} \langle \boldsymbol{\theta}|P_i|\boldsymbol{\theta}\rangle .$$

From this expression, it is clear that by sampling index i with probability p_i, measuring P_i with respect to the state $|\boldsymbol{\theta}\rangle$, and then multiplying the outcome by E we obtain an unbiased estimator for $f(\boldsymbol{\theta}) = \langle \boldsymbol{\theta}|H|\boldsymbol{\theta}\rangle$. Furthermore, since the measurement outcome of P_i is either $+1$ or -1, the output of this estimator is $\pm E$-valued. It is also clear that $|f(\boldsymbol{\theta})| \leq E$ for all $\boldsymbol{\theta}$. We define the behavior of the sampling oracle $\mathcal{O}_H$ for zeroth-order sampling to be essentially the above process.

Definition 4 (Zeroth-order query to $\mathcal{O}_H$)**.** *Let $H = E \sum_{i=1}^m p_i P_i$ be a decomposition of an objective observable as above, where $E > 0$ and p_i is a probability distribution. Given as input a parameterization Θ, parameter $\boldsymbol{\theta}$, and empty multiset $S = \varnothing$, $\mathcal{O}_H$*

behaves as follows. It internally prepares $|\boldsymbol{\theta}\rangle$ and measures the observable P_i with probability proportional to p_i. It then multiplies the outcome by E and outputs the resulting $\pm E$-valued estimator.

3.2.2 Review: analytic gradient measurements

In this section, we review how gradients in variational algorithms can be analytically measured in low depth via a Hadamard test circuit, mostly following [LB17; GS17; Rom+18]. Note that in some cases the gradient can be measured with an even simpler circuit, via the so-called "parameter shift rule" [Li+17; Mit+18]. The choice between these methods for performing the analytic gradient measurements is not important for our results. As a sidenote, an alternative approach for measuring the gradient in variational algorithms was proposed in [GAW19], which builds on Jordan's gradient measurement algorithm [Jor05]. This algorithm offers significantly better performance for obtaining precise estimates of the gradient, but also gives a biased estimator and requires significantly more quantum resources, with coherence time requirements increasing with the desired precision. Hence, the setting and results of [GAW19] are incomparable to ours. In particular, since their construction yields a *biased* gradient estimator, one cannot apply convergence guarantees of SGD which require an unbiased estimator, precluding their gradient estimation algorithm from straightforwardly being used in conjunction with the bounds we consider in this paper.

Consider a particular variational ansatz

$$|\boldsymbol{\theta}\rangle = |\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_p\rangle = e^{-iA_p\boldsymbol{\theta}_p/2} \cdots e^{-iA_1\boldsymbol{\theta}_1/2} |\Psi\rangle .$$

For notational convenience, we define $U_i := e^{-iA_i\boldsymbol{\theta}_i/2}$ to be the unitary corresponding to pulse i, and for $i \leq j$ we define $U_{i:j} := e^{-iA_j\boldsymbol{\theta}_j/2} \cdots e^{-iA_i\boldsymbol{\theta}_i/2}$ to be the sequence of pulses from i through j, inclusive. Note that in using this notation we are hiding the dependence on $\boldsymbol{\theta}$ to avoid notational clutter. Recall that the objective function corresponding to objective observable H is given by

$$\begin{aligned} f(\boldsymbol{\theta}) &:= \langle\boldsymbol{\theta}|H|\boldsymbol{\theta}\rangle \\ &:= \langle\Psi|e^{iA_1\boldsymbol{\theta}_1/2} \cdots e^{iA_p\boldsymbol{\theta}_p/2} H e^{-iA_p\boldsymbol{\theta}_p/2} \cdots e^{-iA_1\boldsymbol{\theta}_1/2}|\Psi\rangle \\ &:= \langle\Psi|U_{1:p}^\dagger H U_{1:p}|\Psi\rangle . \end{aligned}$$

It is straightforward to calculate the following relation via the chain rule applied to the above expression:

$$\frac{\partial f}{\partial \boldsymbol{\theta}_j} = -\operatorname{Im} \langle\Psi|U_{1:j}^\dagger A_j U_{(j+1):p}^\dagger H U_{1:p}|\Psi\rangle .$$

We now describe how the above quantity could be measured in a variational algorithm. Denote the Pauli decomposition of A_j as $A_j = \sum_{k=1}^{n_j} \beta_k^{(j)} Q_k^{(j)}$ where $Q_k^{(j)}$ are products of Pauli operators. As in the previous sections, denote the Pauli decomposition of H as $H = \sum_{i=1}^m \alpha_i P_i$. Then by linearity we can rewrite the above derivative

as

$$\frac{\partial f}{\partial \boldsymbol{\theta}_j} = -\sum_{k=1}^{n_j}\sum_{l=1}^{m} \beta_k^{(j)}\alpha_l \,\mathrm{Im}\,\langle\Psi|U_{1:j}^\dagger Q_k^{(j)} U_{(j+1):p}^\dagger P_l U_{1:p}|\Psi\rangle\,.$$

Now, we can obtain an unbiased estimator for $\mathrm{Im}\,\langle\Psi|U_{1:j}^\dagger Q_k^{(j)} U_{(j+1):p}^\dagger P_l U_{1:p}|\Psi\rangle$ via the Hadamard test. In particular, the following procedure may be used for estimating $\mathrm{Im}\,\langle\Psi|U_{1:j}^\dagger Q_k^{(j)} U_{(j+1):p}^\dagger P_l U_{1:p}|\Psi\rangle$.

Hadamard test for estimating $-\mathrm{Im}\,\langle\Psi|U_{1:j}^\dagger Q_k^{(j)} U_{(j+1):p}^\dagger P_l U_{1:p}|\Psi\rangle$

1. Initialize Register A in the qubit state $|+\rangle_A$. Initialize Register B in the state $|\Psi\rangle_B$.
2. Apply $U_{1:j}$ to Register B.
3. Apply a Controlled-$Q_k^{(j)}$ gate to Register B, controlled on Register A.
4. Apply $U_{(j+1):p}$ to Register B.
5. Apply a Controlled-P_l gate to Register B, controlled on Register A.
6. Measure the Pauli Y operator on Register A.

The above procedure yields a ± 1-valued unbiased estimator for $-\mathrm{Im}\,\langle\Psi|U_{1:j}^\dagger Q_k^{(j)} U_{(j+1):p}^\dagger P_l U_{1:p}|\Psi\rangle$, requiring one quantum measurement.

Algorithm 1: Hadamard test for estimating $-\mathrm{Im}\,\langle\Psi|U_{1:j}^\dagger Q_k^{(j)} U_{(j+1):p}^\dagger P_l U_{1:p}|\Psi\rangle$

Hence, one may estimate $\nabla f(\boldsymbol{\theta})$ by expanding the derivatives as above, and then estimating each term of the expansion using Algorithm 1.

We finally describe an equivalent way of understanding analytic gradients. Observe that

$$\begin{aligned}\frac{\partial f}{\partial \boldsymbol{\theta}_j} &= -\,\mathrm{Im}\,\langle\Psi|U_{1:j}^\dagger A_j U_{(j+1):p}^\dagger H U_{1:p}|\Psi\rangle \\ &= \frac{1}{2}\,\langle\boldsymbol{\theta}|i[U_{(j+1):p} A_j U_{(j+1):p}^\dagger, H]|\boldsymbol{\theta}\rangle\,.\end{aligned}$$

Hence, if we define the Hermitian operators

$$G_j := \frac{i}{2}[U_{(j+1):p} A_j U_{(j+1):p}^\dagger, H],$$

and we define $\vec{G} := (G_1, \ldots, G_p)^\top$, then we may write $\nabla f(\boldsymbol{\theta}) = \langle\boldsymbol{\theta}|\vec{G}|\boldsymbol{\theta}\rangle$. An alternative commutator expression for the derivatives was noted in [McC+18].

The case of a constraint $\theta_i = \theta_j$

There are cases in which one may want to impose a constraint that some parameters are always equal. For example, this situation occurs for the "Hamiltonian variational" ansatz proposed in [WHT15]. Hence, one could have a p-pulse ansatz but a smaller number of independent variational parameters. We note that this situation is easily addressed within the framework of this paper. For example, consider the case in which $\boldsymbol{\theta}_i$ is constrained to always equal $\boldsymbol{\theta}_j$, i.e. $\boldsymbol{\theta}_i = \boldsymbol{\theta}_j := \xi$. It is straightforward to show by linearity that $\frac{\partial f}{\partial \xi} = \langle \boldsymbol{\theta} | (G_i + G_j) | \boldsymbol{\theta} \rangle$, where G_i and G_j are defined as above. Hence, $\frac{\partial f}{\partial \xi}$ may be estimated via Algorithm 1 just as in the unconstrained case. For simplicity, we assume that there are no such constraints on the parameters. However, all results in this paper can be easily generalized to work with such constraints via this observation.

3.2.3 First-order sampling

In the previous section, we described how information about the derivatives of the objective function can be extracted in low depth. In this section, we describe a specific estimator of a derivative of the objective function which requires one Pauli measurement. We will use this estimator to define the behavior of the oracle $\mathcal{O}_H$ upon a first-order query.

As in the previous section, denote the Pauli expansion of H as $H = \sum_{i=1}^{m} \alpha_i P_i$ and the Pauli expansion of A_j as $A_j = \sum_{k=1}^{n_j} \beta_k^{(j)} Q_k^{(j)}$, where all α and β coefficients are positive real numbers. Then we may write $\frac{\partial f}{\partial \boldsymbol{\theta}_j}$ as the following expansion:

$$\frac{\partial f}{\partial \boldsymbol{\theta}_j} = \sum_{k=1}^{n_j} \sum_{l=1}^{m} \beta_k^{(j)} \alpha_l \langle \boldsymbol{\theta} | \frac{i}{2} \left[U_{(j+1):p} Q_k^{(j)} U_{(j+1):p}^\dagger, P_l \right] | \boldsymbol{\theta} \rangle .$$

We now rewrite this expansion as a certain expectation value, similarly to what we did in the definition of zeroth-order sampling. First, we observe that some of the commutators in the expansion may trivially be zero, if the operators $U_{(j+1):p} Q_k^{(j)} U_{(j+1):p}^\dagger$ and P_l act nontrivially on disjoint sets of qubits. This will often be the case in the toy model we analyze in Section 3.4. Removing terms that are trivially zero will improve convergence in our optimization algorithms. To this end, we define a new set of coefficients:

$$\gamma_{kl}^{(j)} := \begin{cases} 0, & \text{qubits}\left(U_{(j+1):p} Q_k^{(j)} U_{(j+1):p}^\dagger \right) \cap \text{ qubits}(P_l) = \varnothing \\ \beta_k^{(j)} \alpha_l, & \text{qubits}\left(U_{(j+1):p} Q_k^{(j)} U_{(j+1):p}^\dagger \right) \cap \text{ qubits}(P_l) \neq \varnothing \end{cases}$$

where $\text{qubits}(U_{(j+1):p} Q_k^{(j)} U_{(j+1):p}^\dagger)$ denotes the set of qubits on which $U_{(j+1):p} Q_k^{(j)} U_{(j+1):p}^\dagger$ acts nontrivially, after removing pulses which trivially commute through $Q_k^{(j)}$ and

cancel the corresponding inverse pulse. We define the associated normalization factors

$$\Gamma_j = \sum_{k=1}^{n_j} \sum_{l=1}^{m} \gamma_{kl}^{(j)},$$

and probability distributions $q_{kl}^{(j)} := \frac{1}{\Gamma_j}\gamma_{kl}^{(j)}$ over the indices k and l, where j is considered fixed. Note that we have the bound $\Gamma_j \leq EB_j$ where $B_j := \sum_{k=1}^{n_j} \beta_k^{(j)}$. Equipped with these definitions, we may write

$$\frac{\partial f}{\partial \boldsymbol{\theta}_j} = \Gamma_j \mathop{\mathbb{E}}_{(K,L)\sim q_{KL}^{(j)}} \langle \boldsymbol{\theta} | \frac{i}{2} \Big[U_{(j+1):p} Q_K^{(j)} U_{(j+1):p}^\dagger, P_L \Big] | \boldsymbol{\theta} \rangle .$$

It is straightforward to see that $|\frac{\partial f}{\partial \boldsymbol{\theta}_j}| \leq \Gamma_j$ for all $\boldsymbol{\theta}$. Given the above representation of $\frac{\partial f}{\partial \boldsymbol{\theta}_j}$, it is clear that the following procedure provides an unbiased estimator for $\frac{\partial f}{\partial \boldsymbol{\theta}_j}$ which requires a single measurement.

An unbiased one-measurement estimator for $\frac{\partial f}{\partial \boldsymbol{\theta}_j}$.

1. Sample (K, L) from the distribution $q_{KL}^{(j)}$ as defined above.
2. Use a Hadamard test (Algorithm 1) to obtain a one-measurement unbiased estimate of $\langle \boldsymbol{\theta} | \frac{i}{2} \Big[U_{(j+1):p} Q_K^{(j)} U_{(j+1):p}^\dagger, P_L \Big] | \boldsymbol{\theta} \rangle = -\operatorname{Im} \langle \Psi | U_{1:j}^\dagger Q_K^{(j)} U_{(j+1):p}^\dagger P_L U_{1:p} | \Psi \rangle$.
3. Multiply the resulting number by Γ_j.

The estimator for $\frac{\partial f}{\partial \boldsymbol{\theta}_j}$ described above is $\pm\Gamma_j$-valued.

Algorithm 2: unbiased, one-measurement estimator for $\frac{\partial f}{\partial \boldsymbol{\theta}_j}$.

Motivated by these derivative-estimating procedures, we now define the behavior of the oracle $\mathcal{O}_H$ upon a first-order query.

Definition 5 (First-order query to $\mathcal{O}_H$)**.** *Let H denote an objective observable. Upon input of parameterization Θ, parameter $\boldsymbol{\theta}$, and multiset $S = \{j\}$ for $j \in [p]$, the oracle internally prepares the state $|\boldsymbol{\theta}\rangle$ and runs Algorithm 2 above. It outputs the resulting $\pm\Gamma_j$-valued estimator for $\frac{\partial f}{\partial \boldsymbol{\theta}_j}$.*

3.2.4 Higher-order sampling

The sampling procedure we have described above for obtaining unbiased estimates of derivatives in low depth generalizes to higher-order derivatives. In this section, we outline how the procedure would work. Start by recalling the derivative operators we derived above:

$$G_j := \frac{i}{2}[U_{(j+1):p} A_j U_{(j+1):p}^\dagger, H].$$

To compress notation, we define the Hermitian operators $\tilde{A}_j := U_{(j+1):p} A_j U_{(j+1):p}^\dagger$ so that $G_j = \frac{i}{2}\Big[\tilde{A}_j, H\Big]$ and $\frac{\partial f}{\partial \boldsymbol{\theta}_j} = \langle \boldsymbol{\theta} | G_j | \boldsymbol{\theta} \rangle$. Note that the operator G_j is independent of $\boldsymbol{\theta}_k$ for $k \leq j$. Hence, if we take the partial derivatives of both sides of the above expression with respect to $\boldsymbol{\theta}_k$ with $k \leq j$, we get

$$\frac{\partial^2 f}{\partial \boldsymbol{\theta}_k \partial \boldsymbol{\theta}_j} = \langle \boldsymbol{\theta} | \frac{i}{2}\Big[\tilde{A}_k, G_j\Big] | \boldsymbol{\theta} \rangle \,,$$

where, since G_j is independent of $\boldsymbol{\theta}_k$, this result follows from arguments identical to those we used to derive the expression for G_j. Also, note that from the original definition $\langle \boldsymbol{\theta} | H | \boldsymbol{\theta} \rangle := \langle \Psi | e^{iA_1 \boldsymbol{\theta}_1/2} \cdots e^{iA_p \boldsymbol{\theta}_p/2} H e^{-iA_p \boldsymbol{\theta}_p/2} \cdots e^{-iA_1 \boldsymbol{\theta}_1/2} | \Psi \rangle$, it is clear that $\frac{\partial^2 f}{\partial \boldsymbol{\theta}_k \partial \boldsymbol{\theta}_j} = \frac{\partial^2 f}{\partial \boldsymbol{\theta}_j \partial \boldsymbol{\theta}_k}$. We therefore have, for $k \leq j$,

$$\frac{\partial^2 f}{\partial \boldsymbol{\theta}_k \partial \boldsymbol{\theta}_j} = \frac{\partial^2 f}{\partial \boldsymbol{\theta}_j \partial \boldsymbol{\theta}_k} = -\frac{1}{4} \langle \boldsymbol{\theta} | \Big[\tilde{A}_k, \Big[\tilde{A}_j, H\Big]\Big] | \boldsymbol{\theta} \rangle \,.$$

To see how to estimate this in low depth, note that we have

$$\frac{\partial^2 f}{\partial \boldsymbol{\theta}_k \partial \boldsymbol{\theta}_j} = \frac{1}{2} \operatorname{Re}\Big(- \langle \boldsymbol{\theta} | \tilde{A}_k \tilde{A}_j H | \boldsymbol{\theta} \rangle + \langle \boldsymbol{\theta} | \tilde{A}_k H \tilde{A}_j | \boldsymbol{\theta} \rangle \Big).$$

From the above expression, we see how to generalize the first-order sampling procedure to higher orders. To obtain an unbiased estimate of $\frac{\partial^2 f}{\partial \boldsymbol{\theta}_k \partial \boldsymbol{\theta}_j}$ with a single measurement, first expand A_k, A_j, and H as linear combinations of products of Paulis. In turn, this yields an expansion of $\frac{\partial^2 f}{\partial \boldsymbol{\theta}_k \partial \boldsymbol{\theta}_j}$ as a linear combination of real parts of inner products of states that are acted on with pulses and Paulis. For a one-measurement estimator, randomly choose one of these inner products with probability proportional to the magnitude its coefficient, and then get an unbiased estimate of the inner product by performing a Hadamard test, similarly to what we described for the first-order case. Note that, in the second-order case (or more generally for the even-order case), the Hadamard test will involve an X-basis measurement instead of Y-basis measurement, since a real part is being estimated.

This procedure works in general for k^{th} order derivatives. In particular, the observable corresponding to a k^{th} order derivative will be a nested commutator of depth k.

3.2.5 Query complexity in the black-box formalism

Having defined the oracle $\mathcal{O}_H$, we may now quantify the cost of an algorithm by the number of queries it makes. The general setup for a variational optimization problem in the black-box setting is that the classical "outer loop" is promised that the objective observable H to be minimized belongs to a family $\mathcal{H}$ of observables, and is given black-box access to $\mathcal{O}_H$. Note that, from the perspective of the outer loop, the problem of minimizing the objective function is a purely classical black-box optimization problem since it gives classical input to the oracle and receives classical

output. We now formalize the notion of the "error" associated with some variational algorithm $\mathcal{A}$ for optimizing a family $\mathcal{H}$ of objective observables.

Definition 6. *Let $\mathcal{H}$ denote a set of objective observables, and $\mathcal{A}$ be a (possibly randomized) classical algorithm which has access to a sampling oracle $\mathcal{O}_H$ for some $H \in \mathcal{H}$ and outputs a description of a quantum state $|\psi\rangle$. Then the optimization error of $\mathcal{A}$ with respect to $\mathcal{H}$,* $\mathrm{Err}(\mathcal{A}, \mathcal{H})$*, is defined to be*

$$\mathrm{Err}(\mathcal{A}, \mathcal{H}) := \sup_{\mathcal{O}_H \,:\, H \in \mathcal{H}} \mathop{\mathbb{E}}_{\psi} [\, \langle\psi|H|\psi\rangle - \lambda_{min}(H)]$$

where the expectation is over the possible randomness of the output state $|\psi\rangle$.

In other words, $\mathrm{Err}(\mathcal{A}, \mathcal{H})$ is the worst-case expected error in objective function value that $\mathcal{A}$ makes over all objective observables in the set $\mathcal{H}$. We now make a few more definitions that will be convenient later.

Definition 7 (k^{th}-order algorithm)**.** *We say that a black-box algorithm $\mathcal{A}$ is a k^{th}-order algorithm if it does not make an l^{th}-order oracle query for any $l > k$.*

Definition 8 (δ-vicinity algorithm)**.** *Define the δ-optimum of an observable H to be the set of all states $|\psi\rangle$ such that $\langle\psi|H|\psi\rangle - \lambda_{\min}(H) \leq \delta$. Define the δ-optimum of a set of observables $\mathcal{H}$ to be the union of the δ-optima of each observable in the set. We say a black-box algorithm $\mathcal{A}$ is a δ-vicinity algorithm for $\mathcal{H}$ if it only queries the black box with descriptions of states that are in the δ-optimum of $\mathcal{H}$.*

3.3 General upper bounds for variational algorithms in a convex region

In this section, we give general upper bounds on the query cost of variational algorithms in a region where the objective function is convex. This amounts to applying the known upper bounds for stochastic convex optimization from Section 3.1.2 to the setting in which estimates of the objective function, or derivatives of the objective function, come from the oracle specified in Section 3.2 (which is easy to implement in low depth in practice). Note that the oracle returns estimates of partial derivatives w.r.t. specific components. However, there are multiple ways of using these derivative estimates to construct a gradient estimator. We describe two such estimators. The first is designed to be used with SGD, and the second is designed to be used with SMD with an l_1 setup.

For the remainder of this section, fix some objective observable with Pauli expansion $H = \sum_{i=1}^{m} \alpha_i P_i$ and some parameterization Θ whose pulse generators A_j have Pauli expansions $A_j = \sum_{k=1}^{n_j} \beta_k^{(j)} Q_k^{(j)}$. As in Section 3.2, define $E := \sum_{i=1}^{m} \alpha_i$, $B_j = \sum_{k=1}^{n_j} \beta_k^{(j)}$. Define Γ_j to be the normalization factor associated with coordinate j as defined in Section 3.2 (see also Table 3.1). We collect these Γ_j into a vector as

$$\vec{\Gamma} := (\Gamma_1, \ldots, \Gamma_p)^\top.$$

3.3.1 Gradient estimators from oracle queries

First, we specify some unbiased estimators for the gradient that we will use. The estimator of Algorithm 3 is based on l_1 sampling and designed with the goal in mind of achieving a smaller 2-norm of the estimator and will be used in conjunction with SGD. The estimator of Algorithm 4 is based on l_2 sampling and designed with the goal of achieving a smaller ∞-norm of the estimator and will be used in conjunction with SMD. The latter estimator also requires a mild assumption on the ∞-norm of the objective function. The estimators also differ in their number of samples: the former estimator uses a single sample while the latter could be called a "mini-batch" estimator which uses an asymptotically growing number of samples. We first define the two estimators, and then prove their correctness and bound them in the subsequent lemmas.

An unbiased one-query estimator for $\nabla f(\boldsymbol{\theta})$.

1. Select coordinate j with probability $\frac{\Gamma_j}{\|\vec{\Gamma}\|_1}$.
2. Query $\mathcal{O}_H$ with parameter $\boldsymbol{\theta}$ and coordinate multiset $\{j\}$.
3. Multiply the output of the oracle by $\frac{\|\vec{\Gamma}\|_1}{\Gamma_j}\hat{e}_j$.

The above procedure outputs a vector $\hat{\mathbf{g}}(\boldsymbol{\theta})$ such that $\mathbb{E}\,\hat{\mathbf{g}}(\boldsymbol{\theta}) = \nabla f(\boldsymbol{\theta})$ and $\|\hat{\mathbf{g}}(\boldsymbol{\theta})\| = \|\vec{\Gamma}\|_1$.

Algorithm 3: l_1-sampling estimator for $\nabla f(\boldsymbol{\theta})$.

An unbiased $\widetilde{O}(p)$-query estimator for $\nabla f(\boldsymbol{\theta})$.

For each $j \in [p]$, query $\mathcal{O}_H$ with parameter $\boldsymbol{\theta}$ and coordinate multiset $\{j\}$ N_j times, where $N_j = \left\lceil p\frac{\Gamma_j^2}{\|\vec{\Gamma}\|_2^2}\ln\left(4p^2\frac{\|\vec{\Gamma}\|_\infty^2}{\|\vec{\Gamma}\|_2^2}\right)\right\rceil$. Letting $\hat{G}_j$ denote the average of the N_j oracle outputs corresponding to component j, output $\hat{\mathbf{g}}(\boldsymbol{\theta}) = \sum_{i=1}^p \hat{G}_i\hat{e}_i$.

Assuming $\|\nabla f(\boldsymbol{\theta})\|_\infty \le \frac{\|\vec{\Gamma}\|_2}{\sqrt{2p}}$, the above procedure outputs a vector $\hat{\mathbf{g}}(\boldsymbol{\theta})$ such that $\mathbb{E}\hat{\mathbf{g}}(\boldsymbol{\theta}) = \nabla f(\boldsymbol{\theta})$ and $\mathbb{E}\|\hat{\mathbf{g}}(\boldsymbol{\theta})\|_\infty^2 \le \frac{5\|\vec{\Gamma}\|_2^2}{2p}$, while requiring at most $N = \sum_{j=1}^p N_j \le p\left[1 + \ln\left(4p^2\frac{\|\vec{\Gamma}\|_\infty^2}{\|\vec{\Gamma}\|_2^2}\right)\right]$ samples.

Algorithm 4: l_2-sampling estimator for $\nabla f(\boldsymbol{\theta})$.

Lemma 1 (Correctness of Algorithm 3)**.** *Algorithm 3 outputs a vector $\hat{\mathbf{g}}(\boldsymbol{\theta})$ such that $\mathbb{E}\,\hat{\mathbf{g}}(\boldsymbol{\theta}) = \nabla f(\boldsymbol{\theta})$ and $\|\hat{\mathbf{g}}(\boldsymbol{\theta})\| = \|\vec{\Gamma}\|_1$.*

Proof. Recalling that the output of $\mathcal{O}_H$ upon querying a first-order derivative of the jth component is $\pm\Gamma_j$, it is clear that the vector output by the above procedure will have norm $\|\vec{\Gamma}\|_1$. Now we show that $\mathbb{E}\,\hat{\mathbf{g}}(\boldsymbol{\theta}) = \nabla f(\boldsymbol{\theta})$. Recall that the probability of selecting index j is $\Gamma_j/\|\vec{\Gamma}\|_1$, and conditioned on index j being selected, the expected output of the procedure is $\frac{\|\vec{\Gamma}\|_1}{\Gamma_j}\frac{\partial f}{\partial \boldsymbol{\theta}_j}\hat{e}_j$. It follows that $\mathbb{E}\,\hat{\mathbf{g}}(\boldsymbol{\theta}) = \nabla f(\boldsymbol{\theta})$ as desired. □

Lemma 2 (Correctness of Algorithm 4)**.** *Algorithm 4 outputs a vector* $\hat{\mathbf{g}}(\boldsymbol{\theta})$ *such that* $\mathbb{E}\,\hat{\mathbf{g}}(\boldsymbol{\theta}) = \nabla f(\boldsymbol{\theta})$ *and* $\mathbb{E}\|\hat{\mathbf{g}}(\boldsymbol{\theta})\|_\infty^2 \leq \frac{5\|\vec{\Gamma}\|_2^2}{2p}$.

Proof. Since the output of $\mathcal{O}_H$ upon receiving as input the parameter $\boldsymbol{\theta}$ and derivative multiset $\{j\}$ is a $\pm\Gamma_j$-valued random variable with expectation $\frac{\partial f}{\partial \theta_j}$, we have $\mathbb{E}\hat{\mathbf{g}}(\boldsymbol{\theta}) = \sum_{i=1}^p \mathbb{E}\hat{G}_i\hat{e}_i = \nabla f(\boldsymbol{\theta})$.

We now seek to upper bound $\mathbb{E}\|\hat{\mathbf{g}}\|_\infty^2$. We first turn our attention to the distribution of the random variable $\hat{G}_j$. Since $\hat{G}_j$ is an average of i.i.d. $\pm\Gamma_j$-valued random variables, Hoeffding's inequality implies

$$\Pr\left(|\hat{G}_j - (\nabla f)_j| \geq t\right) \leq 2\exp\left(-\frac{2t^2 N_j}{\Gamma_j^2}\right)$$

for $t \geq 0$. Using this bound with $t = \frac{\|\vec{\Gamma}\|_2}{\sqrt{2p}}$ and recalling $N_j = \left\lceil p\frac{\Gamma_j^2}{\|\vec{\Gamma}\|_2^2}\ln\left(4p^2\frac{\|\vec{\Gamma}\|_\infty^2}{\|\vec{\Gamma}\|_2^2}\right)\right\rceil$ yields

$$\Pr\left(|\hat{G}_j - (\nabla f)_j| \geq \frac{\|\vec{\Gamma}\|_2}{\sqrt{2p}}\right) \leq \frac{\|\vec{\Gamma}\|_2^2}{2p^2\|\vec{\Gamma}\|_\infty^2}.$$

By the union bound, the probability that $|\hat{G}_j - (\nabla f)_j| \geq \frac{\|\vec{\Gamma}\|_2}{\sqrt{2p}}$ for some j is upper bounded by $\frac{\|\vec{\Gamma}\|_2^2}{2p\|\vec{\Gamma}\|_\infty^2}$. If this event occurs, then we only have the trivial upper bound $\|\hat{\mathbf{g}}\|_\infty^2 \leq \|\vec{\Gamma}\|_\infty^2$. Conditioned on this "bad" event not occurring, we have the bound $\|\hat{\mathbf{g}}\|_\infty^2 \leq 2\frac{\|\vec{\Gamma}\|_2^2}{p}$, where we used the assumption that $|(\nabla f)_j| \leq \frac{\|\vec{\Gamma}\|_2}{\sqrt{2p}}$ for all j. It follows that

$$\mathbb{E}\|\hat{\mathbf{g}}\|_\infty^2 \leq 2\frac{\|\vec{\Gamma}\|_2^2}{p} + \|\vec{\Gamma}\|_\infty^2 \cdot \frac{\|\vec{\Gamma}\|_2^2}{2p\|\vec{\Gamma}\|_\infty^2} = \frac{5\|\vec{\Gamma}\|_2^2}{2p}.$$

□

3.3.2 Upper bounds

Fix an objective observable H, parameterization Θ, and a closed, convex feasible set $\mathcal{X}$. Define $f(\boldsymbol{\theta}) \coloneqq \langle\boldsymbol{\theta}|H|\boldsymbol{\theta}\rangle$, and define $\vec{\Gamma}$ as above (see also Table 3.1). Let $\boldsymbol{\theta}^*$ denote a minimizer of $f(\boldsymbol{\theta})$ on $\mathcal{X}$.

Lemma 3 (SGD bound)**.** *If* f *is convex on* $\mathcal{X}$*, and if* $\mathcal{X}$ *is contained in a Euclidean ball of radius* R_2*, then querying* $\mathcal{O}_H$ *with first-order queries and using the outputs to run projected SGD (with appropriate stepsizes) finds a (random) parameter* $\bar{\boldsymbol{\theta}}$ *such that* $\mathbb{E}\,f(\bar{\boldsymbol{\theta}}) - f(\boldsymbol{\theta}^*) \leq \epsilon$ *with* $\frac{2R_2^2\|\vec{\Gamma}\|_1^2}{\epsilon^2}$ *queries.*

Proof. Use the 1-query estimator of Algorithm 3 for $\nabla f(\boldsymbol{\theta})$ in conjunction with Theorem 4. □

Lemma 4 (SGD bound, strongly convex case)**.** *If* f *is* λ_2*-strongly convex on* $\mathcal{X}$ *with respect to* $\|\cdot\|_2$*, then querying* $\mathcal{O}_H$ *with first-order queries and using the outputs to run*

projected SGD (with appropriate stepsizes) finds a parameter $\bar{\boldsymbol{\theta}}$ *such that* $\mathbb{E}\, f(\bar{\boldsymbol{\theta}}) - f(\boldsymbol{\theta}^*) \leq \epsilon$ *with* $\frac{2\|\vec{\Gamma}\|_1^2}{\lambda_2 \epsilon}$ *queries.*

Proof. Use the 1-query estimator of Algorithm 3 for $\nabla f(\boldsymbol{\theta})$ in conjunction with Theorem 5. □

Lemma 5 (SMD bound)**.** *If* f *is convex on* $\mathcal{X}$*, and if* $\mathcal{X}$ *is contained in a 1-ball of radius* R_1*, then querying* $\mathcal{O}_H$ *with first-order queries and using the outputs to run projected SMD with an appropriate* l_1 *setup and stepsizes finds a parameter* $\bar{\boldsymbol{\theta}}$ *such that* $\mathbb{E}\, f(\bar{\boldsymbol{\theta}}) - f(\boldsymbol{\theta}^*) \leq \epsilon$ *with* $\frac{5eR_1^2\|\vec{\Gamma}\|_2^2 \ln p}{\epsilon^2}\left[1 + \ln\left(4p^2 \frac{\|\vec{\Gamma}\|_\infty^2}{\|\vec{\Gamma}\|_2^2}\right)\right] = O\left(\frac{R_1^2\|\vec{\Gamma}\|_2^2 (\ln p)^2}{\epsilon^2}\right)$ *queries.*

Proof. Use the estimator of Algorithm 4 for $\nabla f(\boldsymbol{\theta})$, which outputs a gradient estimate $\hat{\mathbf{g}}$ such that $\mathbb{E}\|\hat{\mathbf{g}}\|_\infty^2 \leq \frac{5\|\vec{\Gamma}\|_2^2}{2p}$ and requires at most $p\left[1 + \ln\left(4p^2 \frac{\|\vec{\Gamma}\|_\infty^2}{\|\vec{\Gamma}\|_2^2}\right)\right]$ queries per gradient estimate. Use these gradient estimates in conjunction with Theorem 6. □

Lemma 6 (SMD bound, strongly convex case)**.** *If* f *is* λ_1*-strongly convex on* $\mathcal{X}$ *with respect to* $\|\cdot\|_1$*, then querying* $\mathcal{O}_H$ *with first-order queries and using the outputs to run projected SMD with an appropriate choice of mirror map and stepsizes can find a parameter* $\bar{\boldsymbol{\theta}}$ *such that* $\mathbb{E}\, f(\bar{\boldsymbol{\theta}}) - f(\boldsymbol{\theta}^*) \leq \epsilon$ *with* $\frac{40\|\vec{\Gamma}\|_2^2}{\epsilon\lambda_1}\left[1 + \ln\left(4p^2 \frac{\|\vec{\Gamma}\|_\infty^2}{\|\vec{\Gamma}\|_2^2}\right)\right] = O\left(\frac{\|\vec{\Gamma}\|_2^2 \ln p}{\epsilon\lambda_1}\right)$ *queries.*

Proof. Use the estimator of Algorithm 4 for $\nabla f(\boldsymbol{\theta})$ in conjunction with Theorem 7. □

For comparison, we also present an upper bound for the case in which we only make zeroth-order queries to $\mathcal{O}_H$.

Lemma 7 (Derivative-free bound)**.** *If* $\mathcal{X}$ *is a closed convex set contained in a Euclidean ball of radius* R_2 *and containing a Euclidean ball of radius* r_2*, and* f *is* L_2*-Lipschitz w.r.t.* $\|\cdot\|_2$ *and convex on* $\mathcal{X}$*, then querying* $\mathcal{O}_H$ *with zeroth-order queries and using the outputs in conjunction with the algorithm of [FKM05] or [Bel+15] finds a parameter* $\bar{\boldsymbol{\theta}}$ *such that, with high probability,* $\mathbb{E}\, f(\bar{\boldsymbol{\theta}}) - f(\boldsymbol{\theta}^*) \leq \epsilon$ *with* $O\left(\frac{p^2E^2R_2^2(L_2+E/r_2)^2}{\epsilon^4}\right)$ *queries or* $\widetilde{O}\left(\frac{p^{7.5}E^2}{\epsilon^2}\right)$ *queries, respectively.*

Proof. Note that the outputs of zeroth-order queries to $\mathcal{O}_H$ have magnitude E, and apply Theorem 8. □

3.3.3 When is SMD superior to SGD?

In our Letter, we gave intuition for why we might hope that using the 1-norm instead of 2-norm and using SMD with an l_1 setup instead of SGD might be beneficial in some cases. In particular, we noted that the 1-norm of a parameter vector $\boldsymbol{\theta}$ has a natural interpretation as the duration of evolution from the starting state $|\Psi\rangle$ to the trial state associated with $\boldsymbol{\theta}$, $|\boldsymbol{\theta}\rangle$. The l_1-distance between two parameter vectors may be interpreted as the amount of time for while the two associated pulse sequences differ.

Comparing the upper bounds from the previous section, we see that where SGD has a factor of $\|\vec{\Gamma}\|_1^2$, SMD with an l_1 setup has instead a factor of $\|\vec{\Gamma}\|_2^2$. Note that $\|\vec{\Gamma}\|_2^2$ is never larger than $\|\vec{\Gamma}\|_1^2$, and in fact can a factor of p smaller. Consider for example the case in which $\Gamma_1 \approx \Gamma_2 \approx \cdots \approx \Gamma_p$, which may be a realistic scenario in practice. In this case, we have $\|\vec{\Gamma}\|_2^2 \approx p\Gamma_1^2$ for the SMD bound, whereas we have $\|\vec{\Gamma}\|_1^2 \approx p^2\Gamma_1^2$ for the SGD bound, which is quadratically worse in the dimension of parameter space.

On the other hand, where the SGD bounds involve a factor of R_2^2, the SMD bounds involve a factor of R_1^2. R_2^2 is never larger than R_1^2, and can be significantly smaller. This could be the case when, for example, the feasible set $\mathcal{X}$ is a Euclidean ball. On the other hand, if $\mathcal{X}$ is a 1-ball, then $R_1 = R_2$, and SMD could potentially achieve substantially better performance than SGD due to the $\|\vec{\Gamma}\|_2^2$ versus $\|\vec{\Gamma}\|_1^2$ discrepancy.

Another consideration is the issue of strong convexity. As is evident from the above bounds, the presence of strong convexity can substantially accelerate the optimization. SGD can take advantage of strong convexity w.r.t. the 2-norm, but SMD in the l_1 setup measures strong convexity w.r.t. the 1-norm, and in fact it is straightforward to show that the strong convexity parameters are related by $\lambda_1 \leq \lambda_2 \leq p\lambda_1$. In the toy problem we analyze in Section 3.4, we have $\lambda_2 = \Theta(1)$ but $\lambda_1 = \Theta(1/n)$ where n is the number of qubits. However, $\|\vec{\Gamma}\|_1^2 = \Theta(n^2)$ while $\|\vec{\Gamma}\|_2^2 = \Theta(n)$, so up to log factors and constants, SGD and SMD achieve the same asymptotic convergence rate for this toy model.

In conclusion, it is not clear from the upper bounds in the previous section or from the toy model we study in Section 3.4 whether SGD or SMD with an l_1 setup would typically achieve better upper bounds in practice. It is an interesting problem for future work to understand whether an l_2 (Euclidean) setup or an l_1 setup is usually more natural for variational algorithms.

3.4 Oracle separation between zeroth-order and first-order optimization strategies for variational algorithms

In this section, we prove a separation between algorithms which make only zeroth-order queries to the sampling oracle, and those which make first-order queries to the sampling oracle, within the vicinity of the global optimum. This separation is proven with respect to a certain simple parameterized family $\mathcal{H}_n^\epsilon$ of objective observables on n qubits. The optima of the observables in $\mathcal{H}_n^\epsilon$ are $O(\epsilon)$ close to each other, in the sense that for any $H, H' \in \mathcal{H}_n^\epsilon$, the ground state of H is an $O(\epsilon)$ optimum of H'. Precisely, we will prove the following.

Theorem 9 (Zeroth-order lower bound)**.** *For any $n > 15$ and $\epsilon < 0.01n$, let $\mathcal{A}$ be any zeroth-order, 100ϵ-vicinity algorithm for the family $\mathcal{H}_n^\epsilon$ that makes T queries to the oracle. Then, if $\mathrm{Err}(\mathcal{A}, \mathcal{H}_n^\epsilon) \leq \epsilon$, it must hold that $T \geq \Omega\left(\frac{n^3}{\epsilon^2}\right)$ where the implicit factor is some fixed constant.*

On the other hand, we prove that this same class of variational problems can be optimized substantially faster if the algorithm makes first-order queries to the oracle, as quantified in the following theorem. In fact, the algorithm that achieves this convergence rate is a simple stochastic gradient descent strategy. Hence, not only is the query complexity much better in this case, but the classical algorithm achieving this query complexity can be implemented efficiently. For comparison, we also obtain a zeroth-order upper bound for this class of problems using the algorithms of [FKM05] and [Aga+11]. Our first-order upper bound is given in the following theorem.

Theorem 10 (First-order upper bound)**.** *For any $\epsilon < 0.01n$, there exists a first-order, 100ϵ-vicinity algorithm $\mathcal{A}$ for the family $\mathcal{H}_n^\epsilon$ that makes $O\Big(\frac{n^2}{\epsilon}\Big)$ queries and achieves an error $\mathrm{Err}(\mathcal{A}, \mathcal{H}_n^\epsilon) \leq \epsilon$. Moreover, $\mathcal{A}$ is a simple stochastic gradient descent algorithm.*

We also prove a very general lower bound for the case in which the algorithm may make k^{th} order queries to the oracle for any k, and is not restricted to any particular domain of states.

Theorem 11 (General lower bound)**.** *For any $n > 15$ and $\epsilon < 0.01n$, suppose $\mathcal{A}$ is an algorithm that makes T queries and satisfies $\mathrm{Err}(\mathcal{A}, \mathcal{H}_n^\epsilon) \leq \epsilon$. Then $T \geq \Omega\Big(\frac{n^2}{\epsilon}\Big)$.*

Since this lower bound is achieved (up to a possible constant factor) by the upper bound of SGD, we see that SGD is essentially optimal among all black-box strategies for optimizing $\mathcal{H}_n^\epsilon$.

3.4.1 Defining $\mathcal{H}_n^\epsilon$

The subset of objective observables we consider are perturbed around a very simple 1-local Hamiltonian.

Definition 9. *Let $\delta \in \mathbb{R}$, and let $v \in \{-1, 1\}^n$. Then we define*

$$H_v^\delta := -\sum_{i=1}^n \Big[\sin\Big(\frac{\pi}{4} + v_i\delta\Big)X_i + \cos\Big(\frac{\pi}{4} + v_i\delta\Big)Z_i\Big].$$

Intuitively, for a fixed small parameter δ, the set of 2^n observables $\{H_v^\delta\}_v$ are perturbed around $H^0 = -\frac{1}{\sqrt{2}}\sum_{i=1}^n (X_i + Z_i)$. The parameter δ characterizes the strength of the perturbation, and the binary vector v encodes the direction of the perturbation. It is straightforward to see that the ground state of H^0 is $|\pi/4\rangle^{\otimes n}$, where we have defined $|\pi/4\rangle := \cos(\pi/8)\,|0\rangle + \sin(\pi/8)\,|1\rangle$. Geometrically, the state $|\pi/4\rangle$ corresponds to the pure qubit state with polarization $\frac{1}{\sqrt{2}}(\hat{x} + \hat{z})$. In the remainder of this section, we record some facts about these Hamiltonians, and define some quantities.

First, note that we may write $H_v^\delta = -\sum_{i=1}^n \hat{n}^{v_i\delta}\cdot\vec{\sigma}_i$ where $\hat{n}^{v_i\delta} = \big(\sin\big(\frac{\pi}{4} + v_i\delta\big), 0, \cos\big(\frac{\pi}{4} + v_i\delta\big)\big)$ and $\vec{\sigma}_i$ is the vector of Pauli operators acting on qubit i. We may now read off

$\lambda_{\min}(H_v^\delta) = -n$, and the associated eigenvector is

$$\left|\psi_v^\delta\right\rangle = \otimes_i \left[\cos\left(\frac{\pi}{8} + \frac{v_i\delta}{2}\right)|0\rangle_i + \sin\left(\frac{\pi}{8} + \frac{v_i\delta}{2}\right)|1\rangle_i\right].$$

Next, we calculate the expectation value of H_v^δ with respect to any quantum state on n qubits.

Lemma 8. *Suppose ρ is a quantum state such that the polarization of ρ_i, the reduced state of ρ on qubit i, is $\vec{r}_i$. Then* $\operatorname{tr}\left[H_v^\delta \rho\right] = -\sum_{i=1}^n \vec{r}_i \cdot \hat{n}^{\delta v_i}$.

Proof. We have

$$\begin{aligned}
\operatorname{tr}\left[H_v^\delta \rho\right] &= -\sum_{i=1}^n \operatorname{tr}\left[(\hat{n}^{v_i\delta} \cdot \vec{\sigma}_i)\rho_i\right] \\
&= -\frac{1}{2}\sum_{i=1}^n \operatorname{tr}\left[(\hat{n}^{v_i\delta} \cdot \vec{\sigma}_i)(I + \vec{r}_i \cdot \vec{\sigma}_i)\right] \\
&= -\frac{1}{2}\sum_{i=1}^n \operatorname{tr}\left[(\hat{n}^{v_i\delta} \cdot \vec{\sigma}_i)(\vec{r}_i \cdot \vec{\sigma}_i)\right] \\
&= -\frac{1}{2}\sum_{i=1}^n \operatorname{tr}\left[(\hat{n}^{v_i\delta} \cdot \vec{r}_i)I\right] \\
&= -\sum_{i=1}^n \hat{n}^{v_i\delta} \cdot \vec{r}_i.
\end{aligned}$$

□

Finally, we define the set $\mathcal{H}_n^\epsilon$ which we will prove the separation with respect to. To do so, we first define a bias parameter $\delta(\epsilon)$ associated with the precision parameter ϵ.

Definition 10. *For a given "precision parameter" ϵ, define the associated "bias parameter"*

$$\delta(\epsilon) := \sqrt{\frac{45\epsilon}{n}}.$$

Now, we define $\mathcal{H}_n^\epsilon$ to be the set of such observables with bias parameter $\delta(\epsilon)$.

Definition 11. $\mathcal{H}_n^\epsilon := \{H_v^{\delta(\epsilon)} : \forall v \in \{-1, 1\}^n\}$.

For the remainder of the paper, we often hide the dependence of δ on ϵ for notational simplicity, and simply write δ where we implicitly mean $\delta(\epsilon)$. Note that our constraint $\epsilon \leq 0.01n$ implies $\delta < 0.7$.

3.4.2 Proof of Theorem 9: zeroth-order lower bound for $\mathcal{H}_n^\epsilon$ in the vicinity of the optimum

In this section, we prove Theorem 9. Our proof strategy for the lower bound is to reduce a statistical learning problem to the optimization problem, and then lower bound the number of oracle calls required to solve the learning problem. Precisely, we will take an appropriate subset $\mathcal{M}_n^\epsilon \subset \mathcal{H}_n^\epsilon$, parameterized by some subset $\mathcal{V}$ of the n-dimensional hypercube $\{-1,+1\}^n$. That is, we will have $\mathcal{M}_n^\epsilon = \{H_v^{\delta(\epsilon)} : v \in \mathcal{V}\}$ where $\mathcal{V} \subset \{-1,1\}^n$ will be strategically chosen. We prove that, if there exists an algorithm $\mathcal{A}$ that satisfies $\mathrm{Err}(\mathcal{A}, \mathcal{M}_n^\epsilon) \leq \epsilon$, then the same algorithm could be used to identify the hidden parameter $v \in \mathcal{V}$ associated with the objective observable $H_v^\delta \in \mathcal{M}_n^\epsilon$. By employing information theoretic methods, we will lower bound the number of oracle calls required to identify the parameter v, which in turn lower bounds the number of calls required to optimize to precision ϵ.

Our proof in some parts adapts techniques from [Aga+09] and [JNR12], which lower bound the query cost of certain convex first-order and derivative-free optimization problems. These results in turn draw on methods from statistical minimax and learning theory.

3.4.2.1 Choosing a well-separated subset $\mathcal{M}_n^\epsilon \subset \mathcal{H}_n^\epsilon$

We begin by defining, for fixed ϵ, a subset $\mathcal{M}_n^\epsilon \subset \mathcal{H}_n^\epsilon$ of objective observables that are well-separated, in the sense that if a state is close to the optimal of $H_v^\delta \in \mathcal{M}_n^\epsilon$, then it must be far from the optimal of $H_{v'}^\delta \in \mathcal{M}_n^\epsilon$ for any other parameter v'. We make this precise below.

We make use of the following classical fact about packings of the hypercube (see for example [Gun11] for a simple proof).

Lemma 9 (Gilbert-Varshamov bound)**.** *There exists a subset $\mathcal{V}$ of the n-dimensional hypercube $\{-1,1\}^n$ of size $|\mathcal{V}| \geq e^{n/8}$ such that, if $\Delta(v,v')$ denotes the Hamming distance between v and v',*

$$\Delta(v,v') \geq \frac{n}{4}$$

for all $v \neq v'$ with $v, v' \in \mathcal{V}$.

Fix $\mathcal{V}$ to be such a subset of $\{-1,1\}^n$, and define $\mathcal{M}_n^\epsilon := \{H_v^\delta : v \in \mathcal{V}\}$. The Hamming distance provides a natural distance measure between points of the hypercube. We now define a notion of distance d between objective observables H_v^δ and $H_{v'}^\delta$. Intuitively, if $d(v,v')$ is large, then a state that is close to the optimal of H_v^δ cannot be close to the optimal of $H_{v'}^\delta$.

Definition 12. *For $v, v' \in \{-1,1\}^n$, we define the semimetric*

$$d(v,v') := \min_{|\psi\rangle} \big[\big(\langle\psi|H_v^\delta|\psi\rangle - \lambda_{\min}(H_v^\delta)\big) + \big(\langle\psi|H_{v'}^\delta|\psi\rangle - \lambda_{\min}(H_{v'}^\delta)\big)\big]$$

where the minimization is over all normalized pure states on n qubits.

Note that $\lambda_{\min}(H_v^\delta)$ is simply $-n$, but we oftentimes write $\lambda_{\min}(H_v^\delta)$ for clarity. We now define a packing parameter β which quantifies how packed the subset $\mathcal{V}$ is, with respect to the semimetric d.

Definition 13. *The packing parameter β corresponding to the above subset $\mathcal{V} \subset \{-1,1\}^n$ and semimetric d on the hypercube is defined to be*

$$\beta := \min_{v \neq v' \in \mathcal{V}} d(v, v').$$

We now have the following lemma.

Lemma 10. *Suppose that for some state $|\psi\rangle$ and parameter $v \in \mathcal{V}$, $\langle\psi|H_v^\delta|\psi\rangle - \lambda_{\min}(H_v^\delta) \leq \beta/3$. Then for all $v' \neq v$ with $v' \in \mathcal{V}$, $\langle\psi|H_{v'}^\delta|\psi\rangle - \lambda_{\min}(H_{v'}^\delta) > \beta/3$.*

Proof. Suppose there exists some parameter $v' \in \mathcal{V}$, $v' \neq v$ for which $\langle\psi|H_{v'}^\delta|\psi\rangle - \lambda_{\min}(H_{v'}^\delta) \leq \beta/3$. From Definition 12, this implies that $d(v, v') \leq 2\beta/3$, which contradicts the assumption that β is the packing parameter. □

We now show that any algorithm which optimizes the observables in the set $\mathcal{M}_n^\epsilon$ with error ϵ can be used to identify the parameter v with high probability.

Lemma 11. *Suppose that $\mathcal{A}$ is an algorithm such that $\mathrm{Err}(\mathcal{A}, \mathcal{M}_n^\epsilon) \leq \beta/9$. Then, one may use the output of $\mathcal{A}$ to construct an estimator $\hat{v}$ such that, if the objective observable is H_v^δ for $v \in \mathcal{V}$, then $\Pr[\hat{v} = v] \geq 2/3$.*

Proof. By assumption, if the observable that is realized is H_v^δ for $v \in \mathcal{V}$, $\mathcal{A}$ outputs a description ψ of a quantum state $|\psi\rangle$ such that

$$\mathop{\mathbb{E}}_{\psi} \langle\psi|H_v^\delta|\psi\rangle - \lambda_{\min}(H_v^\delta) \leq \beta/9.$$

Markov's inequality therefore implies

$$\Pr_{\psi}[\,\langle\psi|H_v^\delta|\psi\rangle - \lambda_{\min}(H_v^\delta) \leq \beta/3] \geq 2/3.$$

Define the estimator $\hat{v}(\psi) := \mathrm{argmin}_{v' \in \mathcal{V}}\, \langle\psi|H_{v'}^\delta|\psi\rangle - \lambda_{\min}(H_{v'}^\delta) = \mathrm{argmin}_{v' \in \mathcal{V}}\, \langle\psi|H_{v'}^\delta|\psi\rangle$. Lemma 10 implies that, if $\langle\psi|H_v^\delta|\psi\rangle - \lambda_{\min}(H_v^\delta) \leq \beta/3$, this estimator returns $\hat{v} = v$ with probability one. Since this event occurs with probability at least $2/3$, the estimator returns $\hat{v} = v$ with probability at least $2/3$. □

We have shown that the ability to optimize $\mathcal{M}_n^\epsilon$ well implies the ability to identify the hidden parameter $v \in \mathcal{V}$ with high probability. We now compute the packing parameter β for the family $\mathcal{M}_n^\epsilon$.

Lemma 12. *For the subset $\mathcal{V}$, semimetric d, and packing parameter β as defined above,*

$$\beta \geq \frac{n}{2}(1 - \cos(\delta)) \geq \frac{n\delta^2}{5}.$$

Proof. Recall that for all $v, v' \in \{-1, 1\}^n$,

$$\begin{aligned} d(v, v') &= \min_{|\psi\rangle} \big[\big(\langle \psi | H_v^\delta | \psi \rangle - \lambda_{\min}(H_v^\delta) \big) \\ &\qquad + \big(\langle \psi | H_{v'}^\delta | \psi \rangle - \lambda_{\min}(H_{v'}^\delta) \big) \big] \\ &= \min_{|\psi\rangle} \langle \psi | (H_v^\delta + H_{v'}^\delta) | \psi \rangle + 2n, \end{aligned}$$

where the minimization is over all normalized pure states on n qubits. Therefore, to compute $d(v, v')$, it suffices to compute the smallest eigenvalue of $H_v^\delta + H_{v'}^\delta$.

We may write

$$\begin{aligned} H_v^\delta + H_{v'}^\delta &= - \sum_{i : v_i = v_i'} \Big[2 \sin\Big(\frac{\pi}{4} + v_i \delta\Big) X_i + 2 \cos\Big(\frac{\pi}{4} + v_i \delta\Big) Z_i \Big] \\ &\quad - \sum_{i : v_i \neq v_i'} \Big[\sqrt{2} \cos(\delta) X_i + \sqrt{2} \cos(\delta) Z_i \Big] \\ &= -2 \sum_{i : v_i = v_i'} \Big[\sin\Big(\frac{\pi}{4} + v_i \delta\Big) X_i + \cos\Big(\frac{\pi}{4} + v_i \delta\Big) Z_i \Big] \\ &\quad - 2 \cos(\delta) \sum_{i : v_i \neq v_i'} \Big[\frac{1}{\sqrt{2}} X_i + \frac{1}{\sqrt{2}} Z_i \Big] \end{aligned}$$

where we used the trigonometric identities $\sqrt{2}\cos(\delta) = \cos(\pi/4 + \delta) + \cos(\pi/4 - \delta) = \sin(\pi/4 + \delta) + \sin(\pi/4 - \delta)$. From this expression, it is clear that the smallest eigenvalue of $H_v^\delta + H_{v'}^\delta$ is $-2(n - \Delta(v, v')) - 2\cos(\delta)\Delta(v, v')$, from which it follows that $d(v, v') = 2\Delta(v, v')(1 - \cos(\delta))$. By construction, for all $v \neq v'$ with $v, v' \in \mathcal{V}$, we have $\Delta(v, v') \geq n/4$. It follows that $\beta \geq \frac{n}{2}(1 - \cos(\delta))$.

The final inequality follows from the fact that $\cos(\delta) \leq 1 - \frac{2\delta^2}{5}$ for $\delta \leq 0.7$. □

Lemma 13. *Any algorithm $\mathcal{A}$ for which $\mathrm{Err}(\mathcal{A}, \mathcal{M}_n^\epsilon) \leq \epsilon$ can be used to construct an estimator $\hat{v}$ which correctly identifies the parameter v of the realized observable $H_v^\delta \in \mathcal{M}_n^\epsilon$ with probability at least $2/3$.*

Proof. By Lemma 12, the packing parameter is at least $\frac{n\delta^2}{5}$. Then by Lemma 11, if we can optimize observables in the set $\mathcal{M}_n^\epsilon$ with expected error at most $\frac{1}{9}\frac{n\delta^2}{5} = \frac{n\delta^2}{45} = \epsilon$, we can identify v with probability at least $2/3$. □

Our proof will proceed as follows. We restrict to the subset $\mathcal{M}_n^\epsilon \subset \mathcal{H}_n^\epsilon$ and prove a lower bound on the number of zeroth-order, 100ϵ-vicinity queries one must make in order to identify the hidden parameter v associated with the realized objective observable $H_v^\delta \in \mathcal{M}_n^\epsilon$. By Lemma 13, this number also lower bounds the number of such queries an algorithm $\mathcal{A}$ must make to satisfy $\mathrm{Err}(\mathcal{A}, \mathcal{M}_n^\epsilon) \leq \epsilon$. Since $\mathcal{M}_n^\epsilon$ is a subset of $\mathcal{H}_n^\epsilon$, optimizing $\mathcal{M}_n^\epsilon$ is no harder than optimizing $\mathcal{H}_n^\epsilon$, and so this number also lower bounds the number of such queries needed to optimize $\mathcal{H}_n^\epsilon$ to precision ϵ.

We next prove two simple lemmas we will need.

Lemma 14. *Suppose $|\phi\rangle$ is in the μ-optimum of H_v^δ, i.e. $\langle\phi|H_v^\delta|\phi\rangle - \lambda_{\min}(H_v^\delta) \leq \mu$. Let $\vec{r}_i$ be the polarization of the reduced state of $|\phi\rangle$ on qubit i, and let $\alpha_i \in [0,\pi]$ be the (unoriented) angle between the vector $\vec{r}_i$ and the unit vector $\hat{n}^{\delta v_i}$. Then $\frac{1}{n}\sum_{i=1}^n \alpha_i^2 \leq \frac{10\mu}{n}$ and $\frac{1}{n}\sum_{i=1}^n \alpha_i \leq \sqrt{\frac{10\mu}{n}}$.*

Proof. From Lemma 8 and rearranging terms,

$$\frac{1}{n}\sum_{i=1}^n \vec{r}_i \cdot \hat{n}^{\delta v_i} \geq 1 - \frac{\mu}{n}.$$

Now, note that $\vec{r}_i \cdot \hat{n}^{\delta v_i} = |\vec{r}_i|\cos\alpha_i \leq |\vec{r}_i|\left(1 - \frac{\alpha_i^2}{10}\right) \leq 1 - \frac{\alpha_i^2}{10}$ for $\alpha_i \in [0,\pi]$. This gives us

$$\frac{1}{n}\sum_{i=1}^n \alpha_i^2 \leq \frac{10\mu}{n}.$$

It immediately follows from Jensen's inequality that

$$\frac{1}{n}\sum_{i=1}^n \alpha_i \leq \sqrt{\frac{10\mu}{n}}.$$

□

Lemma 15. *Suppose $|\phi\rangle$ is in the $k\epsilon$-optimum of $H_{v'}^\delta$ for some $k > 0$. Then $|\phi\rangle$ is in the $(k + 30\sqrt{2k} + 90)\epsilon$-optimum of H_v^δ for any $v \in \{-1,1\}^n$.*

Proof. As in the previous lemma, let $\vec{r}_i$ denote the polarization of the reduced state on qubit i, and α_i denote the angle between $\vec{r}_i$ and $\hat{n}^{\delta v_i}$. We have

$$\begin{aligned}\langle\phi|H_v^\delta|\phi\rangle - \lambda_{\min}(H_v^\delta) &= \langle\phi|[H_{v'}^\delta + (H_v^\delta - H_{v'}^\delta)]|\phi\rangle - (-n) \\ &\leq k\epsilon + \langle\phi|(H_v^\delta - H_{v'}^\delta)|\phi\rangle.\end{aligned}$$

We now make the observation that

$$H_v^\delta - H_{v'}^\delta = -2\sin(\delta)\sum_{i\,:\,v_i \neq v_i'} v_i \frac{X_i - Z_i}{\sqrt{2}}.$$

From this observation, it follows that

$$\begin{aligned}
\langle\phi|H_v^\delta|\phi\rangle - \lambda_{\min}(H_v^\delta) &\leq k\epsilon + 2\sin(\delta)\sum_{i=1}^{n}\left|\langle\phi|\frac{X_i - Z_i}{\sqrt{2}}|\phi\rangle\right| \\
&= k\epsilon + 2\sin(\delta)\sum_{i=1}^{n}\left|\vec{r}_i \cdot \frac{\hat{x} - \hat{z}}{\sqrt{2}}\right| \\
&\leq k\epsilon + 2\sin(\delta)\sum_{i=1}^{n}[\alpha_i + \delta] \\
&\leq k\epsilon + 2\delta\sum_{i=1}^{n}[\alpha_i + \delta] \\
&\leq k\epsilon + 2\delta\sum_{i=1}^{n}\alpha_i + 2n\delta^2 \\
&\leq k\epsilon + 2\delta\sqrt{10nk\epsilon} + 2n\delta^2 \\
&= (k + 30\sqrt{2k} + 90)\epsilon.
\end{aligned}$$

□

Here, the relation $\left|\vec{r}_i \cdot \frac{\hat{x}-\hat{z}}{\sqrt{2}}\right| \leq \alpha_i + \delta$ can be seen geometrically. We also used the definition $\delta^2 = \frac{45\epsilon}{n}$.

3.4.2.2 Applying Fano's inequality

At this point, it remains to lower bound the number of zeroth-order calls to the oracle needed to correctly identify the unknown bias parameter $v \in \mathcal{V}$. The results from the previous section will then allow us to turn this into a lower bound for optimization. We will need the following well-known variant of Fano's inequality. For this result and other information-theoretic results used in this section, see (for example) [CT91].

Lemma 16 (Fano's inequality)**.** *Suppose the random variable V is uniformly distributed on the discrete set $\mathcal{V}$, and the variable X may be correlated with V. Suppose $\mathcal{A}$ is an algorithm that attempts to identify V given the variable X. Then the probability of error p_e satisfies*

$$p_e \geq 1 - \frac{I(V;X) + 1}{\log|\mathcal{V}|}$$

where $I(V;X)$ is the mutual information between V and X. When we use this inequality in our proof, we will let $\mathcal{V}$ be the set of bias parameters $v \in \mathcal{V}$ associated with $\mathcal{M}_n^\epsilon$ defined in the previous section, and X will be the set of queries to and outputs from the zeroth-order sampling oracle.

First, recall how the oracle behaves for zeroth-order queries. It selects a term in the Pauli expansion of the objective observable with probability proportional to the magnitude of the coefficient of that term. Consider the objective observable H_v^δ. Note that the sum of coefficients of Pauli operators acting on qubit i is $\sin(\pi/4 + v_i\delta) +$

$\cos(\pi/4 + v_i\delta) = \sqrt{2}\cos(\delta)$ where we have used a standard trigonometric identity. Note that this quantity is independent of the parameter v. This means that, when we do a zeroth-order query of the oracle $\mathcal{O}_{H_v^\delta}$ encoding this Hamiltonian, the oracle is equally likely to select X_i or Z_i for measurement as it is X_j or Z_j for some other $j \neq i$. Thus, we may equivalently describe the oracle $\mathcal{O}_{H_v^\delta}$ as operating in the following manner. Note that the below algorithm is simply a specialization of the zeroth-order behavior of the sampling oracle (Definition 4) to the particular objective observable H_v^δ.

Zeroth order behavior of $\mathcal{O}_{H_v^\delta}$

Upon input of a parameterization Θ, parameter $\boldsymbol{\theta}$, and empty coordinate multiset $S = \varnothing$,

1. Select an index $i \in [n]$ uniformly at random.
2. Flip a coin with probability of heads $p = \frac{1}{\sqrt{2}\cos(\delta)}\sin(\pi/4 + v_i\delta) = \frac{1}{2}(1 + v_i\tan(\delta))$.
3. If heads, measure $-X_i$ w.r.t. the state $|\boldsymbol{\theta}\rangle$. If tails, measure $-Z_i$.
4. Multiply the above measurement outcome by $E = \sqrt{2}n\cos(\delta)$ and output the result.

Algorithm 5: zeroth-order behavior of $\mathcal{O}_{H_v^\delta}$.

Let the parameter $v \in \mathcal{V}$ be uniformly distributed, and denote the associated random variable V. Suppose an algorithm makes T zeroth-order queries to the oracle. Let ξ_i be the input to the oracle in query i. Let Y_i denote the output of query i. The algorithm may use information from steps one through i to decide the input ξ_{i+1} to query the oracle with on iteration $i + 1$. Formally, we have the variables $\xi_1, Y_1, \xi_2, \ldots, \xi_T, Y_T$, where ξ_1 (the algorithm's first guess) is independent of V, and ξ_{i+1} is a deterministic or stochastic function of $\xi_1, Y_1, \ldots, \xi_i, Y_i$. We begin with a simple lemma. Note that versions of this relation are well-known (e.g. [Aga+09; RR11; JNR12]).

Lemma 17. $I(V;(\xi_1,Y_1,\ldots,\xi_T,Y_T)) \leq T \max_{\xi_1} I(V;Y_1|\xi_1)$.

Proof.

$$\begin{aligned}
&I(V;(\xi_1,Y_1,\ldots,\xi_T,Y_T)) \\
&\quad= \sum_{i=1}^{T} I(V;\xi_i|\xi_1,Y_1,\ldots,\xi_{i-1},Y_{i-1}) \\
&\qquad\qquad + I(V;Y_i|\xi_1,Y_1,\ldots,\xi_{i-1},Y_{i-1},\xi_i) \\
&\quad= \sum_{i=1}^{T} I(V;Y_i|\xi_1,Y_1,\ldots,\xi_{i-1},Y_{i-1},\xi_i) \\
&\quad= \sum_{i=1}^{T} H(Y_i|\xi_1,Y_1,\ldots,Y_{i-1},\xi_i) \\
&\qquad\qquad - H(Y_i|\xi_1,Y_1,\ldots,Y_{i-1},\xi_i,V) \\
&\quad\leq \sum_{i=1}^{T} (H(Y_i|\xi_i) - H(Y_i|\xi_i,V)) \\
&\quad= \sum_{i=1}^{T} I(V;Y_i|\xi_i) \\
&\quad\leq T \max_{\xi_1} I(V;Y_1|\xi_1)
\end{aligned}$$

where in the first line we have used the chain rule for mutual information, in the second we used the fact that ξ_i depends only on $(\xi_1,Y_1,\ldots,\xi_{i-1},Y_{i-1})$, in the third we used the definition of mutual information, and in the fourth we used subadditivity and the fact that Y_i depends only on ξ_i and V. □

With this inequality in hand, we seek to upper bound $I(V;Y_1|\xi_1)$. To do so, we will write $I(V;Y_1|\xi_1)$ in terms of relative entropies. It will be helpful to introduce some additional notation. Let $\mathbb{P}$ be the distribution of Y_1 conditioned on ξ_1, $\mathbb{P}_v$ be the distribution of $\mathbb{P}$ conditioned on the hidden parameter being v, $\mathbb{P}^j$ be the distribution of $\mathbb{P}$ conditioned on the oracle selecting qubit j for measurement in Step 1 of Algorithm 5, and $\mathbb{P}^j_{\pm 1}$ be the same distribution with the additional conditioning on $v_j = \pm 1$.

Letting $D(\cdot\|\cdot)$ denote the relative entropy of two distributions, we have for any ξ_1,

$$\begin{aligned} I(V;Y_1|\xi_1) &= \frac{1}{|\mathcal{V}|}\sum_{v\in\mathcal{V}} D(\mathbb{P}_v\|\mathbb{P}) \\ &\leq \frac{1}{|\mathcal{V}|^2}\sum_{v,v'\in\mathcal{V}} D(\mathbb{P}_v\|\mathbb{P}_{v'}) \\ &\leq \frac{1}{n|\mathcal{V}|^2}\sum_{v,v'\in\mathcal{V}}\sum_{j=1}^{n} D(\mathbb{P}^j_{v_j}\|\mathbb{P}^j_{v'_j}) \\ &\leq \max_{v,v'\in\mathcal{V}} \frac{1}{n}\sum_{j=1}^{n} D(\mathbb{P}^j_{v_j}\|\mathbb{P}^j_{v'_j}) \end{aligned}$$

where the first line is a well-known expression for the mutual information, and the next two lines follow from convexity of the relative entropy. It remains to upper bound
$\max_{v,v'} \frac{1}{n}\sum_{j=1}^{n} D(\mathbb{P}^j_{v_j}\|\mathbb{P}^j_{v'_j})$.

3.4.2.3 Upper bounding $\max_{v,v'} \frac{1}{n}\sum_{j=1}^{n} D(\mathbb{P}^j_{v_j}\|\mathbb{P}^j_{v'_j})$.

Recall that since $\mathcal{A}$ only queries states in the 100ϵ-optimum of $\mathcal{H}^{\epsilon}_n$, then for any state $|\boldsymbol{\theta}\rangle$ that is queried, $\langle\boldsymbol{\theta}|H^{\delta}_v|\boldsymbol{\theta}\rangle \leq 650\epsilon$ by Lemma 15. As we have done before, let α_i denote the angle between $\vec{r}_i$ and $\hat{n}^{\delta v_i}$. By Lemma 14, we know that $\left(\frac{1}{n}\sum_{i=1}^{n}\alpha_i\right)^2 \leq \frac{1}{n}\sum_{i=1}^{n}\alpha_i^2 \leq \frac{6500\epsilon}{n}$ for any state that is queried.

Continuing on, we now calculate the distribution $\mathbb{P}^j_{v_j}$ in terms of previously defined parameters. Recall that $\mathbb{P}^j_{v_j}$ is a $\pm E$-valued Bernoulli distribution. Letting $\mathbb{P}^j_{v_j}[+E]$ denote the probability of obtaining $+E$, we find

$$\begin{aligned} \mathbb{P}^i_{v_i}[+E] &= \frac{1}{2}(1+v_i\tan(\delta))\Pr[-X_i=+1] \\ &\quad + \frac{1}{2}(1-v_i\tan(\delta))\Pr[-Z_i=+1] \\ &= \frac{1}{2}(1+v_i\tan(\delta))\frac{1}{2}(1-\vec{r}_i\cdot\hat{x}) \\ &\quad + \frac{1}{2}(1-v_i\tan(\delta))\frac{1}{2}(1-\vec{r}_i\cdot\hat{z}) \\ &= \frac{1}{2} - \frac{1}{4}\vec{r}_i\cdot(\hat{x}+\hat{z}) - \frac{v_i}{4}\tan(\delta)\vec{r}_i\cdot(\hat{x}-\hat{z}) \\ &= \frac{1}{2} - \frac{1}{2\sqrt{2}}\vec{r}\cdot\left(\frac{\hat{x}+\hat{z}}{\sqrt{2}} + v_i\tan(\delta)\frac{\hat{x}-\hat{z}}{\sqrt{2}}\right). \end{aligned}$$

Similarly,

$$\mathbb{P}^i_{v_i}[-E] = \frac{1}{2} + \frac{1}{2\sqrt{2}}\vec{r}\cdot\left(\frac{\hat{x}+\hat{z}}{\sqrt{2}} + v_i\tan(\delta)\frac{\hat{x}-\hat{z}}{\sqrt{2}}\right).$$

We are interested in bounding the relative entropy between $\mathbb{P}^j_{+1}$ and $\mathbb{P}^j_{-1}$. To this end, we first prove the following elementary lemma.

Lemma 18. *Suppose Q and R are two $\pm E$-valued Bernoulli distributions, with $Q[+E] = q$ and $R[+E] = r$. Then*

$$D(Q\|R) \le \frac{1}{\ln 2}\frac{(q-r)^2}{r(1-r)}.$$

Proof.

$$\begin{aligned} D(Q\|R) &= q\log\left(\frac{q}{r}\right) + (1-q)\log\left(\frac{1-q}{1-r}\right) \\ &\le \frac{q}{\ln 2}\left(\frac{q}{r}-1\right) + \frac{1-q}{\ln 2}\left(\frac{1-q}{1-r}-1\right) \\ &= \frac{1}{\ln 2}\frac{(q-r)^2}{r(1-r)} \end{aligned}$$

where in the second line we used the inequality $\log x \le \frac{1}{\ln 2}(x-1)$ with $x > 0$. □

Note that $\left|\frac{1}{2\sqrt{2}}\vec{r}\cdot\left(\frac{\hat{x}+\hat{z}}{\sqrt{2}} + v_i\tan(\delta)\frac{\hat{x}-\hat{z}}{\sqrt{2}}\right)\right| \le \frac{\sqrt{1+\tan^2(\delta)}}{2\sqrt{2}} \le 0.49$ for $\delta \le 0.7$, which implies $0.01 \le \mathbb{P}^i_{v_i}[+E] \le 0.99$. Further, we have

$$\begin{aligned} &|\mathbb{P}^i_{+1}[E] - \mathbb{P}^i_{-1}[E] \\ &\quad= \left|\left(\frac{1}{2} - \frac{1}{2\sqrt{2}}\vec{r}_i\cdot\frac{\hat{x}+\hat{z}}{\sqrt{2}} + \frac{1}{2\sqrt{2}}\tan(\delta)\vec{r}_i\cdot\frac{\hat{x}+\hat{z}}{\sqrt{2}}\right)\right. \\ &\qquad\left. - \left(\frac{1}{2} - \frac{1}{2\sqrt{2}}\vec{r}_i\cdot\frac{\hat{x}+\hat{z}}{\sqrt{2}} - \frac{1}{2\sqrt{2}}\tan(\delta)\vec{r}_i\cdot\frac{\hat{x}+\hat{z}}{\sqrt{2}}\right)\right| \\ &\quad= \left|\frac{1}{\sqrt{2}}\tan(\delta)\vec{r}_i\cdot\frac{\hat{x}-\hat{z}}{\sqrt{2}}\right| \\ &\quad\le \tan(\delta)[\alpha_i + \delta], \end{aligned}$$

where the last line follows from the same reasoning as in the proof of Lemma 15. Now, using Lemma 18 we have

$$\begin{aligned} D(\mathbb{P}^i_{+1}\|\mathbb{P}^i_{-1}) &\le \frac{1}{\ln 2}\frac{\tan^2(\delta)(\alpha_i+\delta)^2/2}{(0.99)(1-0.99)} \\ &\le \Theta(1)\delta^2(\alpha_i+\delta)^2 \end{aligned}$$

for $\delta \leq 0.7$, where $\Theta(1)$ denotes some fixed constant. An identical calculation shows that $D(\mathbb{P}^i_{-1} \| \mathbb{P}^i_{+1}) \leq \Theta(1)\delta^2(\alpha_i + \delta)^2$. Finally, we have

$$\begin{aligned}
&\max_{v,v' \in \mathcal{V}} \frac{1}{n} \sum_{i=1}^n D(\mathbb{P}^i_{v_i} \| \mathbb{P}^i_{v'_i}) \\
&\quad \leq \max_{v,v' \in \{-1,1\}^n} \frac{1}{n} \sum_{i=1}^n D(\mathbb{P}^i_{v_i} \| \mathbb{P}^i_{v'_i}) \\
&\quad = \frac{1}{n} \sum_{i=1}^n \max_{v_i, v'_i \in \{-1,1\}} D(\mathbb{P}^i_{v_i} \| \mathbb{P}^i_{v'_i}) \\
&\quad \leq \frac{1}{n} \sum_{i=1}^n \Theta(1)\delta^2(\alpha_i + \delta)^2 \\
&\quad \leq \Theta(1) \left[\delta^4 + 2\delta^3 \frac{1}{n} \sum_{j=1}^n \alpha_i + \delta^2 \frac{1}{n} \sum_{j=1}^n \alpha_i^2 \right] \\
&\quad \leq \Theta(1)\delta^4,
\end{aligned}$$

where we have used $\left(\frac{1}{n}\sum_{i=1}^n \alpha_i\right)^2 \leq \frac{1}{n}\sum_{i=1}^n \alpha_i^2 \leq \frac{6500\epsilon}{n} = \Theta(1)\delta^2$.

3.4.2.4 Completing the proof

Combining the above bound with Lemma 17, upon making T zeroth-order queries to the oracle within the 100ϵ-optimum of $\mathcal{H}^\epsilon_n$, and obtaining outcomes $(Y_1, \ldots, Y_T)$, the mutual information $I(V; (\xi_1, Y_1, \ldots, \xi_T, Y_T))$ between the hidden vector V and the inputs and outputs of the oracle is upper bounded by $O(T\delta^4)$. Lemma 16 implies that for any algorithm $\mathcal{A}$ which attempts to identify the bias vector V given $(\xi_1, Y_1, \ldots, \xi_T, Y_T)$, the error probability is lower bounded by $1 - \frac{I(V;(\xi_1,Y_1,\ldots,\xi_T,Y_T))+1}{\log|\mathcal{V}|}$. Recalling that $|\mathcal{V}| \geq e^{n/8}$, we have

$$\begin{aligned}
p_e &\geq 1 - \frac{I(V; (\xi_1, Y_1, \ldots, \xi_T, Y_T)) + 1}{\log|\mathcal{V}|} \\
&\geq 1 - \frac{\Theta(1)T\delta^4 + 1}{\frac{1}{\ln 2}\frac{n}{8}} \geq 1 - \frac{\Theta(1)T\delta^4 + 1}{n/10}.
\end{aligned}$$

Let $T_{1/3}$ be value of T such that the final expression above is equal to $1/3$. A simple calculation shows $T_{1/3} = \frac{1}{\Theta(1)\delta^4}\left(\frac{n}{15} - 1\right)$. In particular, for $n > 15$, $T_{1/3} \geq \Theta(1)\frac{n^3}{\epsilon^2}$. Note that, if an algorithm makes fewer than $T_{1/3}$ zeroth-order queries, the probability that it can correct identify the hidden parameter V is less than $1/3$.

We have shown that for $n > 15$ and $\epsilon < 0.01n$, when constrained to the 100ϵ-optimum of $\mathcal{H}^\epsilon_n$, at least $\Omega\left(\frac{n^3}{\epsilon^2}\right)$ zeroth-order queries to the oracle are required to identify the bias parameter v with probability of success at least $2/3$. Our previous reduction from learning to optimization then implies that at least this many samples

are required to optimize observables in the set $\mathcal{H}_n^\epsilon$ with expected error at most ϵ. We have therefore shown Theorem 9.

3.4.3 Proof of Theorem 10: upper bound for optimizing $\mathcal{H}_n^\epsilon$

We have shown that $\Omega\left(\frac{n^3}{\epsilon^2}\right)$ zeroth-order queries to the sampling oracle are required for a 100ϵ-vicinity algorithm to optimize the family $\mathcal{H}_n^\epsilon$ to precision ϵ. In this section, we show that with a certain natural state parameterization and making only first-order queries to the sampling oracle, the family $\mathcal{H}_n^\epsilon$ can be optimized to precision ϵ with $O\left(\frac{n^2}{\epsilon}\right)$ queries by a 100ϵ-vicinity algorithm based on SGD.

We start by defining the variational ansatz that we will use in our first-order optimization procedure. We define the following n-parameter parameterization Θ:

$$|\boldsymbol{\theta}\rangle := |\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_n\rangle := \otimes_{j=1}^n e^{-i(\boldsymbol{\theta}_j + \pi/4)Y_j/2} |0\rangle^{\otimes n}.$$

This parameterization has a simple geometric interpretation: $|\boldsymbol{\theta}\rangle$ is the product state on n qubits for which the polarization of qubit j is $\sin(\pi/4 + \boldsymbol{\theta}_j)\hat{x} + \cos(\pi/4 + \boldsymbol{\theta}_j)\hat{z}$. Clearly this ansatz is natural for the family $\mathcal{H}_n^\epsilon$ in some sense.

Consider some objective observable $H_v^\delta \in \mathcal{H}_n^\epsilon$. From Lemma 8, we have that the induced objective function $f(\boldsymbol{\theta})$ is given by

$$f(\boldsymbol{\theta}) := \langle \boldsymbol{\theta} | H_v^\delta | \boldsymbol{\theta} \rangle = -\sum_{i=1}^n \cos(\boldsymbol{\theta}_i - \delta v_i).$$

Let $\mathcal{B}_\infty(\delta) \subset \mathbb{R}^n$ denote the ∞-ball of radius δ centered at the origin. Precisely, $\mathcal{B}_\infty(\delta) = \{\boldsymbol{\theta} : \|\boldsymbol{\theta}\|_\infty \leq \delta\}$. Note that the ground state of H_v^δ is the state $|\delta v_1, \delta v_2, \ldots, \delta v_n\rangle$, and hence corresponds to a parameter inside the set $\mathcal{B}_\infty(\delta)$ for any bias vector v. Furthermore, the set of states associated with $\mathcal{B}_\infty(\delta)$ is contained in the 100ϵ-optimum of $\mathcal{H}_n^\epsilon$. We state this fact as the following lemma.

Lemma 19. *The set of states associated with $\mathcal{B}_\infty(\delta)$ is contained in the 100ϵ-optimum of $\mathcal{H}_n^\epsilon$.*

Proof. For any $H_v^\delta \in \mathcal{H}_n^\epsilon$ and $\boldsymbol{\theta} \in \mathcal{B}_\infty(\delta)$, we have

$$\begin{aligned}
\langle \boldsymbol{\theta} | H_v^\delta | \boldsymbol{\theta} \rangle - \lambda_{\min}(H_v^\delta) &= \langle \boldsymbol{\theta} | H_v^\delta | \boldsymbol{\theta} \rangle - (-n) \\
&\leq n(1 - \cos(2\delta)) \\
&\leq 2n\delta^2 \\
&= 90\epsilon
\end{aligned}$$

where we used $\cos(x) \geq 1 - x^2/2$. □

We now will show that $f(\boldsymbol{\theta})$ is 0.1-strongly convex on $\mathcal{B}_\infty(\delta)$ w.r.t. the Euclidean norm. To do so, we compute its Hessian matrices $\nabla^2 f(\boldsymbol{\theta})$. We have $(\nabla^2 f(\boldsymbol{\theta}))_{ij} = \frac{\partial^2 f}{\partial \boldsymbol{\theta}_i \partial \boldsymbol{\theta}_j} = 0$ for $i \neq j$, and $(\nabla^2 f(\boldsymbol{\theta}))_{ii} = \frac{\partial^2 f}{\partial \boldsymbol{\theta}_i^2} = \cos(\boldsymbol{\theta}_i - \delta v_i)$. Since $\boldsymbol{\theta}_i \in [-\delta, \delta]$, it

must hold that $(\nabla^2 f(\boldsymbol{\theta}))_{ii} \geq \cos(2\delta) \geq 0.1$ where we used our assumption $\delta < 0.7$ for the last inequality. Since all eigenvalues of $\nabla^2 f(\boldsymbol{\theta})$ for $\boldsymbol{\theta} \in \mathcal{B}$ are at least 0.1, f is 0.1-strongly convex on $\mathcal{B}_\infty(\delta)$.

We now calculate $\vec{\Gamma}$ for this particular parameterization and some objective observable H_v^δ. Expanding the gradient as in Section 3.2 (see also Table 3.1),

$$\nabla f(\boldsymbol{\theta}) = -\sum_{j=1}^{n}\sum_{k=1}^{n} \langle \boldsymbol{\theta}| \frac{i}{2}\Big[U_{(j+1):n} Y_j U_{(j+1):n}^\dagger, \sin\Big(\frac{\pi}{4} + \delta v_k\Big) X_k + \cos\Big(\frac{\pi}{4} + \delta v_k\Big) Z_k\Big] |\boldsymbol{\theta}\rangle \, \hat{e}_j$$

where, as usual, $U_{(j+1):n} := e^{-i\boldsymbol{\theta}_n Y_n/2} \cdots e^{-i\boldsymbol{\theta}_{j+1} Y_{j+1}/2}$. We now remove terms which are trivially zero because the commutator involves operators which act nontrivially on disjoint qubits. In particular, since $U_{(j+1):n} Y_j U_{(j+1):n}^\dagger = Y_j$ in this case, then clearly $\text{qubits}(U_{(j+1):n} Y_j U_{(j+1):n}^\dagger) = \{j\}$. Dropping such terms in the expansion,

$$\nabla f(\boldsymbol{\theta}) = -\sum_{j=1}^{n} \langle \boldsymbol{\theta}| \frac{i}{2}\Big[U_{(j+1);n} Y_j U_{(j+1);n}^\dagger, \sin\Big(\frac{\pi}{4} + \delta v_j\Big) X_j + \cos\Big(\frac{\pi}{4} + \delta v_j\Big) Z_j\Big] |\boldsymbol{\theta}\rangle \, \hat{e}_j.$$

Recall that Γ_j is the sum of the magnitudes of the coefficients of the above expansion for component j. In particular, we have $\Gamma_j = \sin\big(\frac{\pi}{4} + \delta v_j\big) + \cos\big(\frac{\pi}{4} + \delta v_j\big) = \sqrt{2}\cos(\delta) = \Theta(1)$. Now, Lemma 4 implies that projected SGD, using the feasible set $B_\infty(\delta)$, outputs a parameter $\bar{\boldsymbol{\theta}}$ such that $\mathbb{E}\, f(\bar{\boldsymbol{\theta}}) - \lambda_{\min}(H) \leq \epsilon$ for all $H \in \mathcal{H}_n^\epsilon$ using $O(\|\vec{\Gamma}\|_1^2/\lambda_2\epsilon) = O(n^2/\epsilon)$ queries, where λ_2 is the strong convexity parameter (w.r.t. Euclidean norm) which is $\Theta(1)$ in our case.

As a sidenote, we point out that simply running some version of SGD with no projections (using $\mathbb{R}^n$ as the feasible set) would likely perform very well for this problem, since all local optima are also global minima in this case (even though the objective function is nonconvex on $\mathbb{R}^n$).

3.4.4 Proof of Theorem 11: general query lower bound for optimizing $\mathcal{H}_n^\epsilon$

Using a very similar argument to that of the proof of Theorem 9, we may lower bound the number of calls to $\mathcal{O}_H$ required to optimize any objective observable in the family $\mathcal{H}_n^\epsilon$ with expected error at most ϵ. In the setting of Theorem 9, the algorithm was restricted to querying the oracle with states in the 100ϵ-optimum of $\mathcal{H}_n^\epsilon$. In this section, the algorithm is allowed to query the oracle with states which may be outside this domain. We also allow the algorithm to make queries of any order, instead of just zeroth-order. As before, we actually prove a lower bound for the strictly easier problem of optimizing the subset $\mathcal{M}_n^\epsilon \subset \mathcal{H}_n^\epsilon$.

We will essentially bound the amount of information contained in a single oracle query for any order derivative and for any state. As usual, let Θ denote the parameterization given by $|\boldsymbol{\theta}\rangle = e^{-iA_p\boldsymbol{\theta}_p/2} \cdots e^{-iA_1\boldsymbol{\theta}_1/2} |\Psi\rangle$. Recall from Section 3.2.4 that,

assuming w.l.o.g. that $j_1 \leq \cdots \leq j_r$, the expansion of $\frac{\partial^r f}{\partial\theta_{j_1}\cdots\partial\theta_{j_r}}(\boldsymbol{\theta})$ in terms of nested commutators of conjugated Pauli operators is

$$\frac{\partial^r f}{\partial\theta_{j_1}\cdots\partial\theta_{j_r}}(\boldsymbol{\theta}) = \left(\frac{i}{2}\right)^r \sum_{k_1=1}^{n_{j_1}} \cdots \sum_{k_r=1}^{n_{j_r}} \left(\prod_{i=1}^{r} \beta_{k_i}^{(j_i)}\right) \langle\boldsymbol{\theta}| \left[\tilde{Q}_{k_1}^{(j_1)}, \left[\ldots, \left[\tilde{Q}_{k_r}^{(j_r)}, \sum_{l=1}^{m} \alpha_l P_l\right] \ldots\right]\right] |\boldsymbol{\theta}\rangle$$

where $A_j = \sum_{k=1}^{n_j} \beta_k^{(j)} Q_k^{(j)}$, $H = \sum_{l=1}^{m} \alpha_l P_l$, and the notation $\tilde{Q}$ is defined in Section 3.2.4. Specialize to the case in which $H = H_n^\delta \in \mathcal{M}_n^\epsilon$. Then, after removing nested commutators which are trivially zero, we may write

$$\frac{\partial^r f}{\partial\theta_{j_1}\cdots\partial\theta_{j_r}}(\boldsymbol{\theta}) = \sum_{k_1=1}^{n_{j_1}} \cdots \sum_{k_r=1}^{n_{j_r}} \sum_{l=1}^{n} \zeta_{k_1,\ldots,k_r,l} \langle\boldsymbol{\theta}| \left[\tilde{Q}_{k_1}^{(j_1)}, \left[\ldots, \left[\tilde{Q}_{k_r}^{(j_r)}, \sin(\pi/4 + \delta v_l) X_l + \cos(\pi/4 + \delta v_l) Z_l\right] \ldots\right.\right.$$

for some coefficients $\zeta_{k_1,\ldots,k_r,l}$ that are independent of v. Recall how $\mathcal{O}_{H_v^\delta}$ operates upon a query for the derivative $\frac{\partial^r f}{\partial\theta_{j_1}\cdots\partial\theta_{j_r}}(\boldsymbol{\theta})$. After doing a Pauli decomposition of the original nested commutator expression for the derivative and removing terms that are trivially zero, it samples a term with probability proportional to the magnitude of the coefficient of that term, and then obtains an unbiased estimator for that term using the procedure outlined in Section 3.2.4. Hence, we may equivalently describe the behavior of the oracle upon an r^{th}-order query of some state $\boldsymbol{\theta}$ as follows.

r^{th}-order behavior of $\mathcal{O}_{H_v^\delta}$.

Upon input of a parameterization Θ, parameter $\boldsymbol{\theta}$, and coordinate multiset $S = \{j\}$,

1. Select indices $(k_1, \ldots, k_r, l)$ with probability proportional to $|\zeta_{k_1,\ldots,k_r,l}|$.
2. Flip a coin with probability of heads $p = \frac{1}{\sqrt{2}\cos(\delta)} \sin(\pi/4 + v_l\delta) = \frac{1}{2}(1 + v_l \tan(\delta))$.
3. If heads, estimate $\langle\boldsymbol{\theta}| \left[\tilde{Q}_{k_1}^{(j_1)}, \left[\ldots, \left[\tilde{Q}_{k_r}^{(j_r)}, X_l\right] \ldots\right]\right] |\boldsymbol{\theta}\rangle$ with a single-measurement generalized Hadamard test using the procedure of Section 3.2.4. If tails, estimate $\langle\boldsymbol{\theta}| \left[\tilde{Q}_{k_1}^{(j_1)}, \left[\ldots, \left[\tilde{Q}_{k_r}^{(j_r)}, Z_l\right] \ldots\right]\right] |\boldsymbol{\theta}\rangle$.
4. Multiply the result of Step 3 by the appropriate normalization factor and output the result.

Algorithm 6: r^{th}-order behavior of $\mathcal{O}_{H_v^\delta}$.

The crucial point is that the sampling oracle cannot reveal any more information about the hidden parameter v than the outcome of the internal coin flip in Step 2 of the above box. This is because, since only Step 2 in the above box depends on the hidden parameter v, the algorithm can simulate the oracle if it has knowledge of the outcome of the internal coin flip. More formally, we have the following lemma.

Lemma 20. *Let V be the hidden parameter, ξ be the input to the oracle, W be the outcome of the internal coin flip, and Y be the output of the oracle. Then $I(V;Y|\xi) \leq I(V;W|\xi)$.*

Proof. Note from the above box that the coin flip of Step 2 is the only part of the black box's internal procedure that depends on V; the output Y is simply a stochastic function of W. Hence the variables $V \to W \to Y$ form a Markov chain, and the claim follows from the data processing inequality. □

We now use this observation along with a similar argument to that of Section 3.4.2.2 to derive the desired lower bound. As before, we have $I(V;(\xi_1, Y_1, \ldots, \xi_T, Y_T)) \leq T \max_{\xi_1} I(V;Y_1|\xi_1)$, so we will seek to upper bound $I(V;Y_1|\xi_1)$. Let $\mathbb{Q}$ denote the distribution of W_1. Let $\mathbb{Q}^l$ denote the distribution of W_1 conditioned on the oracle selecting $L = l$ in Step 1 of Algorithm 6. Let $\mathbb{Q}_v$ denote the distribution of W_1 conditioned on the hidden parameter $V = v$. Let $\mathbb{Q}^l_{v_l}$ denote the distribution of W_1 conditioned on $V_l = v_l$ and the oracle selecting $L = l$ in Step 1 of Algorithm 6. Using convexity of relative entropy multiple times, we have

$$\begin{aligned}
I(V;Y_1|\xi_1) &\leq I(V;W_1|\xi_1) \\
&= \frac{1}{|\mathcal{V}|} \sum_{v \in \mathcal{V}} D(\mathbb{Q}_v \| \mathbb{Q}) \\
&\leq \frac{1}{|\mathcal{V}|^2} \sum_{v,v' \in \mathcal{V}} D(\mathbb{Q}_v \| \mathbb{Q}_{v'}) \\
&\leq \frac{1}{|\mathcal{V}|^2} \sum_{v,v' \in \mathcal{V}} \mathop{\mathbb{E}}_{L} D(\mathbb{Q}^L_{v_L} \| \mathbb{Q}^L_{v'_L}) \\
&\leq \max_{v,v' \in \mathcal{V}} \mathop{\mathbb{E}}_{L} D(\mathbb{Q}^L_{v_L} \| \mathbb{Q}^L_{v'_L}) \\
&\leq \max_{v,v' \in \mathcal{V}} \max_{l} D(\mathbb{Q}^l_{v_l} \| \mathbb{Q}^l_{v'_l})
\end{aligned}$$

where we have used the fact that $\mathbb{Q}_v = \mathbb{E}_L \mathbb{Q}^L_{v_L}$, where the expectation value is over the choice of parameter L made by the black box.

It remains to bound the final expression above. Clearly $D(\mathbb{Q}^l_{+1} \| \mathbb{Q}^l_{+1}) = D(\mathbb{Q}^l_{-1} \| \mathbb{Q}^l_{-1}) = 0$. Now we calculate $D(\mathbb{Q}^l_{+1} \| \mathbb{Q}^l_{-1})$. Recall that for the distribution $\mathbb{Q}^l_{v_l}$, the probability of "heads" is $\frac{1}{2}(1 + v_l \tan(\delta))$. By nearly identical arguments to those in the proof of Theorem 9, it then follows immediately from Lemma 18 that there exists some constant c such that $D(\mathbb{Q}^l_{+1} \| \mathbb{Q}^l_{-1}) \leq c\delta^2$ for $\delta < 0.7$. Similarly, $D(\mathbb{Q}^l_{-1} \| \mathbb{Q}^l_{+1}) \leq c\delta^2$.

At this point, we may follow a virtually identical argument to that in Section 3.4.2.4 to find that, for $n \geq 15$ and $\epsilon \leq 0.01n$, at least $\Omega\left(\frac{n^2}{\epsilon}\right)$ oracle queries are required to identify the hidden bias parameter v with probably at least $2/3$. Hence, at least $\Omega\left(\frac{n^2}{\epsilon}\right)$ oracle queries are required to optimize $\mathcal{H}^\epsilon_n$ with worst-case expected error at most ϵ.

Since this lower bound has a matching upper bound via first-order oracle queries and SGD (up to constant factors), we see that SGD is in fact essentially optimal among all black-box strategies for optimizing the family $\mathcal{H}_n^\epsilon$.

3.5 Further discussion and open questions

It would be interesting to understand the behavior of the objective function $f(\boldsymbol{\theta})$ near a local minimum for problems and variational ansatzes which appear in practice. In particular, it would be interesting to understand how the strong convexity of $f(\boldsymbol{\theta})$ typically behaves near a local minimum. Without a strong convexity guarantee, stochastic descent methods typically have query upper bounds scaling with the precision like $O(1/\epsilon^2)$. However, given a promise of λ-strong convexity, the cost is typically $O(1/\lambda\epsilon)$. For the toy model $\mathcal{H}_n^\epsilon$ that we analyzed, we showed that with an appropriate choice of ansatz, the problem was $\Theta(1)$-strongly convex with respect to the 2-norm. As a result of this property, we were able to obtain a $O(1/\epsilon)$ query upper bound for optimizing this family with SGD. We apparently were able to exploit strong convexity by making a prudent choice of variational ansatz for the problem class at hand.

This situation may be viewed as the opposite of that studied in [McC+18], which essentially considered a situation in which the variational ansatz looks random. In this situation, the gradient of the objective function is highly concentrated around zero. One way to view the difference in our models is that in our paper there are n independent (i.e. commuting) degrees of freedom while in [McC+18] different terms in the Hamiltonian and pulses have the commutation relations that we would expect from Haar-random projectors. Our model could be seen as justified by the common intuition in many-body physics that local unitaries applied to the ground state create quasiparticles, and that in an n-qubit system $O(n)$ independent quasiparticles are possible. Their model, on the other hand, could be justified by the assumption that the variational ansatz is far from a local minimum and so the pulses act like random local unitaries. It would be interesting to understand which of these scenarios is more realistic in practice. In particular, one might hope that theoretically motivated ansatzes, such as the unitary-coupled-cluster ansatz in quantum chemistry, could possess properties near an optimum (such as strong convexity) that make them more amenable to efficient optimization.

As another open problem, recall that for the toy problem $\mathcal{H}_n^\epsilon$ we proved that taking analytic k^{th}-order derivative measurements for $k \geq 2$ provides no benefit over taking zeroth- and first-order measurements. However, the observables in the family $\mathcal{H}_n^\epsilon$ are extremely simple, being merely 1-local and having unentangled ground states. It seems plausible that taking second (or higher) order measurements could be beneficial for more complicated problems. It would be interesting to understand how higher-order measurements could improve convergence in such cases.

Another point that we left unaddressed is the issue of noise. It would be interesting to study how to take analytic gradient measurements in the presence of noise, and what impact this has on the convergence rate of stochastic optimization meth-

ods. In particular, these methods are quite robust against unbiased noise, but their effectiveness in the presence of biased noise is less understood.

Finally, it would be interesting to understand whether our SGD-based convergence bounds could be improved upon by approximating the *natural gradient* [Ama98] via shallow circuits, as recently proposed in [Sto+20] and further studied in [KB19; SK20]; this method is essentially equivalent to a prior proposal [McA+19] for simulating imaginary-time dynamics in variational algorithms.

Chapter 4

Classical Algorithms for Random Shallow 2D Quantum Circuits, I: Technical Exposition

4.1 Introduction

As quantum computers add more qubits and gates, where is the line between classically simulable and classically hard to simulate? And once the size and runtime of the quantum computer are chosen, which gate sequence is hardest to simulate?

So far, our answers to these questions have been informal or incomplete. On the simulation side, Markov and Shi [MS08] showed that a quantum circuit could be classically simulated by contracting a tensor network with cost exponential in the treewidth of the graph induced by the circuit. When applied to n qubits in a line running a circuit with depth d, the simulation cost of this algorithm is $\exp(\Theta(\min(n, d)))$. More generally we could consider $n = L_1 L_2$ qubits arranged in an $L_1 \times L_2$ grid running for depth d, in which case the simulation cost would be

$$\exp\Big(\Theta(\min(L_1 L_2, L_1 d, L_2 d))\Big). \tag{4.1}$$

In other words, we can think of the computation as taking up a space-time volume of $L_1 \times L_2 \times d$ and the simulation cost is dominated by the size of the smallest cut bisecting this volume. An exception is for $d = 1$ or $d = 2$, which have simple exact simulations [TD04]. Some restricted classes such as stabilizer circuits [Got98] or one dimensional systems that are sufficiently unentangled [Vid03; Vid04; Osb06] may also be simulated efficiently. However, the conventional wisdom has been that in general, for 2D circuits with $d \geq 3$, the simulation cost scales as Equation (4.1).

These considerations led IBM to propose the benchmark of "quantum volume" [Cro+19] which in our setting is $\exp(\sqrt{d \min(L_1, L_2)})$; this does not exactly coincide with Equation (4.1) but qualitatively captures a similar phenomenon. The idea of quantum volume is to compare quantum computers with possibly different architectures by evaluating their performance on a simple benchmark. This benchmark task is to

perform n layers of random two-qubit gates on n qubits, and being able to perform this with $\lesssim 1$ expected gate errors corresponds to a quantum volume of $\exp(n)$.[1] Google's quantum computing group has also proposed random unitary circuits as a benchmark task for quantum computers [Boi+18]. While their main goal has been quantum computational supremacy [Nei+18; Aru+19], random circuits could also be used to diagnose errors including those that go beyond single-qubit error models by more fully exploring the configuration space of the system [Cro+19].

These proposals from industry reflect a rough consensus that simulating a 2D random quantum circuit should be nearly as hard as exactly simulating an arbitrary circuit with the same architecture, or in other words that random circuit simulation is nearly as hard as the *worst case*, given our current state of knowledge.

To the contrary, we prove (assuming standard complexity-theoretic conjectures) that for a certain family of constant-depth architectures, classical simulation of typical instances with small allowed error is easy, despite worst-case simulation being hard (by which we mean, it is classically intractable to simulate an arbitrary random circuit realization with arbitrarily small error). For these architectures, we show that a certain algorithm exploiting the randomness of the gates and the allowed small simulation error can run much more quickly than the scaling in Equation (4.1), running in time $O(L_1 L_2)$. While our proof is architecture-specific, we give numerical and analytical evidence that for sufficiently low constant values of d, the algorithm remains efficient more generally. The intuitive reason for this is that the simulation of 2D shallow random circuits can be reduced to the simulation of a form of effective 1D dynamics which includes random local unitaries and weak measurements. The measurements cause the 1D process to generate much less entanglement than it could in the worst case, making efficient simulation possible. Such dynamics consisting of random local gates with interspersed measurements has in fact recently become the subject of an intensive research focus [LCF18; Cha+19; SRN19; LCF19; SRS19; Cho+20; GH20a; BCA20; Jia+20; GH20b; Zab+20; TZ20; NS20; AB20; Fan+20; Li+20; LAB21; SH20; Ipp+21; FA20; SRS20; Vij20; LP20; LF20; TFD20; FHH21; Nah+21; IK21], and our simulation algorithm can be viewed as an application of this line of work. Furthermore, the measurement-strength-driven entanglement phase transitions observed in these processes are closely related to the computational phase transition we observe for our algorithms. Before discussing this in greater detail, we review the main arguments for the prevailing belief that random circuit simulation should be nearly as hard as the worst case.

Evidence from complexity theory. A long line of work has shown that it is worst-case hard to either sample from the output distributions of quantum circuits or compute their output probabilities with exponentially small error [TD04; Aar05; BJS10; AA11; BMS16; BMS17; HM17]. While the requirements of worst-case and near-exact simulation are rather strong, these results do apply to any quantum circuit family that becomes universal once post-selection [Aar05] is allowed, thereby including

[1] Our calculation of quantum volume for 2D circuits above uses the additional fact that, assuming for simplicity that $L_1 \leq L_2$, we can simulate a fully connected layer of gates on $L_2 x$ qubits (for $x \leq L_1$) with $O(x L_2 / L_1)$ locally connected 2D layers using the methods of [Ros13]. Then x is chosen to maximize $\min(L_2 x, d/(x L_2 / L_1))$.

noninteracting bosons [AA11] and 2D depth-3 circuits [TD04]. The hardness results are also based on the widely believed conjecture that the polynomial hierarchy (PH) is infinite, or more precisely that approximate counting is weaker than exact counting. Since these results naturally yield worst-case hardness, they do not obviously imply that random circuits should be hard. In some cases, additional conjectures can be made to extend the hardness results to some form of average-case hardness (as well as ruling out approximate simulations) [AA11; BMS16; AC17], but these conjectures have not received widespread scrutiny. Besides stronger conjectures, these hardness results usually require that the quantum circuits have an "anti-concentration" property, meaning roughly that their outputs are not too far from the uniform distribution [HM18]. While random circuits are certainly not the only route to anti-concentration (applying a Hadamard gate to each qubit of $|0\rangle^{\otimes n}$ would do), they are a natural way to combine anti-concentration with an absence of any obvious structure (e.g. Clifford gates) that might admit a simple simulation (however, note that constant-depth random quantum circuits do not have the anti-concentration property [DHB20]). Furthermore, a line of work beginning with Ref. [Bou+19] (see [Mov19; Bou+21; KMM21] for subsequent improvements) has established that random circuit simulation admits a worst-to-average case reduction for the computation of output probabilities. In particular, the ability to near-exactly compute the probability of some output string for a $1 - 1/\operatorname{poly}(n)$ fraction of Haar-random circuit instances on some architecture is essentially as hard as computing output probabilities for an arbitrary circuit instance with this architecture, which is known to be #P-hard even for certain 2D depth-3 architectures.

Near-maximal entanglement in random circuits. Haar-random states on n qudits are nearly maximally entangled across all cuts simultaneously [Pag93; HLW06]. Random quantum circuits on $L \times L \times \cdots$ arrays of qudits achieve similar near-maximal entanglement across all possible cuts once the depth is $\Omega(L)$ [DOP07; HM18] and before this time, the entanglement often spreads "ballistically" [LS14; BKP19]. Random tensor networks with large bond dimension nearly obey a min-flow/max-cut-type theorem [Hay+16; Has17], again meaning that they achieve nearly maximal values of an entanglement-like quantity. These results suggest that when running algorithms based on tensor contraction, random gates should be nearly the hardest possible gates to simulate.

Absence of algorithms taking advantage of random inputs. There are not many algorithmic techniques known that simulate random circuits more easily than worst-case circuits. There are a handful of exceptions. In the presence of any constant rate of noise, random circuits [YG17; GD18], IQP circuits [BMS17] and (for photon loss) boson sampling [KK14; OB18] can be efficiently simulated. These results can also be viewed as due to the fact that fault-tolerant quantum computing is not a generic phenomenon and requires structured circuits to achieve (see [BMS17] for discussion in the context of IQP). Permanents of random matrices whose entries have small nonzero mean can be approximated efficiently [EM18], while the case of boson sampling corresponds to entries with zero mean and the approach of [EM18] is known to fail there. A heuristic approximate simulation algorithm based on tensor network contraction [Pan+20] was recently proposed and applied to random

circuits, although for this algorithm it is unclear how the approximations made are related to the overall simulation error incurred (in contrast, our algorithm based on matrix product states can bound the overall simulation error it is making, even when comparison with exact simulation is not feasible). In practice, evidence for a hardness conjecture often is no more than the absence of algorithms. Indeed, while some approximation algorithms are known for estimating output probabilities of constant-depth circuits [BGM21], IQP circuits [SB09] and boson sampling [AA11] up to additive error δ in time $\mathrm{poly}(n, 1/\delta)$, these are not very helpful for random circuits where typical output probabilities are $\sim 2^{-n}$.

Despite the above intuitive arguments for why the simulation of uniformly random circuits should be nearly as hard as the worst case, we (1) prove that there exist architectures for which this is not the case, and (2) give evidence that this result is not architecture-specific, but is rather a general property of sufficiently shallow random circuits. To this end, we propose and implement a simulation algorithm based on a 2D-to-1D mapping in conjunction with tensor network methods. While for brevity we focus on only this algorithm in the current chapter, in Section 5.1.3 we introduce and study a second simulation algorithm (referred to as `Patching`) based on locally simulating spatially disconnected regions which are then "stitched" together. The performance of both algorithms is related to certain entropic quantities.

We also give evidence of computational phase transitions for our proposed simulation algorithms driven by circuit depth and qudit dimension. Previously it was known that phase transitions between classical and quantum computation existed as a function of the noise parameter in conventional quantum computation [Sho96; AB96; HN03; Raz04; VHP05; Buh+06; Kem+08] as well as in measurement-based quantum computing (MBQC) [RBH05; Bar+09]. In the noiseless setting, besides the gap between depth-2 and depth-3 circuits [TD04], a computational phase transition as function of rate of qubit loss during the preparation of a resource state for MBQC [Bro+08] and (under additional assumptions) as a function of duration of time evolution for simulating dynamics generated by quadratic bosonic Hamiltonians [Des+18; MMD19] was also known.

4.1.1 Our results

We give two classes of results, which we summarize in more detail below. The first consists of rigorous separations in complexity between worst-case simulation[2] and approximate average-case simulation (for sampling) and between near-exact average-case simulation and approximate average-case simulation (for computing output probabilities) for random circuit families defined with respect to certain circuit architectures. While these results are rigorous, they are proved with respect to a contrived architecture and therefore do not address the question of whether random shallow circuits are classically simulable more generally. To address this issue, we also give conjectures on the performance of our algorithms for more general and more nat-

[2]Unless specified otherwise, we use *worst-case simulation* to refer to the problem of exactly simulating an arbitrary circuit instance.

ural architectures. Our second class of results consists of analytical and numerical evidence in favor of these conjectures.

4.1.2 Provable complexity separations

We now summarize our provable results for particular circuit architectures. We first define more precisely what we mean by an "architecture".

Definition 14 (Architecture)**.** *An* architecture A *is an efficiently computable mapping from positive integers L to circuit layouts $A(L)$ defined on rectangular grids with sidelengths $L \times f(L)$ for some function $f(L) \leq \mathrm{poly}(L)$. A "circuit layout" is a specification of locations of gates in space and time and the number of qudits acted on by each gate. (The gate itself is not specified.) For any architecture A, we obtain the associated Haar-random circuit family acting on qudits of constant dimension q, $C_{A,q}$, by specifying every gate in A to be distributed according to the Haar measure and to act on qudits of dimension q which are initialized in a product state $|1\rangle^{\otimes(L\times f(L))}$.*

In this paper, we only consider architectures that are constant depth and spatially 2-local (that is, a gate either acts on a single site or two adjacent sites); therefore, "architecture" for our purposes always refers to a constant-depth spatially 2-local architecture. The above definition permits architectures for which the layout of the circuit itself may be different for different sizes. However, it is natural for a circuit architecture to be spatially periodic, and furthermore for the "unit cells" of the architecture to be independent of L. We formalize this as a notion of *uniformity*, which we define more precisely below.

Definition 15 (Uniformity)**.** *We call a constant-depth architecture A* uniform *if there exists some spatially periodic circuit layout B on an infinite square lattice such that, for all positive integers L, $A(L)$ is a restriction of B to a rectangular sub-grid with sidelengths $L \times f(L)$ for some $f(L) \leq \mathrm{poly}(L)$. A random circuit family $C_{A,q}$ associated with a uniform architecture A is said to be a uniform random circuit family.*

While uniformity is a natural property for a circuit architecture to possess, our provable separations are with respect to certain non-uniform circuit families. In particular, we prove in Section 4.3 that for any fixed $0 < c < 1$, there exists some non-uniform circuit architecture A acting on n qubits such that, if C_A is the Haar-random circuit family associated with A acting on qubits,

1. **Exact worst-case sampling is hard**: There does not exist a $\mathrm{poly}(n)$-time classical algorithm that exactly samples from the output distribution of arbitrary realizations of C_A unless the polynomial hierarchy collapses to the third level.

2. **Near-exact average-case computation of output probabilities is hard**: Given an arbitrary fixed output string $\mathbf{x}$, there does not exist a $\mathrm{poly}(n)$-time classical algorithm for computing the probability of obtaining $\mathbf{x}$, $|\langle \mathbf{x}| C_A |1\rangle^{\otimes n}|^2$, up to additive error $2^{-\Theta(n\log(n))}$ with probability at least $1 - 1/\mathrm{poly}(n)$ over

choice of circuit instance, unless a #P-hard function can be computed in randomized polynomial time.

3. **Approximate average-case sampling is easy**: There exists a classical algorithm that runs in time $O(n)$ and, with probability at least $1 - 2^{-n^c}$ over choice of circuit instance, samples from the output distribution of C_A up to error at most 2^{-n^c} in total variation distance.

4. **Approximate average-case computation of output probabilities is easy**: There exists a classical algorithm that runs in time $O(n)$ and, for an arbitrary output string $\mathbf{x}$, with probability at least $1 - 2^{-n^c}$ over choice of circuit instance, estimates $|\langle \mathbf{x}| C_A |0\rangle^{\otimes n}|^2$ up to additive error $2^{-n}/2^{n^c}$. (This should be compared with 2^{-n}, which is the average output probability over choices of $\mathbf{x}$.)

The first two points above follow readily from prior works (respectively [TD04] and [Bou+21; KMM21]), while the latter two follow from an analysis of the behavior of one of our simulation algorithms for this architecture. These algorithms improve on the previously best known simulation time for this family of architectures of $2^{\Theta(L)} = 2^{\Theta(n^{c'})}$ for some constant $c'(c) < 1$ based on an exact simulation based on tensor network contraction. We refer to the architectures for which we prove the above separations as "extended brickwork architectures" (see Figure 4-3 for a specification), as they are related to the "brickwork architecture" [BFK09] studied in the context of MBQC.

Implications for quantum computational supremacy. The worst-case to average-case reductions that imply the second item above have been widely cited as evidence for the conjectures that underpin random-circuit-based quantum computational supremacy proposals. Yet, the existence of an architecture for which the fourth item is also true indicates that the robustness of the reduction could not be sufficiently improved to actually prove those conjectures, barring the introduction of some new technique that is sensitive to the circuit depth. Thus, although our algorithms can only efficiently simulate shallow random circuits, they accentuate a fundamental weakness in the main source of formal evidence for hardness even in the case of deep circuits. (See Section 5.3 for further discussion of the relationship to this line of work.)

4.1.3 Conjectures for uniform architectures

While the above results are provable, they are unfortunately proved with respect to a contrived non-uniform architecture, and furthermore do not provide good insight into how the simulation runtime scales with simulation error and simulable circuit fraction. An obvious question is then whether efficient classical simulation remains possible for "natural" random circuit families that are sufficiently shallow, and if so, how the runtime scales with system size and error parameters. We argue that it does, but that a computational phase transition occurs for our algorithms when the depth (d) or local Hilbert space dimension (q) becomes too large. Here we are studying the simulation cost as $n \to \infty$ for fixed d and q. Intuitively, there are many constant-depth

random circuit families for which efficient classical simulation is possible, including many "natural" circuit architectures (it seems plausible that *any* depth-3 random circuit family on qubits is efficiently simulable). However, we expect a computational phase transition to occur for sufficiently large constant depths or qudit dimensions, at which point our algorithms become inefficient. The location of the transition point will in general depend on the details of the architecture. The conjectures stated below are formalizations of this intuition.

We now state our conjectures more precisely. Conjecture 1 essentially states that there are *uniform* random circuit families for which worst-case simulation (in the sense of sampling or computing output probabilities) is hard, but approximate average-case simulation can be performed efficiently. (Worst-case hardness for computing probabilities also implies a form of average-case hardness for computing probabilities, as discussed above.) This is stated in more-or-less the weakest form that seems to be true and would yield a polynomial-time simulation. However, we suspect that the scaling is somewhat more favorable. Our numerical simulations and toy models are in fact consistent with a stronger conjecture, Conjecture 1', which if true would yield stronger run-time bounds. Conversely, Conjecture 2 states that if the depth or local qudit dimension of such an architecture is made to be a sufficiently large constant, our two proposed algorithms experience computational phase transitions and become inefficient even for approximate average-case simulation.

Conjecture 1. *There exist uniform architectures and choices of q such that, for the associated random circuit family $C_{A,q}$, (1) worst-case simulation of $C_{A,q}$ (in terms of sampling or computing output probabilities) is classically intractable unless the polynomial hierarchy collapses, and (2) our algorithms approximately simulate $C_{A,q}$ with high probability. More precisely, given parameters ε and δ, our algorithms run in time bounded by $\mathrm{poly}(n, 1/\varepsilon, 1/\delta)$ and can, with probability $1-\delta$ over the random circuit instance, sample from the classical output distribution produced by C_q up to variational distance error ε and compute a fixed output probability up to additive error ε/q^n.*

Conjecture 1'. *For any uniform random circuit family $C_{A,q}$ satisfying the conditions of Conjecture 1, efficient simulation is possible with runtime replaced by*

$$n^{1+o(1)} \cdot \exp\Big(O(\sqrt{\log(1/\varepsilon\delta)})\Big). \tag{4.2}$$

Conjecture 2. *For any uniform random circuit family $C_{A,q}$ satisfying the conditions of Conjecture 1, there exists some constant q^* such that our algorithms become inefficient for simulating $C_{A,q'}$ for any constant $q' > q^*$, where $C_{A,q'}$ has the same architecture as as C_q but acts on qudits of dimension q'. There also exists some constant k^* such that, for any constant $k > k^*$, our algorithms become inefficient for simulating the composition of k layers of the random circuit, $C^k_{A,q} \circ \cdots \circ C^2_{A,q} \circ C^1_{A,q}$, where each $C^i_{A,q}$ is i.i.d. and distributed identically to $C_{A,q}$. In the inefficient regime, for fixed ε and δ the runtime of our algorithms is $2^{O(L)}$.*

Our evidence for these conjectures, which we elaborate upon in the following sections, consists primarily of the following elements.

1. A rigorous reduction from the 2D simulation problem to a 1D simulation problem that can be efficiently solved with high probability if certain conditions on expected entanglement in the 1D state are met (Section 4.2).

2. Convincing numerical evidence that these conditions are indeed met for a specific worst-case-hard uniform random circuit family and that in this case the algorithm is extremely successful in practice (Section 4.4).

3. Heuristic analytical evidence for both conjectures using a mapping from random unitary circuits to classical statistical mechanical models (Section 4.5), and for Conjecture 1' using a toy model which can be more rigorously studied (Section 4.2.4).

The uniform random circuit family for which we collect the most evidence for classical simulability is associated with the depth-3 "brickwork architecture" [BFK09] (see also Figure 4-3 for a specification).

In the remainder of the paper we develop the evidence for our conjectures outlined in the three items above, and also present our rigorous complexity separation in Section 4.3.

4.2 Simulation by reduction to 1D dynamics

We reduce the problem of simulating a constant-depth quantum circuit acting on a $L \times L'$ grid of qudits to the problem of simulating an associated "effective dynamics" in 1D on L qudits which is iterated for L' timesteps, or alternatively on L' qudits which is iterated for L timesteps. This mapping is rigorous and is related to previous maps from 2D quantum systems to 1D system evolving in time [RB01; Kim17a; Kim17b]. The effective 1D dynamics is then simulated using the time-evolving block decimation algorithm of Vidal [Vid04]. In analogy, we call this algorithm space-evolving block decimation (`SEBD`). In Section 4.2.1, we specify the details of `SEBD` and rigorously bound the simulation error made by the algorithm in terms of quantities related to the entanglement spectra of the effective 1D dynamics and give conditions in which it is provably asymptotically efficient for sampling and estimating output probabilities with small error. `SEBD` is self-certifying in the sense that it can construct confidence intervals for its own simulation error and for the fraction of random circuit instances it can simulate. This makes numerically studying the algorithm's performance feasible, and is a crucial difference between `SEBD` and heuristics based on approximate tensor network contractions (e.g. [Pan+20]) in which the error incurred by truncating bonds of the tensor network cannot be directly related to the overall simulation error.

A 1D unitary quantum circuit on L qubits iterated for L^c timesteps with $c > 0$ is generally hard to simulate classically in poly(L)-time, as the entanglement across any cut can increase linearly in time. However, the form of 1D dynamics that a shallow circuit maps to includes measurements as well as unitary gates. While the unitary gates tend to build entanglement, the measurements tend to destroy entanglement

and make classical simulation more tractable. It is *a priori* unclear which effect has more influence.

Fortunately, unitary-and-measurement processes have been studied in a flurry of recent papers from the physics community [LCF18; Cha+19; SRN19; LCF19; SRS19; Cho+20; GH20a; BCA20; Jia+20; GH20b; Zab+20; TZ20; NS20; AB20; Fan+20; Li+20; LAB21; SH20; Ipp+21; FA20; SRS20; Vij20; LP20; LF20; TFD20; FHH21; Nah+21; IK21]. The consensus from this work is that processes consisting of entanglement-creating unitary evolution interspersed with entanglement-destroying measurements can be in one of two phases, where the entanglement entropy equilibrates to either an area law (constant), or to a volume law (extensive). When we vary parameters like the fraction of qudits measured between each round of unitary evolution, a phase transition is observed. The existence of a phase transition appears to be robust to variations in the exact model, such as replacing projective measurements on a fraction of the qudits with weak measurements on all of the qudits [LCF19; SRS19], or replacing Haar-random unitary evolution with Clifford [LCF18; LCF19; GH20a; Cho+20] or Floquet [SRN19; LCF19] evolution. This suggests that the efficiency of the `SEBD` algorithm depends on whether the particular circuit depth and architecture being simulated yields effective 1D dynamics that falls within the area-law or the volume-law regime. It also suggests a computational phase transition in the complexity of the `SEBD` algorithm. Essentially, decreasing the measurement strength or increasing the qudit dimension in these models is associated with moving toward a transition into the volume-law phase. Since increasing the 2D circuit depth is associated with decreasing the measurement strength and increasing the local dimension of the associated effective 1D dynamics, this already gives substantial evidence in favor of a computational phase transition in `SEBD`.

`SEBD` is inefficient if the effective 1D dynamics are on the volume-law side of the transition, and we expect it to be efficient on the area-law side because, in practice, dynamics obeying an area law for the von Neumann entanglement entropy are generally efficiently simulable. However, definitively proving that `SEBD` is efficient on the area-law side faces the obstacle that there are known contrived examples of states which obey an area law but cannot be efficiently simulated with matrix product states [Sch+08]. We address this concern by directly studying the entanglement spectrum of unitary-and-measurement processes in the area-law phase. To do this, we introduce a toy model for such dynamics which may be of independent interest. For this model, discussed more in Section 4.2.4, we rigorously derive an asymptotic scaling of Schmidt values across some cut as $\lambda_i \propto \exp\left(-\Theta(\log^2 i)\right)$ which is consistent with the scaling observed in our numerical simulations. Moreover, for this toy model we show that with probability at least $1-\delta$, the equilibrium state after iterating the process can be ε-approximated by a state with Schmidt rank $r \leq \exp\left(O(\sqrt{\log(n/\varepsilon\delta)})\right)$. Taking this toy model analysis as evidence that the bond dimension of `SEBD` when simulating a circuit whose effective 1D dynamics is in an area-law phase obeys this asymptotic scaling leads to Conjecture 1'.

4.2.1 Specification of algorithm

In this section, we assume the reader is familiar with standard tensor network methods, particularly algorithms for manipulating matrix product states (see e.g. [Orú14; BC17] for reviews).

For concreteness, we consider a rectangular grid of qudits with local Hilbert space dimension q, although the algorithm could be similarly defined for different lattices. Assume WLOG that the grid consists of $n = L_1 \times L_2$ qudits, where L_1 is the number of rows, L_2 is the number of columns, and $L_1 \leq L_2$. For each qudit, let $|i\rangle, i \in [q]$ label a set of basis states which together form the computational basis. Assume all gates act on one site or two neighboring sites, and the starting state is $|1\rangle^{\otimes n}$. Let d denote the circuit depth, which should be regarded as a constant. For a fixed circuit instance C, the goal is to sample from a distribution close to $\mathcal{D}_C$, defined to be the distribution of the output of C upon measuring all qudits in the computational basis. For an output string $\mathbf{x} \in [q]^n$, we let $\mathcal{D}_C(\mathbf{x})$ denote the probability of the circuit outputting $\mathbf{x}$ after measurement. The high-level behavior of the algorithm is illustrated in Figure 4-1. Recall that C can always be *exactly* simulated in time $L_2 q^{\Theta(dL_1)}$ using standard tensor network algorithms [MS08].

Since all of the single-qudit measurements commute, we can measure the qudits in any order. In particular, we can first measure all of the sites in `column 1`, then those in `column 2`, and iterate until we have measured all L_2 columns. This is the measurement order we will take. Now, consider the first step in which we measure `column 1`. Instead of applying all of the gates of the circuit and then measuring, we may instead apply only the gates in the *lightcone* of `column 1`, that is, the gates that are causally connected to the measurements in `column 1`. We may ignore qudits that are outside the lightcone, by which we mean qudits that are outside the support of all gates in the lightcone.

Let $\rho_1 = |1\rangle\langle 1|^{\otimes L_1}$ denote the trivial starting state that is a tensor product of $|1\rangle$ states in `column 1`, which the algorithm represents as an MPS. Let V_1 denote the isometry corresponding to applying all gates in the lightcone of this column. The algorithm simulates the application of V_1 by adding qudits in the lightcone of `column 1` as necessary and applying the associated unitary gates, maintaining the description of the state as an MPS of length L_1 as illustrated in Figure 4-2. Since there are up to $d + 1$ columns in the lightcone of `column 1`, each tensor of the MPS after the application of V_1 has up to $d + 1$ dangling legs corresponding to physical indices, for a total physical dimension of at most q^{d+1}. Since in the application of V_1, there are up to $O(d^2)$ gates that act between any two neighboring rows, the (virtual) bond dimension of the updated MPS is at most $q^{O(d^2)}$.

We now simulate the computational basis measurement of `column 1`. More precisely, we measure the qudits of `column 1` one by one. We first compute the respective probabilities $p_1, p_2, \ldots, p_q$ of the q possible measurement outcomes for the first qudit. This involves contracting the MPS encoding $V_1 \rho_1 V_1^\dagger$. We now use these probabilities to classically sample an outcome $i \in [q]$, and update the MPS to condition on this outcome. That is, if (say) we obtain outcome `1` for site i, we apply the projector $|1\rangle\langle 1|$ to site i of the state and subsequently renormalize. After doing this for every qudit in

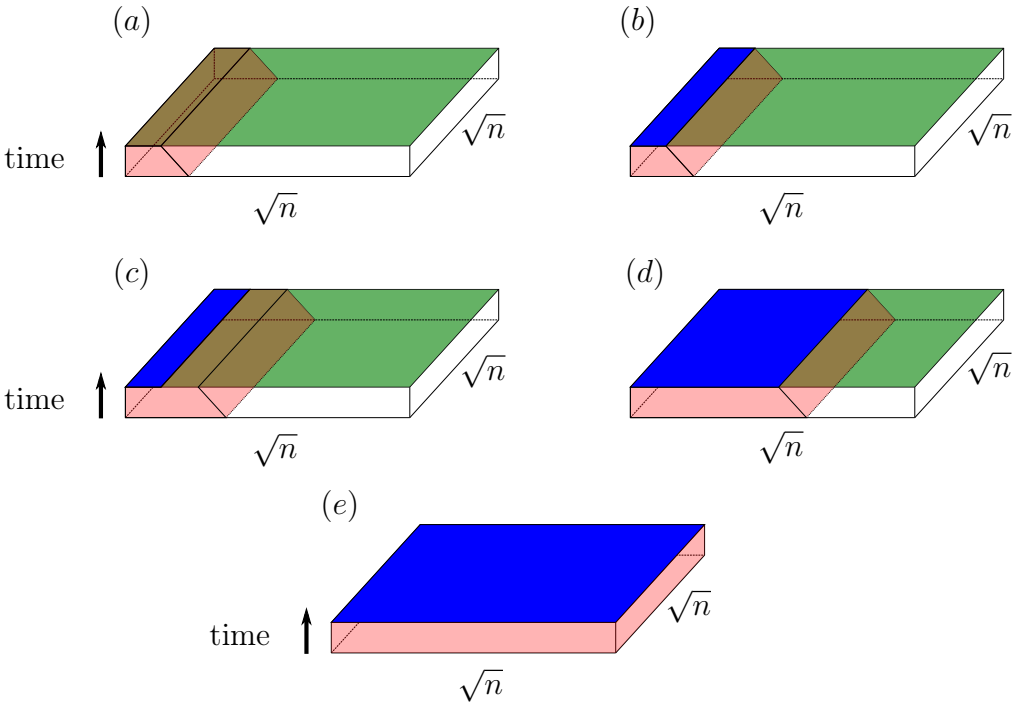


Figure 4-1: Schematic depiction of `SEBD` simulating a shallow 2D circuit. In all figures, the 2D circuit is depicted as a spacetime volume, with time flowing upwards. The blue regions correspond to sites for which measurements have been simulated, while green regions correspond to unmeasured sites. In (a), we apply all gates in the lightcone of `column 1`, namely, those gates intersecting the spacetime volume shaded red. In (b), we simulate the computational basis measurement of `column 1`. In (c), we apply all gates in the lightcone of `column 2` that were previously unperformed. Figure (d) depicts the algorithm at an intermediate stage of the simulation, after the measurements of about half of the qudits have been simulated. The algorithm stores the current state as an MPS at all times, which may be periodically compressed to improve efficiency. Figure (e) depicts the algorithm at completion: the measurements of all n of the qudits have been simulated.

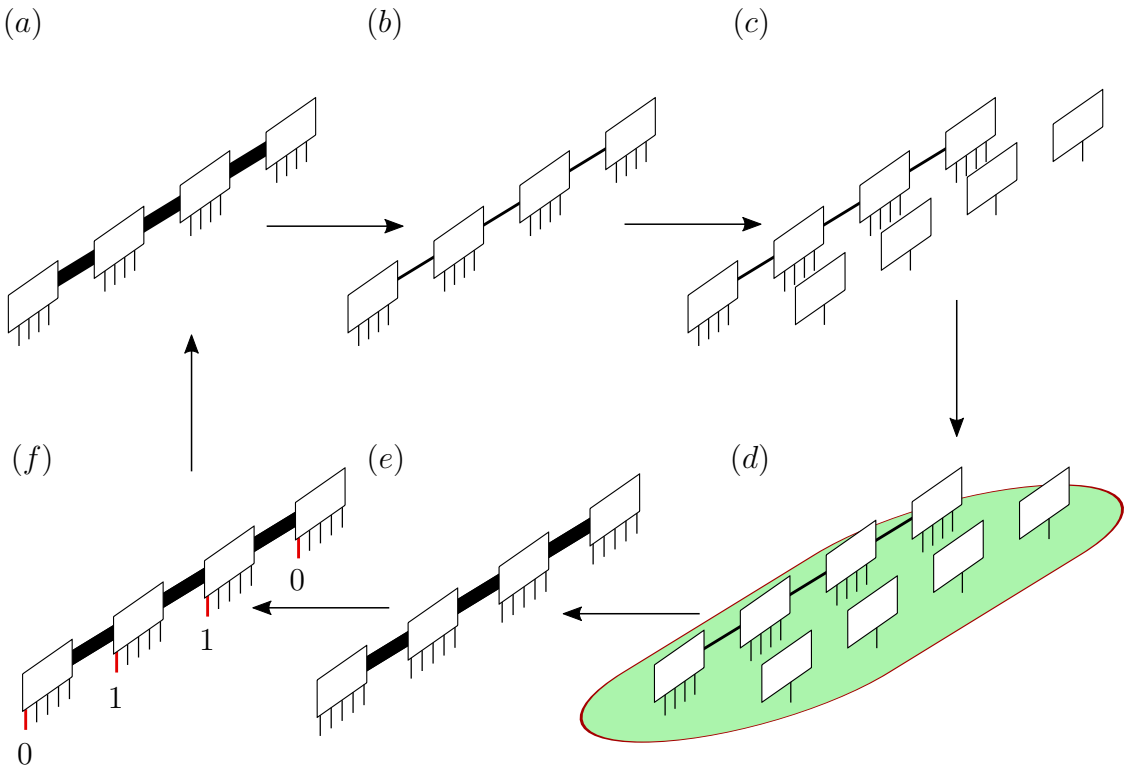


Figure 4-2: Iteration of `SEBD`. In (a), we begin with an MPS describing the current state ρ_j. In (b), the MPS is compressed via truncation of small Schmidt values. This will generally decrease the bond dimension of the MPS, depicted in the cartoon by a reduction in the thickness of the lines between tensors. In (c), qudits acted on by V_j that are not already incorporated into the current state are added to the MPS (increasing the physical bond dimension of the MPS) and initialized in $|0\rangle$ states. In (d), the unitary gates associated with V_j are applied. Figure (e) depicts the MPS after the application of V_j; the virtual bond dimension generally is increased by the application of V_j. In (f), the measurement of `column j` is performed, and the outcome `0110` is obtained. Subsequently, `column j` is projected onto the outcome `0110`, removing the physical legs associated with these sites from the MPS. The resulting state is ρ_{j+1}.

the column, we have exactly sampled an output string $\mathbf{x}_1 \in [q]^{L_1}$ from the marginal distribution on `column 1`, and are left with an MPS description of the pure, normalized, post-measurement state ρ_2 proportional to $\mathrm{tr}_{\texttt{column 1}}\left(\Pi_1^{\mathbf{x}} V_1 \rho_1 V_1^\dagger \Pi_1^{\mathbf{x}}\right)$, where $\Pi_1^{\mathbf{x}}$ denotes the projection of `column 1` onto the sampled output string $\mathbf{x} = \mathbf{x}_1$. Using standard tensor network algorithms, the time complexity of these steps is $L_1 q^{O(d^2)}$.

We next consider `column 2`. At this point, we add the qudits and apply the gates that are in the lightcone of `column 2` but were not applied previously. Denote this isometry by V_2. It is straightforward to see that this step respects causality. That is, if some gate U is in the lightcone of `column 1`, then any gate W that is in the lightcone of `column 2` but not `column 1` cannot be required to be applied before U, because if it were, then it would be in the lightcone of `column 1`. Hence, when we apply gates in this step, we never apply a gate that was required to be applied before some gate that was applied in the first step. After this step, we have applied all gates in the lightcone of `columns (1, 2)`, and we have also projected `column 1` onto the measurement outcomes we observed.

By simulating the measurements of `column 2` in a similar way to those of `column 1`, we sample a string $\mathbf{x}_2$ from the marginal distribution on `column 2`, conditioned on the previously observed outcomes from `column 1`. Each time an isometry V_j is applied, the bond dimension of the MPS representation of the current state will in general increase by a multiplicative factor. In particular, if we iterate this procedure to simulate the entire lattice, we will eventually encounter a maximal bond dimension of up to $q^{O(dL_1)}$ and will obtain a sample $\mathbf{x} = (\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_{L_2}) \in [q]^n$ from the true output distribution.

To improve the efficiency at the expense of accuracy, we may compress the MPS in each iteration to one with smaller bond dimension using standard MPS compression algorithms. In particular, in each iteration j before we apply the corresponding isometry V_j, we first discard as many of the smallest singular values (i.e. Schmidt values) associated with each cut of the MPS as possible up to a total *truncation error* per bond of ϵ, defined as the sum of the squares of the discarded singular values. The bond dimension across any cut is reduced by the number of discarded values. This truncation introduces some error that we quantify below.

If the maximal bond dimension of this truncated version of the simulation algorithm is D, the total runtime of the full algorithm to obtain a sample is bounded by (taking q and d to be constants) $O(nD^3)$ using standard MPS compression algorithms.

We assume that for a specified maximal bond dimension D and truncation error per bond ϵ, if a bond dimension ever exceeds D then the algorithm terminates and outputs a failure flag FAIL. Hence, the runtime of the algorithm when simulating some circuit C with parameters ϵ and D is bounded by $O(nD^3)$, and the algorithm has some probability of failure $p_{f,C}$. We summarize the `SEBD` algorithm in Algorithm 1.

The untruncated version of the algorithm presented above samples from the true distribution $\mathcal{D}_C$ of the measurement outcomes of the original 2D circuit C. However, due to the MPS compression which we perform in each iteration and the possibility of failure, the algorithm incurs some error which causes it to instead sample from some distribution $\mathcal{D}'_C$. Here, we bound the total variation distance between these

Algorithm 1 SEBD

Input: circuit instance C, truncation error ϵ, bond dimension cutoff D
Output: string $\mathbf{x} \in [q]^n$ or FAIL
Runtime: $O(nD^3)$ [q and d assumed to be constants]

initialize an MPS in the state $|1\rangle\langle 1|^{\otimes L_1}$, corresponding to `column 1`
for $t = 1 \dots L_2$ **do**
    compress MPS describing state by truncating small singular values, up to error ϵ per bond
    apply V_t, corresponding to gates in the lightcone of `column` t not yet applied
    if some bond dimension is greater than D, terminate and output FAIL
    simulate measurement of all qudits in via MPS contraction and sampling
    apply $\Pi_t^{\mathbf{x}_t}$ to condition on measurement string $\mathbf{x}_t$ observed for that column
return $(\mathbf{x}_1, \dots, \mathbf{x}_{L_2}) \in [q]^n$

distributions, given by

$$\frac{1}{2}\|\mathcal{D}'_C - \mathcal{D}_C\|_1 = \frac{1}{2}\sum_{\mathbf{x}} |\mathcal{D}'_C(\mathbf{x}) - \mathcal{D}_C(\mathbf{x})| + \frac{1}{2}p_{f,C}, \tag{4.3}$$

where the sum runs over the q^n possible output strings (not including FAIL), in terms of the truncation error made by the algorithm.

We first obtain a very general bound on the error made by `SEBD` with no bond dimension cutoff in terms of the truncation error. Note that the truncation error may depend on the (random) measurement outcomes, and is itself therefore a random variable. See Section 5.4 in the for a proof.

Lemma 21. *Let ϵ_i denote the sum of the squares of all singular values discarded in the compression during iteration i of the simulation of a circuit C with output distribution $\mathcal{D}_C$ by `SEBD` with no bond dimension cutoff, and let Λ denote the sum of all singular values discarded over the course of the algorithm. Then the distribution $\mathcal{D}'_C$ sampled from by `SEBD` satisfies*

$$\frac{1}{2}\|\mathcal{D}'_C - \mathcal{D}_C\|_1 \leq \mathbb{E}\sum_{i=1}^{L_2}\sqrt{2\epsilon_i} \leq \sqrt{2}\,\mathbb{E}\,\Lambda, \tag{4.4}$$

where the expectations are over the random measurement outcomes.

From Lemma 21 we immediately obtain two corollaries. The first is useful for empirically bounding the sampling error in total variation distance made by `SEBD` when the algorithm also has a bond dimension cutoff. The second is a useful asymptotic statement. The corollaries follow straightforwardly from the coupling formulation of variational distance, Markov's inequality, and the triangle inequality.

Corollary 1. *Let $\mathcal{A}$ denote a `SEBD` algorithm with truncation error parameter ϵ and bond dimension cutoff D. Consider a fixed circuit C, and suppose that $\mathcal{A}$ applied to*

this circuit fails with probability $p_{f,C}$. Then $\mathcal{A}$ samples from the output distribution of C with total variation distance error bounded by $L_2\sqrt{2\epsilon L_1} + p_{f,C}$.

If the failure probability of $\mathcal{A}$ averaged over random choice of circuit instance and measurement outcome is p_f, then for any δ, on at least $1-\delta$ fraction of circuit instances, $\mathcal{A}$ samples from the true output distribution with total variation distance error bounded by $L_2\sqrt{2\epsilon L_1} + p_f/\delta$.

In practice, the variational distance error of SEBD with truncation error ϵ applied to the simulation of some circuit C can be bounded by constructing a confidence interval for $p_{f,C}$ and applying the above bound.

Corollary 2. *Let $\mathcal{A}$ denote a SEBD algorithm with truncation error parameter ϵ and no bond dimension cutoff. Suppose that, for some random circuit family with $q = O(1)$ and $d = O(1)$, the expected bond dimension across any cut is bounded by $\mathrm{poly}(n, 1/\epsilon)$. Then, SEBD with some choice of $\epsilon = 1/\mathrm{poly}(n)$ and $D = \mathrm{poly}(n)$ runs in time $\mathrm{poly}(n, 1/\varepsilon, 1/\delta)$ and, with probability at least $1-\delta$ over the choice of circuit instance C, samples from the output distribution of C with variational distance error less than ε.*

Thus, to prove the part of Conjecture 1 about sampling up to total variation distance error ε for uniform random circuit families, it would suffice to show that there is a 2D constant-depth uniform random quantum circuit family with the worst-case-hard property for which the expected bond dimension across any cut while running SEBD with truncation parameter ϵ is bounded by $\mathrm{poly}(n, 1/\epsilon)$. Later, we will introduce two candidate circuit families for which we can give numerical and analytical evidence that this criterion is indeed met.

In the next subsection, we show how the other part of Conjecture 1, regarding computing output probabilities, would also follow from a $\mathrm{poly}(n, 1/\epsilon)$ bound on the bond dimension of states encountered by SEBD on uniform worst-case-hard circuit families.

4.2.2 Computing output probabilities with SEBD

In the previous section, we described how a SEBD algorithm with a truncation error parameter ϵ and a bond dimension cutoff D applied to a circuit C samples from a distribution $\mathcal{D}'_C$ satisfying $\|\mathcal{D}'_C - \mathcal{D}_C\|_1 \leq 2L_2\sqrt{2\epsilon L_1} + 2p_{f,C}$ where $p_{f,C}$ is the probability that some bond dimension exceeds D and the algorithm terminates and indicates failure. Expanding the expression for the 1-norm and rearranging, we have

$$\frac{1}{q^n}\sum_{\mathbf{x}} |\mathcal{D}'_C(\mathbf{x}) - \mathcal{D}_C(\mathbf{x})| \leq \frac{2L_2\sqrt{2\epsilon L_1} + p_{f,C}}{q^n}. \tag{4.5}$$

SEBD with bond dimension cutoff D can be used to compute $\mathcal{D}'_C(\mathbf{x})$ for any output string $\mathbf{x}$ in time $O(nD^3)$ (taking q and d to be constants). To do this, for a fixed output string $\mathbf{x}$, SEBD proceeds similarly to the case in which it's being used for sampling, but rather than sampling from the output distribution of some column, it simply projects

that column onto the outcome specified by the string $\mathbf{x}$, and computes the conditional probability of that outcome via contraction of the MPS. That is, at iteration t, the algorithm computes the conditional probability of measuring the string $\mathbf{x}_t \in [q]^{L_1}$ in `column` t , $\mathcal{D}'_C(\mathbf{x}_t|\mathbf{x}_1, \ldots, \mathbf{x}_{t-1})$, by projecting `column` t onto the relevant string via the projector $\Pi_t^{\mathbf{x}_t}$ and then contracting the relevant MPS. If the bond dimension ever exceeds D, then it must hold that $\mathcal{D}'_C(\mathbf{x}) = 0$, and so the algorithm outputs zero and terminates. Otherwise, the algorithm outputs $\mathcal{D}'_C(\mathbf{x}) = \prod_{t=1}^{L_2} \mathcal{D}'_C(\mathbf{x}_t|\mathbf{x}_1, \ldots, \mathbf{x}_{t-1})$. We summarize this procedure in Algorithm 2.

Algorithm 2 `SEBD` for computing output probabilities

Input: circuit instance C, truncation error ϵ, bond dimension cutoff D, string $\mathbf{x} \in [q]^n$

Output: $\mathcal{D}'_C(\mathbf{x})$

Runtime: $O(nD^3)$ [q and d assumed to be constants]

initialize an MPS in the state $|1\rangle\langle 1|^{\otimes L_1}$, corresponding to `column 1`
for $t = 1 \ldots L_2$ **do**
    compress MPS describing state by truncating small singular values, up to error ϵ per bond
    apply V_t, corresponding to gates in the lightcone of `column` t not yet applied
    if some bond dimension is greater than D, terminate and output zero
    apply $\Pi_t^{\mathbf{x}_t}$ to condition on string $\mathbf{x}_t$
    compute $\mathcal{D}'_C(\mathbf{x}_t|\mathbf{x}_1, \ldots, \mathbf{x}_{t-1})$ via MPS contraction
return $\mathcal{D}'_C(\mathbf{x}) = \prod_{t=1}^{L_2} \mathcal{D}'_C(\mathbf{x}_t|\mathbf{x}_1, \ldots, \mathbf{x}_{t-1})$

We have therefore shown the following.

Lemma 22. *Let $p_{f,C}$ be the failure probability of `SEBD` when used to simulate a circuit instance C with truncation error parameter ϵ and bond dimension cutoff D. Suppose $\mathbf{x} \in [q]^n$ is an output string drawn uniformly at random. Then Algorithm 2 outputs a number $\mathcal{D}'_C(\mathbf{x})$ satisfying*

$$\mathop{\mathbb{E}}_{\mathbf{x}} |\mathcal{D}'_C(\mathbf{x}) - \mathcal{D}_C(\mathbf{x})| \leq \frac{2L_2\sqrt{2\epsilon L_1} + p_{f,C}}{q^n}. \tag{4.6}$$

The above lemma bounds the expected error incurred while estimating a uniformly random output probability for a fixed circuit instance C. We may use this lemma to straightforwardly bound the expected error incurred while estimating the probability of a fixed output string over a distribution of random circuit instances. The corollary is applicable if the distribution of circuit instances has the property of being invariant under an application of a final layer of arbitrary single-qudit gates. This includes circuits in which all gates are Haar-random (as long as every qudit is acted on by some gate), but is more general. In particular, any circuit distribution in which the final gate to act on any given qudit is Haar-random satisfies this property. This fact will be relevant in subsequent sections.

Corollary 3. *Let p_f be the failure probability of* SEBD *when used to simulate a random circuit instance C with truncation error parameter ϵ and bond dimension cutoff D, where C is drawn from a distribution that is invariant under application of a final layer of arbitrary single-qudit gates. Then for any fixed string $\mathbf{x} \in [q]^n$ the output of Algorithm 2 satisfies*

$$\mathop{\mathbb{E}}_{C} |\mathcal{D}'_C(\mathbf{x}) - \mathcal{D}_C(\mathbf{x})| \leq \frac{2L_2\sqrt{2\epsilon L_1} + p_f}{q^n}. \tag{4.7}$$

Proof. Averaging the bound of Equation (4.6) over random circuit instances, we have

$$\mathop{\mathbb{E}}_{\mathbf{y}} \mathop{\mathbb{E}}_{C} |\mathcal{D}'_C(\mathbf{y}) - \mathcal{D}_C(\mathbf{y})| \leq \frac{2L_2\sqrt{2\epsilon L_1} + p_f}{q^n}. \tag{4.8}$$

Let $L_{\mathbf{y}}$ denote a layer of single-qudit gates with the property that $L_{\mathbf{y}} |\mathbf{x}\rangle = |\mathbf{y}\rangle$. By assumption, C is distributed identically to the composition of C with $L_{\mathbf{y}}$, denoted $L_{\mathbf{y}} \circ C$. Together with the observation that $\mathcal{D}_{L_{\mathbf{y}} \circ C}(\mathbf{y}) = \mathcal{D}_C(\mathbf{x})$, we have

$$\mathop{\mathbb{E}}_{\mathbf{y}} \mathop{\mathbb{E}}_{C} |\mathcal{D}'_C(\mathbf{y}) - \mathcal{D}_C(\mathbf{y})| = \mathop{\mathbb{E}}_{\mathbf{y}} \mathop{\mathbb{E}}_{C} |\mathcal{D}'_{L_{\mathbf{y}} \circ C}(\mathbf{y}) - \mathcal{D}_{L_{\mathbf{y}} \circ C}(\mathbf{y})| \tag{4.9}$$

$$= \mathop{\mathbb{E}}_{C} |\mathcal{D}'_C(\mathbf{x}) - \mathcal{D}_C(\mathbf{x})|, \tag{4.10}$$

from which the result follows. □

The following asymptotic statement follows straightforwardly.

Corollary 4. *Let $\mathcal{A}$ denote a* SEBD *algorithm with truncation error parameter ϵ and no bond dimension cutoff. Suppose that, for some random circuit family with $q = O(1)$ and $d = O(1)$, the expected bond dimension across any cut is bounded by $\mathrm{poly}(n, 1/\epsilon)$. Then,* SEBD *with some choice of $\epsilon = 1/\mathrm{poly}(n)$ and $D = \mathrm{poly}(n)$ runs in time $\mathrm{poly}(n, 1/\varepsilon, 1/\delta)$ and, with probability at least $1 - \delta$ over the choice of circuit instance C, estimates $\mathcal{D}_C(\mathbf{x})$ for some fixed $\mathbf{x} \in [q]^n$ up to additive error bounded by ε/q^n.*

Corollary 4 shows how the part of Conjecture 1 about computing arbitrary output probabilities to error ε/q^n would follow from a bound on the bond dimension across any cut when SEBD runs on a uniform worst-case-hard circuit family.

4.2.3 Example: SEBD applied to cluster state with Haar-random measurements (CHR)

To illustrate the connection between SEBD and random-unitary-and-measurement dynamics, we now study the SEBD algorithm in more detail for a simple uniform family of 2D random circuits that possesses the worst-case-hard property required by Conjecture 1. The model we consider is the following: start with a 2D cluster state of n qubits arranged in a $\sqrt{n} \times \sqrt{n}$ grid, apply a single-qubit Haar-random gate to each qubit, and then measure all qubits in the computational basis. Recall that a cluster

state may be created by starting with the product state $|+\rangle^{\otimes n}$ before applying CZ gates between all adjacent sites. An equivalent formulation which we will find convenient in the subsequent section is to measure each qubit of the cluster state in a Haar-random basis. We refer to this model as CHR, for "cluster state with Haar-random measurements".

Following [BJS10], it is straightforward to show that sampling from the output distribution of CHR is classically *worst-case hard* assuming the polynomial hierarchy (PH) does not collapse to the third level. It can also be readily shown, following [Bou+19], that near-exactly computing output probabilities of CHR is #P-hard in the average case. These results rule out, under standard conjectures, the existence of a classical sampling algorithm for CHR that succeeds for all instances, or a classical algorithm for efficiently computing most output probabilities of CHR near-exactly. A natural question is then whether efficient approximate average-case versions of these algorithms may exist. We formalize these questions as the problems $\mathtt{CHR}_{\pm}^{\mathtt{samp/prob}}$.

Problem 1 ($\mathtt{CHR}_{\pm}^{\mathtt{samp/prob}}$)**.** *Given as input a random instance C of CHR (specified by a sidelength $\sqrt{n}$ and a set of n single-qubit Haar-random gates applied to the $\sqrt{n} \times \sqrt{n}$ cluster state) and error parameters ε and δ, perform the following computational task in time* $\mathrm{poly}(n, 1/\varepsilon, 1/\delta)$.

- $\mathit{CHR}_{\pm}^{\mathit{samp}}$. *Sample from a distribution $\mathcal{D}'_C$ that is ε-close in total variation distance to the true output distribution $\mathcal{D}_C$ of circuit C, with probability of success at least $1 - \delta$ over the choice of measurement bases.*

- $\mathit{CHR}_{\pm}^{\mathit{prob}}$. *Estimate $\mathcal{D}_C(\mathbf{0})$, the probability of obtaining the all-zeros string upon measuring the output state of C in the computational basis, up to additive error at most $\varepsilon/2^n$, with probability of success at least $1 - \delta$ over the choice of measurement bases.*

We now show that SEBD solves $\mathtt{CHR}_{\pm}^{\mathtt{samp/prob}}$ if a certain form of 1D dynamics involving local unitary gates and measurements is classically simulable. We first consider the sampling variant of SEBD. Specializing to the CHR model, the algorithm takes on a particularly simple form due to the fact that the cluster state is built by applying CZ gates between all neighboring pairs of qubits, which are initialized in $|+\rangle$ states. Due to this structure, the radius of the lightcone for this model is simply one. In particular, the only gates in the lightcone of `columns 1-j` are the Haar-random single-qubit gates acting on qubits in these columns, as well as CZ gates that act on at least one qubit within these columns. This permits a simple prescription for SEBD applied to this problem.

Initialize the simulation algorithm in the state $\rho_1 = |+\rangle\langle+|^{\otimes\sqrt{n}}$ corresponding to `column 1`. To implement the isometry V_1, initialize the qubits of `column 2` in the state $|+\rangle\langle+|^{\otimes\sqrt{n}}$ and apply CZ gates between adjacent qubits that are both in `column 1` and between adjacent qubits in separate columns. Now, measure the qubits of `column 1` in the specified Haar-random bases (equivalently, apply the specified Haar-random gates and measure in the computational basis), inducing a pure state ρ_2 with

support in `column 2`. Iterating this process, we progress through a random sequence of 1D states on $\sqrt{n}$ qubits $\rho_1 \to \rho_2 \to \cdots \to \rho_{\sqrt{n}}$ which we will see can be equivalently understood as arising from a 1D dynamical process consisting of alternating layers of random unitary gates and weak measurements.

It will be helpful to introduce notation. Define $|\theta, \phi\rangle := \cos\left(\frac{\theta}{2}\right)|0\rangle + e^{i\phi}\sin\left(\frac{\theta}{2}\right)|1\rangle$. In other words, let $|\theta, \phi\rangle$ denote the single-qubit pure state with polar angle θ and azimuthal angle ϕ on the Bloch sphere. Let $\theta_i^{(t)}$ and $\phi_i^{(t)}$ specify the measurement basis of the qubit in row i and column t; that is, the projective measurement on the qubit in row i and column t is $\{\Pi^0_{\theta_i^{(t)},\phi_i^{(t)}}, \Pi^1_{\theta_i^{(t)},\phi_i^{(t)}}\}$ with $\Pi^0_{\theta_i^{(t)},\phi_i^{(t)}} := \left|\theta_i^{(t)}, \phi_i^{(t)}\right\rangle\!\left\langle\theta_i^{(t)}, \phi_i^{(t)}\right|$ and $\Pi^1_{\theta_i^{(t)},\phi_i^{(t)}} := I - \Pi^0_{\theta_i^{(t)},\phi_i^{(t)}}$. We also define

$$M_0(\theta, \phi) := \begin{pmatrix} \cos(\theta/2) & 0 \\ 0 & e^{-i\phi}\sin(\theta/2) \end{pmatrix} \tag{4.11a}$$

$$M_1(\theta, \phi) := \begin{pmatrix} \sin(\theta/2) & 0 \\ 0 & e^{i\phi}\cos(\theta/2) \end{pmatrix}. \tag{4.11b}$$

Note that $\{M_0(\theta, \phi), M_1(\theta, \phi)\}$ defines a weak single-qubit measurement. We now describe, in Algorithm 3, a 1D process which we claim produces a sequence of states identical to that encountered by `SEBD` for the same choice of measurement bases and measurement outcomes, and also has the same measurement statistics.

Algorithm 3 Effective 1D dynamics of a fixed instance of `CHR`

$\varphi_1 \leftarrow |+\rangle\langle+|^{\otimes\sqrt{n}}$.
for $t = 1 \ldots \sqrt{n} - 1$ **do**
    apply a CZ gate between every adjacent pair of qubits
    measure $\{M_0(\theta_i^{(t)}, \phi_i^{(t)}), M_1(\theta_i^{(t)}, \phi_i^{(t)})\}$ on qubit i, obtaining $X_i^{(t)}$, for $i \in [\sqrt{n}]$
    apply a Hadamard transform
    $\varphi_{t+1} \leftarrow$ resulting state
measure $\left\{\Pi^0_{\theta_i^{(\sqrt{n})},\phi_i^{(\sqrt{n})}}, \Pi^1_{\theta_i^{(\sqrt{n})},\phi_i^{(\sqrt{n})}}\right\}$ on qubit i, obtaining $X_i^{(\sqrt{n})}$, for $i \in [\sqrt{n}]$

Lemma 23. *For a fixed choice of $\{\theta_i^{(t)}, \phi_i^{(t)}\}$ parameters, the joint distribution of outcomes $\{X_i^{(t)}\}_{i,t}$ is identical to that of $\{Y_i^{(t)}\}_{i,t}$, where $\{Y_i^{(t)}\}_{i,t}$ are the measurement outcomes obtained upon measuring all qubits of a $\sqrt{n} \times \sqrt{n}$ cluster state, with the measurement on the qubit in row i and column t being $\{\Pi^0_{\theta_i^{(t)},\phi_i^{(t)}}, \Pi^1_{\theta_i^{(t)},\phi_i^{(t)}}\}$. Furthermore, for any fixed choice of measurement outcomes, $\varphi_j = \rho_j$ for all $j \in [\sqrt{n}]$, where ρ_j is the state at the beginning of iteration j of the `SEBD` algorithm.*

Proof. The lemma follows from the above description of the behavior of `SEBD` applied to `CHR`, as well as the following identities holding for any single-qubit state $|\xi\rangle$ which

may be verified by straightforward calculation:

$$(\Pi^0_{\theta,\phi} \otimes I) CZ(|\xi\rangle \otimes |+\rangle) = |\theta, \phi\rangle \otimes HM_0(\theta, \phi) |\xi\rangle \tag{4.12}$$

$$(\Pi^1_{\theta,\phi} \otimes I) CZ(|\xi\rangle \otimes |+\rangle) = |\pi - \theta, -\phi\rangle \otimes HM_1(\theta, \phi) |\xi\rangle . \tag{4.13}$$

□

We have seen that, for a fixed choice of single-qubit measurement bases $\{\theta_j^{(t)}, \phi_j^{(t)}\}_{t,j}$ associated with an instance C, we can define an associated 1D process consisting of alternating layers of single-qubit weak measurements and local unitary gates, such that simulating this 1D process is sufficient for sampling from $\mathcal{D}_C$.

Now, recall that in the context of simulating CHR, each single-qubit measurement basis is chosen randomly according to the Haar measure. That is, the Bloch sphere angles $(\theta_i^{(t)}, \phi_i^{(t)})$ are Haar-distributed. If we define $x_i^{(t)} \equiv \cos\theta_i^{(t)}$, we find that $x_i^{(t)}$ is uniformly distributed on the interval $[-1, 1]$. The parameters $\phi_i^{(t)}$ are uniformly distributed on $[0, 2\pi]$. Using these observations, as well as the observation that the outcome probabilities of the measurement of qubit i in iteration t are independent of the azimuthal angle $\phi_i^{(t)}$ when $t < \sqrt{n}$, we may derive effective dynamics of a random instance.

Define the operators

$$N(x) := \begin{pmatrix} \sqrt{\frac{1+x}{2}} & 0 \\ 0 & \sqrt{\frac{1-x}{2}} \end{pmatrix}, \quad x \in [-1, 1].$$

Note that $\{N(x), N(-x)\}$ defines a weak measurement. Also, define the phase gate

$$P(\phi) := \begin{pmatrix} 1 & 0 \\ 0 & e^{i\phi} \end{pmatrix}, \quad \phi \in [0, 2\pi].$$

By randomizing each single-qubit measurement basis according to the Haar distribution, one finds that the dynamics of Algorithm 3 (which applies for a fixed choice of measurement bases) may be written as Algorithm 4 below, where the notation $x \in_U [-1, 1]$ means that x is a random variable uniformly distributed on $[-1, 1]$. That is, the distribution of random sequences $\varphi_1 \to \varphi_2 \to \cdots \to \varphi_{\sqrt{n}}$ and distribution of output statistics produced by Algorithm 4 is identical to that produced by SEBD applied to CHR.

Hence, if TEBD can efficiently simulate the process of Algorithm 4 with high probability, then SEBD can solve $\mathtt{CHR}_{\pm}^{\mathtt{samp}}$ and $\mathtt{CHR}_{\pm}^{\mathtt{prob}}$. We formalize this in the following lemma.

Lemma 24. *Suppose that TEBD can efficiently simulate the process described in Algorithm 4 in the sense that the expected bond dimension across any cut is bounded by* $\mathrm{poly}(n, 1/\epsilon)$ *where* ϵ *is the truncation error parameter. Then SEBD can be used to solve* $\mathtt{CHR}_{\pm}^{\mathtt{samp}}$ *and* $\mathtt{CHR}_{\pm}^{\mathtt{prob}}$.

Algorithm 4 Effective 1D dynamics of `CHR`

$\varphi_1 \leftarrow |+\rangle\langle+|^{\otimes\sqrt{n}}$.
for $t = 1 \ldots \sqrt{n} - 1$ **do**
    apply a CZ gate between every adjacent pair of qubits
    for $i = 1 \ldots \sqrt{n}$ **do**
        measure $\{N(x), N(-x)\}$ on qubit i with $x \in_U [-1, 1]$
        apply the gate $P(\phi)$ with $\phi \in_U [0, 2\pi]$ to qubit i
    apply a Hadamard transform
    $\varphi_{t+1} \leftarrow$ resulting state
perform a projective measurement on each qubit in a Haar-random basis

Proof. Follows from Corollary 2, Corollary 4, and the equivalence to Algorithm 4 discussed above. □

We have shown how `SEBD` applied to `CHR` can be reinterpreted as `TEBD` applied to a 1D dynamical process involving alternating layers of random unitaries and weak measurements. Up until this point, there has been little reason to expect that `SEBD` is efficient for the simulation of `CHR`. In particular, with no truncation, the bond dimension of the MPS stored by the algorithm grows exponentially as the algorithm sweeps across the lattice.

We now invoke the findings of a number of related recent works [LCF18; Cha+19; SRN19; LCF19; SRS19; Cho+20; GH20a; BCA20; Jia+20; GH20b; Zab+20; TZ20; NS20; AB20; Fan+20; Li+20; LAB21; SH20; Ipp+21; FA20; SRS20; Vij20; LP20; LF20; TFD20; FHH21; Nah+21; IK21]. to motivate the possibility that `TEBD` can efficiently simulate the effective 1D dynamics. These works study various 1D dynamical processes involving alternating layers of measurements and random local unitaries. In some cases, the measurements are considered to be projective and only occur with some probability p. In other cases, similarly to Algorithm 4, weak measurements are applied to each site with probability one. The common finding of these papers is that such models appear to exhibit an entanglement phase transition driven by measurement probability p (in the former case), or measurement strength (in the latter case). On one side of the transition, the entanglement entropy obeys an area law, scaling as $O(1)$ with the length L. On the other side, it obeys a volume law, scaling as $O(L)$.

Based on these works, one expects the entanglement dynamics to saturate to an area-law or volume-law phase. And in fact, our numerical studies (presented in Section 4.4) suggest that these dynamics saturate to an area-law phase. The common intuition that 1D quantum systems obeying an area law for the von Neumann entropy are easy to simulate with matrix product states therefore suggests that `SEBD` applied to this problem is efficient. While counterexamples to this common intuition are known [Sch+08], they are contrived and do not present an obvious obstruction for our algorithm. To better understand the relationship between maximal bond dimension and truncation error when the effective dynamics is in the area-law phase as well as rule out such counterexamples, in the following section we describe a toy model for a unitary-and-measurement process in the area-law phase, which predicts

a superpolynomial decay of Schmidt values across any cut and therefore predicts that a polynomial runtime is sufficient to perform the simulation to $1/\operatorname{poly}(n)$ error. Our numerical results (presented in Section 4.4) suggest that the effective dynamics of the random circuit architectures we consider are indeed in the area-law phase, with entanglement spectra consistent with those predicted by the toy model dynamics. Further analytical evidence for efficiency is given in Section 4.5.

Note that, although we explicitly derived the effective 1D dynamics for the `CHR` model and observed it to be a simple unitary-and-measurement process, the interpretation of the effective 1D dynamics as a unitary-and-measurement process is not specific to `CHR` and is in fact general. In the general case, `SEBD` tracks $O(r)$ columns simultaneously where r is the radius of the lightcone corresponding to the circuit. In each iteration, new qudits that have come into the lightcone are added, unitary gates that have come into the lightcone are performed, and finally projective measurements are performed on a single column of qudits. Similarly to the case of `CHR`, this entire procedure can be viewed as an application of unitary gates followed by weak measurements on a 1D chain of qudits of dimension $q^{O(r)}$. Intuitively, increasing the circuit depth corresponds both to increasing the local dimension in the effective 1D dynamics and decreasing the measurement strength. The former is due to the fact that in general the lightcone radius r will increase as depth is increased, and the local dimension of the effective dynamics is $q^{O(r)}$. The latter is due to the fact that as r increases, the number of tracked columns increases but the number of measured qudits in a single round stays constant. Hence the fraction of measured qudits decreases, and intuitively we expect this to correspond to a decrease in effective measurement strength. This intuition together with the findings of prior works on unitary-and-measurement dynamics suggests that the effective dynamics experiences an entanglement phase transition from an area-law to volume-law phase as q or d is increased, and therefore `SEBD` experiences a computational phase transition, supporting Conjecture 2. While this analogy is not perfect, we provide further analytical evidence in Section 4.5 that the effective 1D dynamics indeed undergoes such a phase transition.

4.2.4 Conjectured entanglement spectrum of unitary-and-measurement dynamics in an area-law phase

Numerical (Section 4.4) and analytical (Section 4.5) evidence suggests that the effective 1D dynamics corresponding to the uniform 2D shallow random circuit families we consider are in the area-law phase, making efficient simulation via `SEBD` very plausible. However, it is desirable to have clear predictions for the scaling of the entanglement spectra for states of the effective 1D dynamics, as this allows us to make concrete predictions for error scaling of `SEBD` and rule out (contrived) examples of states [Sch+08] which cannot be efficiently represented via MPS despite obeying an area law for the von Neumann entanglement entropy.

To this end, we study a simple toy model of how entanglement might scale in the area-law phase of a unitary-and-measurement circuit. Consider a chain of n qubits where we are interested in the entanglement across the cut between $1, \ldots, n/2$ and

$n/2+1, \ldots, n$ (assume n is even). We model the dynamics as follows. In each time step we perform the following three steps:

1. Set the state of sites $n/2$ and $n/2+1$ to be an EPR pair $|\Phi\rangle = (|00\rangle + |11\rangle)/\sqrt{2}$.

2. Perform the cyclic permutations of qubits $(n/2, n/2-1, \ldots, 1)$ and $(n/2+1, n/2+2, \ldots, n)$. That is, move each qubit one step away from the central cut, except for qubits 1 and n, which are moved to $n/2$ and $n/2+1$ respectively.

3. Perform a weak measurement on each qubit with Kraus elements $M_0(\theta) = \cos(\theta/2)\,|0\rangle\langle 0| + \sin(\theta/2)\,|1\rangle\langle 1|$ and $M_1(\theta) = \sin(\theta/2)\,|0\rangle\langle 0| + \cos(\theta/2)\,|1\rangle\langle 1|$. This is based on Equation (4.11), but the phases will not matter here so we have dropped them for simplicity.

Without the measurements this would create one EPR pair in each time step until the system had $n/2$ EPR pairs across the cut after time $n/2$. However, the measurements have the effect of reducing the entanglement. For this process, we derive the functional form of the asymptotic scaling of half-chain Schmidt coefficients $\lambda_1 \geq \lambda_2 \geq \cdots$. Moreover, bounds on the scaling of the entanglement spectrum allows us to derive a relation between the truncation error (sum of squares of discarded Schmidt values) ϵ incurred upon discarding small Schmidt values, and the rank r of the post-truncation state. The bounds are given in the following lemma, which is proved in Section 5.4.

Lemma 25. *Let $\lambda_1 \geq \lambda_2 \geq \cdots$ denote the half-chain Schmidt values after at least $n/2$ iterations of the toy model process. Then with probability at least $1-\delta$ the half-chain Schmidt values indexed by $i \geq i^* = \exp\Big(\Theta(\sqrt{\log(n/\delta)})\Big)$ obey the asymptotic scaling*

$$\lambda_i \propto \exp\big(-\Theta(\log^2(i))\big). \tag{4.14}$$

Furthermore, upon truncating the smallest Schmidt coefficients up to a truncation error of ϵ, with probability at least $1-\delta$, the half-chain Schmidt rank r of the post-truncation state obeys the scaling

$$r \leq \exp\Big(\Theta\Big(\sqrt{\log(n/\epsilon\delta)}\Big)\Big). \tag{4.15}$$

This is the basis for our Conjecture 1'. More precisely, we take this analysis as evidence that the bond dimension D, truncation error ϵ, and system size n obey the scaling $D \leq \exp\Big(\Theta\Big(\sqrt{\log(n/\epsilon\delta)}\Big)\Big)$ with probability $1-\delta$ over random circuit instance and random measurement outcomes when `SEBD` simulates a random constant-depth 2D circuit whose effective 1D dynamics lie in the area-law phase. Recalling that the runtime of `SEBD` scales like $O(nD^3)$ for a maximal bond dimension of D and using the relationship between truncation error, failure probability, variational distance error, and simulable circuit fraction given in Corollary 1, we conclude that `SEBD` with a maximal bond dimension cutoff scaling as $\exp\Big(\Theta\Big(\sqrt{\log(n/\epsilon\delta)}\Big)\Big)$ runs

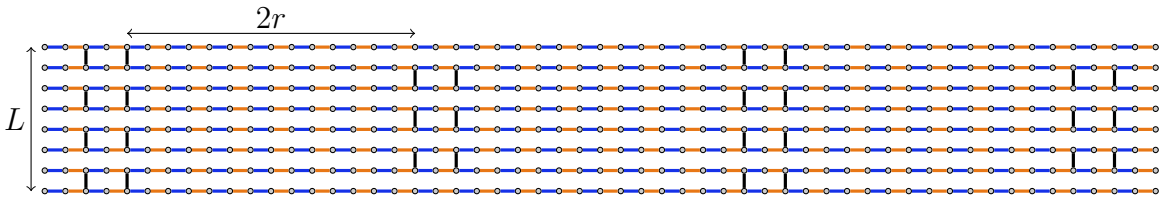


Figure 4-3: Extended brickwork architecture with n qubits. Here, circles represent qubits initialized in the state $|0\rangle^{\otimes n}$, blue lines represent the first layer of gates to act, orange lines represent the second layer, and black lines represent the third and final layer. All gates are chosen Haar-randomly. We let $\texttt{Brickwork}(L, r, v)$ denote the corresponding random circuit with circuit layout depicted in the figure above with vertical sidelength L, "extension parameter" $2r$ (which gives the distance between vertical gates acting on adjacent pairs of rows), and number of pairs of columns of vertical gates v. In the above example, $r = 7$ and $v = 4$. The standard brickwork architecture corresponds to $r = 1$. Note that $n = \Theta(Lrv)$.

in time $n^{1+o(1)} \exp\left(\Theta\left(\sqrt{\log(1/\varepsilon\delta)}\right)\right)$ and simulates $1 - \delta$ fraction of random circuit instances up to variational distance error ε.

It is important to note what this heuristic argument leaves out. While a 1D unitary-and-measurement circuit will indeed create $O(1)$ ebits across any given cut in each round, these will not remain in the form of distinct pairs of qubits. The unitary dynamics *within* each side of the cut will have the effect of transforming the Schmidt bases into entangled ones. This will make the measurements less effective at reducing the entanglement, for reasons that can be understood in terms of quantum state merging [HOW07; Cho+20]. Another simplification of the toy model is that the measurement angle θ is taken to be a fixed constant rather than random. Finally, in the toy model we assume for simplicity that the EPR pairs move cyclically. We expect that, if this effect is significant, it is more likely to make the toy model overly pessimistic compared with the real situation. Despite these simplifications, we believe this model is qualitatively accurate in the area-law phase. Indeed, the scaling of Schmidt values predicted by our toy model analysis is consistent with the scaling we find numerically in Figure 4-5.

4.3 Rigorous analysis of SEBD for the "extended brickwork architecture"

In this section, we show that `SEBD` is provably efficient for certain random circuit families that are worst-case hard. We define the circuit architecture in Figure 4-3. It follows readily from prior works that exactly sampling from the output distribution of this random circuit family for arbitrary circuit instances or near-exactly computing a specific output probability with high probability is classically hard under standard complexity theoretic assumptions. We summarize these observations in the following lemma.

Lemma 26. *Let $r(L)$ and $v(L)$ be any polynomially bounded functions, with $v(L) \geq L^a$ for some $a > 0$. Suppose that there exists a classical algorithm that runs in time* $\mathrm{poly}(n)$ *and samples from the output distribution of an arbitrary realization of* $\texttt{Brickwork}(L, r(L), v(L))$*, as defined in Figure 4-3. Then the polynomial hierarchy collapses to the third level. Suppose there exists a classical algorithm that runs in time* $\mathrm{poly}(n, 1/\delta)$ *and, for an arbitrary fixed output string* $\mathbf{x}$*, with probability at least $1 - \delta$ over choice of random instance, computes the output probability of $\mathbf{x}$ up to additive error $2^{-\tilde{\Theta}(n^2)}$. Then there exists a probabilistic polynomial-time algorithm for computing a #P-hard function.*

Proof. We first note that $\texttt{Brickwork}(L, r(L), v(L))$ supports universal MBQC, in the sense that a specific choice of gates can create a resource state that is universal for MBQC. This is an immediate consequence of the proof of universality of the "standard" brickwork architecture (corresponding to $r = 1$) proved in [BFK09]. Indeed, when using the extended brickwork architecture for MBQC, measurements on the long 1D stretches of length $2r$ may be chosen such that the effective state is simply teleported to the end when computing from left to right (i.e., measurements may be chosen such that the long 1D segments simply amount to applications of identity gates on the effective state). The scaling $v \geq L^a$ ensures that MBQC with an extended brickwork resource state suffices to simulate any BQP computation with polynomial overhead. Since a specific choice of gates creates a resource state for universal MBQC, an algorithm that can simulate an arbitrary circuit realization can be used to simulate arbitrary single-qubit measurements on a resource state universal for MBQC. Under post-selection, such an algorithm can therefore simulate PostBQP [RB01] and hence cannot be efficiently simulated classically unless the polynomial hierarchy collapses to the third level [BJS10].

Similarly, for some subsets of instances, it is #P-hard to compute the output probability of an arbitrary string, since (by choosing gates to create a resource state for universal MBQC) this would allow one to compute output probabilities of universal polynomial-size quantum circuit families which is known to be #P-hard. The result of [Mov19] is then applicable, which implies that if the gates are chosen Haar-randomly, efficiently computing the output probability of some fixed string with probability $1 - 1/\mathrm{poly}(n)$ over the choice of instance up to additive error bounded by $2^{-\tilde{\Theta}(n^3)}$ implies the ability to efficiently compute a #P-hard function with high probability. □

Our goal is to prove that SEBD can efficiently approximately simulate the extended brickwork architecture in the average case for choices of extension parameters for which the above hardness results apply. To this end, we first show a technical lemma which describes how measurements destroy entanglement in 1D shallow random circuits. In particular, given a 1D state generated by a depth-2 Haar-random circuit acting on qubits, after measuring some contiguous region of spins B, the expected entanglement entropy of the resulting post-measurement pure state across a cut going through B is exponentially small in the length of B. We defer the proof to Section 5.4.

Lemma 27. *Suppose a 1D random circuit C is applied to qubits $\{1, \ldots, n\}$ consisting of a layer of 2-qubit Haar-random gates acting on qubits $(k, k+1)$ for odd $k \in \{1, \ldots, n-1\}$, followed by a layer of 2-qubit Haar-random gates acting on qubits $(k, k+1)$ for even $k \in \{1, \ldots, n-1\}$. Suppose the qubits of region $B := \{i, i+1, \ldots, j\}$ for $j \geq i$ are measured in the computational basis, and the outcome b is obtained. Then, letting $|\psi_b\rangle$ denote the post-measurement pure state on the unmeasured qubits, and letting $A := \{1, 2, \ldots, i-1\}$ denote the qubits to the left of B,*

$$\mathbb{E}\, S(A)_{\psi_b} \leq c^{|B|} \tag{4.16}$$

for some universal constant $c < 1$, where the expectation is over measurement outcomes and choice of random circuit C.

We now outline the argument for why `SEBD` should be efficient for the extended brickwork architecture for sufficiently large extension parameters; full details may be found in Section 5.4. During the evolution of `SEBD` as it sweeps from left to right across the lattice, it periodically encounters long stretches of length $2r$ in which no vertical gates are applied. We call these "1-local regions" since the maps applied in the corresponding effective 1D dynamics are 1-local when the algorithm is in such a region. Hence, in the effective 1D dynamics, no 2-qubit maps are applied and therefore the bond dimension of the associated MPS cannot increase during these stretches. It turns out that in 1-local regions, not only does the bond dimension needed to represent the state not increase, but it in fact rapidly decays in expectation. If r is sufficiently large, then the effective 1D state at the end of the 1-local region is very close to a product state with high probability, regardless of how entangled the state was before the region. Hence, when `SEBD` compresses the MPS describing the effective state at the end of the region, it may compress the bond dimension of the MPS to some fixed constant with very small incurred error. The two-qubit maps that are performed in-between 1-local regions only increase the bond dimension by a constant factor. Hence, with high probability, `SEBD` can use a $O(1)$ maximal bond dimension cutoff and simulate a random circuit with extended brickwork architecture with high probability. More precisely, it turns out that the scaling $r \geq \Theta(\log(n))$ is sufficient to guarantee efficient simulation with this argument. A more precise statement of the efficiency of `SEBD` for this architecture is given in the below lemma, whose proof may be found in Section 5.4.

Lemma 28. *Let C be an instance of `Brickwork`(L, r, v). Then, with probability at least $1 - 2^{-\Theta(r)}$ over the circuit instance, `SEBD` running with maximal bond dimension cutoff $D = \Theta(1)$ and truncation error parameter $\epsilon = 2^{-\Theta(r)}$ can be used to (1) sample from the output distribution of C up to error $n2^{-\Theta(r)}$ in variational distance and (2) compute the output probability of an arbitrary output string up to additive error $n2^{-\Theta(r)}/2^n$ in runtime $\Theta(n)$.*

With an appropriate choice of $r = \Theta(\log(L))$, the above result implies that `SEBD` can perform the simulation with error $1/\operatorname{poly}(n)$ for at least $1 - 1/\operatorname{poly}(n)$ fraction of instances. Similarly, choosing r to be a sufficiently large polynomial in L, `SEBD` can

perform the simulation with error $2^{-n^{1-\delta}}$ for $1 - 2^{-n^{1-\delta}}$ fraction of instances, for any constant $\delta > 0$. We summarize these observations as the following corollary.

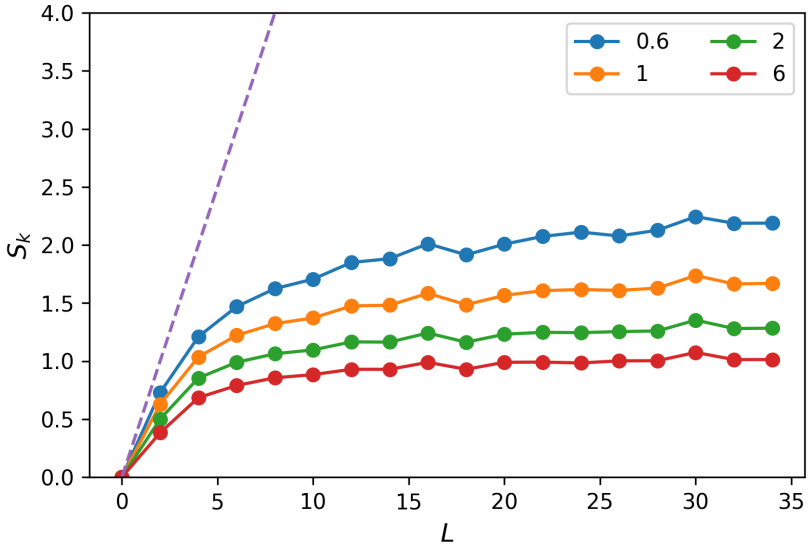
Corollary 5. *For any choice of polynomially bounded v, p_1, p_2, for any sufficiently large constant c `SEBD` can simulate $1 - 1/p_1(n)$ fraction of instances of `Brickwork`$(L, \lceil c \log(L) \rceil, v(L))$ up to error $\varepsilon \leq 1/p_2(n)$ in time $O(n)$. For any choice of $\delta > 0$ and $v(L) \leq \text{poly}(L)$, for any sufficiently large constant c `SEBD` can simulate $1 - 2^{-n^{1-\delta}}$ fraction of instances of `Brickwork`$(L, \lceil L^c \rceil, v(L))$ up to error $\varepsilon \leq 2^{-n^{1-\delta}}$ in time $O(n)$. Here, "simulate with error ε" implies the ability to sample with variational distance error ε and compute the output probability of some fixed string $\mathbf{x}$ with additive error $\varepsilon/2^n$.*

4.4 Numerical results

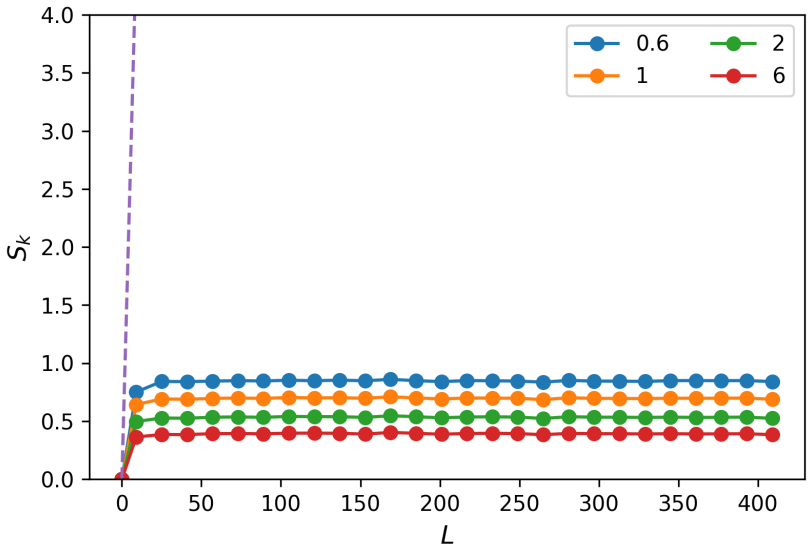
We implemented[3] `SEBD` on two families of random circuits: one consisting of depth-3 random circuits defined on a "brickwork architecture" consisting of three layers of two-qubit Haar-random gates (Figure 4-3 with parameter $r = 1$), and the other being the random circuit family obtained by applying single-qubit Haar-random gates to all sites of a cluster state — we referred to this problem as `CHR` previously. Note that the former architecture has depth three (not including the measurement layer) and the latter has depth four, and both architectures support universal measurement-based quantum computation [BFK09], meaning they have the worst-case-hard property relevant for Conjecture 1. We did not implement `Patching`, due to its larger overhead.

Implementing `SEBD` on a standard laptop, we could simulate typical instances of the 409×409 brickwork model with truncation error 10^{-14} per bond with a runtime on the order of one minute per sample, and typical instances of the 34×34 `CHR` model with truncation error 10^{-10} per bond with a runtime on the order of five minutes per sample (these truncation error settings correspond to sampling errors of less than 0.01 in variational distance as derived previously in Section 4.2). We in fact simulated instances of `CHR` with grid sizes as large as 50×50, although due to the significantly longer runtime for such instances we did not perform large numbers of trials for these cases. In the case of the 409×409 brickwork model, performing over 3000 trials (consisting of generating a random circuit instance and generating a sample from its output distribution using a truncation error of 10^{-14}) and finding no trials for which the bond dimension became large enough for the algorithm to fail, then with 95% confidence, we may conclude that the probability that a random trial fails, p_f, is less than 0.001. Using the bound derived in Section 4.2, we can therefore conclude with 95% confidence that for greater than a 0.9 fraction of 409×409 circuit instances, we can sample from that circuit instance's output distribution with variational distance error less than 0.01. Intuitively, we expect the true simulable fraction to be much larger than this statistical guarantee, as it appears that the entanglement in the effective 1D dynamics only grows extensively for highly structured instances. Note

[3]The code for our implementation is available at https://github.com/random-shallow-2d/random-shallow-2d.



(a) CHR



(b) Brickwork

Figure 4-4: Rényi half-chain entanglement entropies S_k versus sidelength L in the effective 1D dynamics for the CHR and brickwork models, after 80 (resp. 550) iterations. Each point represents the entanglement entropy averaged over 50 random circuit instances, and over the final 10 (resp. 50) iterations for the CHR (resp. brickwork) model. Dashed lines depict the half-chain entanglement entropy scaling of a maximally entangled state, which can be created with a "worst-case" choice of gates for both architectures. The maximal truncation error per bond ϵ was 10^{-10} for CHR and 10^{-14} for the brickwork model.

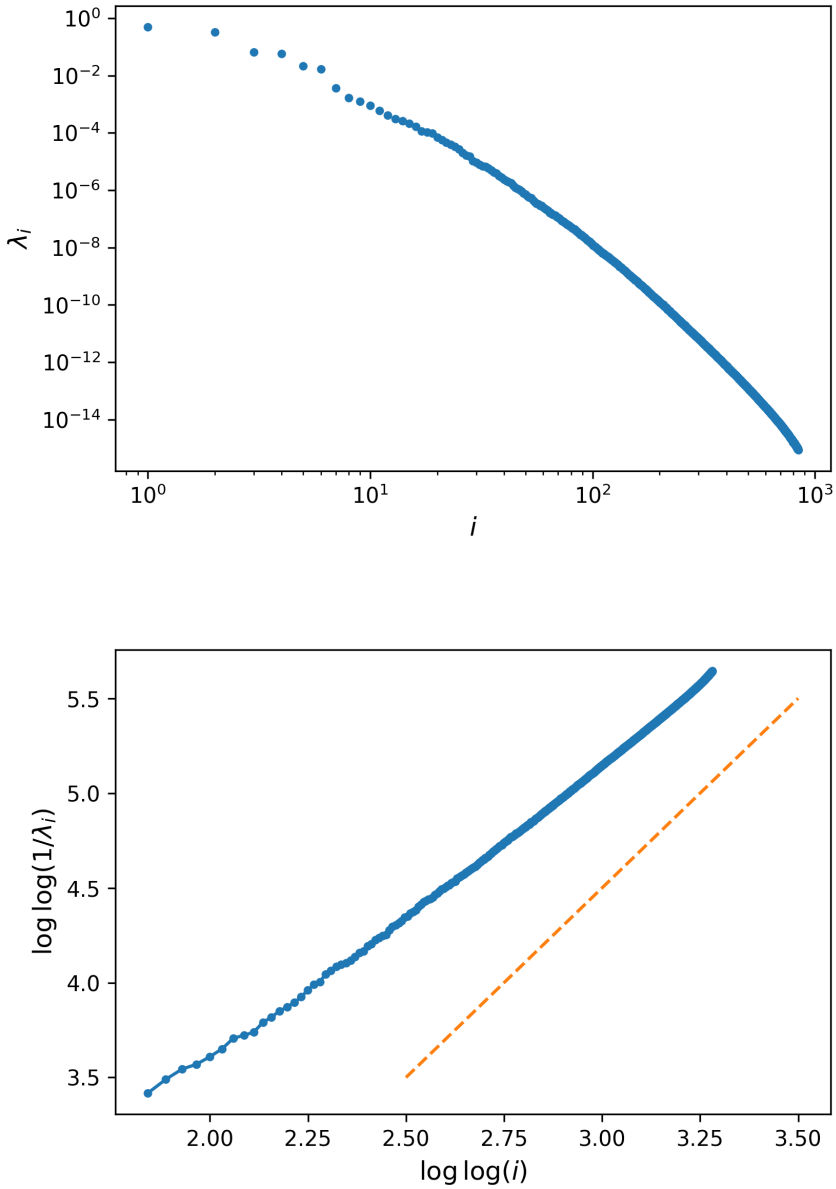


Figure 4-5: Typical half-chain entanglement spectrum $\lambda_1 \geq \lambda_2 \geq \ldots$ observed during the effective 1D dynamics of CHR. These plots were generated from an instance with sidelength $L = 44$ after running for 44 iterations, with squared Schmidt values smaller than approximately 10^{-15} truncated. The left figure shows a spectrum of half-chain eigenvalues. The downward curvature in the log-log scale indicates superpolynomial decay. The right figure displays the same data (minus the few largest values) on a loglog-loglog scale. The toy model predicts that the blue curve asymptotes to a straight line with slope two in the right figure, illustrated by the dashed orange line, corresponding to scaling like $\lambda_i \sim 2^{-\Theta(\log^2(i))}$. The plot is qualitatively consistent with this prediction. The spectrum for the brickwork model decays too quickly to obtain as useful statistics without going to much higher numerical precision.

that for both models, the runtime for a fixed truncation error was qualitatively highly concentrated around the mean. We expect that substantially larger instances of both random circuit families could be quickly simulated with more computing power, although 409×409 simulation of the brickwork architecture is already far beyond what could have been achieved by previous simulation methods that we are aware of.

To make this more precise, it is useful to compare our observed runtime with what is possible by previously known methods. The previously best-known method that we are aware of for computing output probabilities for these architectures would be to write the circuit as a tensor network and perform the contraction of the network [Vil+20]. The cost of this process scales exponentially in the tree-width of a graph related to the quantum circuit, which for a 2D circuit is thought to scale roughly as the surface area of the minimal cut slicing through the circuit diagram, as in Eq. (4.1). By this reasoning, we estimate that simulating a circuit with brickwork architecture on a 400×400 lattice using tensor network contraction would be roughly equivalent to simulating a depth-40 circuit on a 20×20 lattice with the architecture considered in [Vil+20], where the entangling gates are CZ gates. We see that these tasks should be equivalent because the product of the dimensions of the bonds crossing the minimal cut is equal to 2^{200} in both cases: for the brickwork circuit, 100 gates cross the cut if we orient the cut horizontally through the diagram in Figure 4-3 (with $r = 1$) and each gate contributes a factor of 4; meanwhile, for the depth-40 circuit, only one fourth of the unitary layers will contain gates that cross the minimal cut, and each of these layers will have 20 such gates that each contribute a factor of 2 (CZ gates have half the rank of generic gates). The task of simulating a depth-40 circuit on a 7×7 lattice was reported to require more than two hours using tensor network contraction on the 281 petaflop supercomputer Summit [Vil+20], and the exponentiality of the runtime suggests scaling this to 20×20 would take many orders of magnitude longer, a task that is decidedly intractable.

The discrepancy between maximal lattice sizes achieved for the two architectures is a result of the fact that the two have very different effective 1D dynamics. In particular, the entanglement of the effective dynamics for the brickwork architecture saturates to a significantly smaller value than that of the cluster state architecture. And even more directly relevant for prospects of fast simulation, the typical spectrum of Schmidt values across some cut of the effective 1D dynamics for the brickwork architecture decays far more rapidly than that of the 1D dynamics for `CHR`. For this reason, the slower-decaying eigenvalue spectrum of `CHR` was significantly more costly for the runtime of the algorithm. (In fact, the eigenvalue spectrum of the brickwork model decayed sufficiently quickly that we were primarily limited not by the runtime of our algorithm, but by our numerical precision, which could in principle be increased.) But while the slower decay of the spectrum for the `CHR` model necessitated a longer runtime for a given sidelength, it allowed us to study the functional form of the spectrum and in particular compare against the predictions of the toy model of Section 4.2.4 as we discuss below.

While we were computationally limited to probing low-depth and small-size models, our numerical results point toward `SEBD` having an asymptotic running time for both models bounded by $\mathrm{poly}(n, 1/\varepsilon, 1/\delta)$ in order to sample with variational dis-

tance ε or compute output probabilities with additive error ε/q^n with probability $1-\delta$, suggesting that Conjecture 1 is true. Our numerical evidence for this is as follows.

1. We find that the effective 1D dynamics associated with these random circuit families appear to be in area-law phases, as displayed in Figure 4-4. That is, the entanglement does not grow extensively with the sidelength L, but rather saturates to some constant. We furthermore observe qualitatively identical behavior for some Rényi entropies S_α with $\alpha < 1$. It is known [Sch+08] that this latter condition is sufficient to imply that a 1D state may be efficiently described by an MPS, indicating that `SEBD` is efficient for these circuit families and that Conjecture 1 is true.

2. For further evidence of efficiency, we study the functional form of the entanglement spectra of the effective 1D dynamics. For the effective 1D dynamics corresponding to `CHR`, we observe superpolynomial decay of eigenvalues (i.e. squared Schmidt values) associated with some cut, displayed in Figure 4-5, indicating that choosing a maximal bond dimension of $D = \text{poly}(1/\epsilon)$ is more than sufficient to incur less than ϵ truncation error per bond. The observed spectrum tends toward a scaling which is qualitatively consistent with the asymptotic scaling of $\lambda_i \sim 2^{-\Theta(\log^2(i))}$ predicted by the toy model of Section 4.2.4 and consistent with our Conjecture 1'. Note that this actually suggests that the required bond dimension of `SEBD` may be even smaller than $\text{poly}(1/\epsilon)$, scaling like $D = 2^{\Theta(\sqrt{\log(1/\epsilon)})}$.

While these numerical results may be surprising given the prevalence of average-case hardness conjectures for quantum simulation, they are not surprising from the perspective of the recent works (discussed in previous sections) that find strong evidence for an entanglement phase transition from an area-law to volume-law phase for 1D unitary-and-measurement processes driven by measurement strengths. Since the effective dynamics of the 2D random shallow circuits we study are exactly such processes, our numerics simply point out that these systems are likely on the area-law side of the transition. (However, no formal universality theorems are known, so the various models of unitary-and-measurement circuits that have been studied are generally not known to be equivalent to each other.) In the case of the brickwork architecture, we are also able to provide independent analytical evidence (Section 4.5.6) that this is the case by showing the "quasi-entropy" $\tilde{S}_2$ for the 1D process is in the area-law phase. We leave the problem of numerically studying the precise relationship between circuit depth, qudit dimension, properties of the associated stat mech models (including "quasi-entropies") as discussed in subsequent sections, and the performance of `SEBD` for future work. In particular, simulations of larger depth and larger qudit local dimension could be used to provide numerical support for Conjecture 2, which claims that as these parameters are increased the circuit architectures eventually transition to a regime where our algorithms are no longer efficient.

4.5 Analytical evidence for conjectures from statistical mechanics

4.5.1 Overview

In the previous section, we provided strong numerical evidence that `SEBD` is efficient when acting on certain sufficiently shallow architectures. Here we provide complementary, analytical evidence that bolsters the case for `SEBD`'s (and, in Section 5.2, `Patching`'s) efficiency. The method is based on a technique developed in [NVH18; Key+18; ZN19; Hun19; BCA20; Jia+20] that maps random quantum circuits to classical statistical mechanical models. We describe how the method can be applied generally to different 2D architectures, but we give special attention to the depth-3 brickwork architecture because it is a worst-case hard uniform architecture which is simple enough for concrete conclusions to be drawn that act as evidence that the algorithms are efficient. The stat mech mapping also provides evidence of computational phase transitions as qudit dimension and circuit depth are increased.

The mapping produces a classical stat mech model for which the entanglement properties of the underlying random circuit are related to thermodynamic properties of the model. In particular, we examine a quantity we call the "quasi-k entanglement entropy" $\tilde{S}_k$ to quantify the entanglement of the 1D state "tracked" by `SEBD` at any given point in time throughout the effective 1D dynamics; the mapping relates $\tilde{S}_k$ to the free energy cost incurred by twisting boundary conditions of the stat mech system. The quasi-k entropy is related but not exactly equal to the Rényi-k entanglement entropy averaged over random circuit instances and measurement outcomes, denoted by $\langle S_k \rangle$. Ideally, we would find rigorous bounds on $\langle S_k \rangle$ (for $0 < k < 1$) for these states throughout the effective 1D dynamics to show that `SEBD` is efficient. We study the quasi-entropies $\tilde{S}_k$ instead because the stat mech mapping permits for an analytical handle on $\tilde{S}_k$ for integer $k \geq 2$, and the calculations become especially tractable for $k = 2$. Changing the qudit dimension q of the random circuit model corresponds to changing the interaction strengths in the associated stat mech model, which drives a phase transition. This phase transition in the classical stat mech model is accompanied by phase transitions in quasi-entropies. Even though the efficiency of our algorithms is related to different entropic quantities, which are hard to directly analyze, the phase transition in quasi-entropies provides analytical evidence in favor of our conjectures.

In the remaining subsections, we define the quasi-entropy, explain the stat mech map (with special attention for the case of $k = 2$), apply it generally to 2D circuits to reason heuristically about order-disorder behavior, and finally conclude by applying it more rigorously to the depth-3 brickwork architecture, where we observe a q-driven order-disorder phase transition in the corresponding stat mech model. A more general and more detailed formulation of the stat mech mapping, including its mathematical justification, is given in Section 5.1.

4.5.2 Quasi-entropy

Given an ensemble of pure quantum states, the quasi-entropy is a quantity that is related to the expected amount of entanglement in the state. In our case, the ensemble is generated by a random quantum circuit followed by a projective measurement on some subset of the qudits, and the quasi-entropy is computed as follows.

Suppose we fix a random quantum circuit instance drawn according to some specified architecture, as well as a known outcome for a projective measurement performed on some subset of the output qudits. Let ρ be the pure output state on the unmeasured qudits associated with the instance and measurement outcome, and fix the normalization $\mathrm{Tr}(\rho)$ to be equal to the probability of obtaining the specified measurement outcome. Then for any $k \geq 0$ and for some subregion A of the unmeasured qudits, we define

$$Z_{k,\emptyset} = \mathrm{tr}(\rho)^k \tag{4.17}$$

$$Z_{k,A} = \mathrm{tr}\big(\rho_A^k\big). \tag{4.18}$$

where ρ_A is the reduced density matrix of ρ on region A. Letting $\mathbb{E}_U$ denote expectation over choice of instance and uniformly random measurement outcome, the quasi-k entropy $\tilde{S}_k(A)$ for the random circuit ensemble is defined as

$$\tilde{S}_k(A) := \frac{1}{1-k} \log \left(\frac{\mathbb{E}_U(\mathrm{tr}(\rho)^k \frac{Z_{k,A}}{Z_{k,\emptyset}})}{\mathbb{E}_U(\mathrm{tr}(\rho)^k)} \right) \tag{4.19}$$

$$= \frac{1}{1-k} \log \left(\frac{\mathbb{E}_U(Z_{k,A})}{\mathbb{E}_U(Z_{k,\emptyset})} \right) \tag{4.20}$$

$$= \frac{F_{k,\emptyset} - F_{k,A}}{1-k} \tag{4.21}$$

where $F_{k,\emptyset/A} := -\log\big(\mathbb{E}_U(Z_{k,\emptyset/A})\big)$ will be associated with the "free energy" of the classical stat mech model that the circuit maps to. Virtually identical quantities were also considered in two other very recent works [BCA20; Jia+20].

Note the similarity of the above expression to the average Rényi-k entanglement entropy.

$$\langle S_k(A)_\rho \rangle := \frac{\mathbb{E}_U(\mathrm{tr}(\rho) S_k(A)_\rho)}{\mathbb{E}_U(\mathrm{tr}(\rho))} \tag{4.22}$$

$$= \frac{1}{1-k} \frac{\mathbb{E}_U\left(\mathrm{tr}(\rho) \log \frac{Z_{k,A}}{Z_{k,\emptyset}}\right)}{\mathbb{E}_U(\mathrm{tr}(\rho))}. \tag{4.23}$$

Indeed, the two formulas are the same, except that the quasi-entropy weights instances by $\mathrm{Tr}(\rho)^k$ instead of $\mathrm{Tr}(\rho)$, and takes the logarithm after taking the expectation.

Also note that in the limit $k \to 1$, both $\tilde{S}_k$ and $\langle S_k \rangle$ approach the expected von Neumann entropy

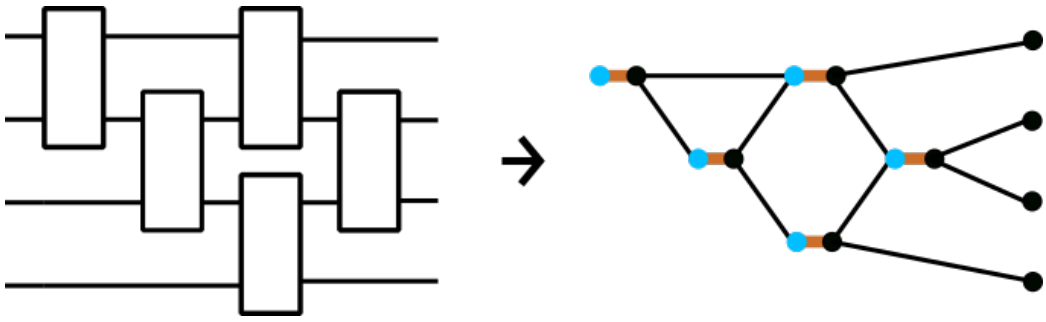


Figure 4-6: Example of stat mech mapping applied to a circuit diagram with 4 qudits and 5 Haar-random gates. Thick orange edges carry Weingarten weight. Black edges carry measurement-dependent weight.

$$\langle S(A) \rangle = - \mathop{\mathbb{E}}_{U} \left(\mathrm{tr} \left(\frac{\rho_A}{\mathrm{tr}(\rho)} \log \left(\frac{\rho_A}{\mathrm{tr}(\rho)} \right) \right) \right) \tag{4.24}$$

This observation lends some justification to the use of $\tilde{S}_k$ as a proxy for $\langle S_k \rangle$ even when $k \neq 1$. This is further justified by previous work studying random 1D circuits without measurements; in [Key+18], the growth rate of $\tilde{S}_2$ in random 1D circuits was calculated using the stat mech mapping and no significant difference was found with numerical calculations of $\langle S_2 \rangle$. Moreover, [ZN19] used the *replica trick* to directly compute $\langle S_2 \rangle$ as a series in $1/q$, where q is the qudit local dimension, and found that the leading term of this expansion agrees with $\tilde{S}_2$, indicating that $\tilde{S}_2$ is a valid substitute for $\langle S_2 \rangle$ in the $q \to \infty$ limit and suggesting it is a good approximation when q is finite.

4.5.3 Mapping

We now describe the procedure for mapping a random quantum circuit family to a classical statistical mechanical model, such that quantities $\mathbb{E}_U(Z_{k,\emptyset})$ and $\mathbb{E}_U(Z_{k,A})$ for integers $k \geq 2$ are given by partition functions of the stat mech model. This follows work in [NVH18; Key+18; ZN19; Hun19; BCA20; Jia+20], although our presentation is for the most part self-contained. Here we present merely how to perform the mapping, leaving the details of its justification to Section 5.1. In that appendix, we also present a more generalized version of the mapping that accounts for the possibility of weak measurements acting in between Haar-random gates.

To define the stat mech model we must specify two ingredients: first, the nodes and edges that form the interaction graph on which the model lives, and second, the details of the interactions between nodes that share an edge. The graph, which is the same for all k, is formed from the circuit diagram as follows. First, we replace each Haar-random unitary (labeled by integer u) in the circuit diagram with a pair of nodes, which we refer to as the *incoming* node t_u and *outgoing* node s_u for that unitary, and we connect nodes t_u and s_u by an edge. Then, we add edges between the outgoing node s_{u_1} of unitary u_1 and the incoming node t_{u_2} of another unitary

u_2 when u_2 acts immediately after u_1 on the same qudit. Finally, we introduce a single auxiliary node x_a for each qudit $a \in [n]$ that is not measured (recall ρ is the output only on the unmeasured qubits), and we add a single edge connecting x_a to the outgoing node s_u for unitary u if u is the final unitary of the circuit to act on qudit a. Thus, all of the incoming and outgoing nodes have degree equal to three, unless they are associated with the first unitary to act on a certain qubit or the last unitary to act on a measured qubit. We provide a simple example of this mapping in Figure 4-6.

Each node in the graph may now be viewed as a spin that takes on one of $k!$ values, corresponding to an element of the symmetric group S_k. A spin configuration is given by an assignment $(\sigma_u, \tau_u) \in S_k \times S_k$ for each pair of nodes (s_u, t_u), as well as an assignment $\chi_a \in S_k$ to each auxiliary node x_a. The main utility of the stat mech mapping is then given by the following equation, expressing the quantities $\mathbb{E}_U(Z_{k,\emptyset/A})$ as a sum over spin configurations on this graph

$$\mathop{\mathbb{E}}_{U}(Z_{k,\emptyset/A}) = \sum_{\{\sigma_u\}_u,\{\tau_u\}_u} \prod_u \text{weight}(\langle s_u t_u \rangle) \prod_{\langle s_{u_1} t_{u_2} \rangle} \text{weight}(\langle s_{u_1} t_{u_2} \rangle) \prod_{\langle s_u x_a \rangle} \text{weight}(\langle s_u x_a \rangle) \tag{4.25}$$

This is a partition function — a weighted sum over spin configurations where the weight of each term is given by a product of factors that depend only on the spin value of a pair of nodes (s, t) connected by an edge, denoted $\langle st \rangle$. In this case, the sum runs only over the values σ_u and τ_u, of the incoming and outgoing nodes; the values χ_a of the auxiliary nodes are fixed across all the terms and encode the boundary conditions that differ between $\mathbb{E}_U(Z_{k,\emptyset})$ and $\mathbb{E}_U(Z_{k,A})$. We define the free energy to be the negative logarithm of this partition function (see Eq. (4.21)), mirroring the standard relationship $F = -k_B T \log(Z)$ between the free energy and the partition function from statistical mechanics, with $k_B T$ set to 1.

We now specify the details of the interaction by defining the weight function for different edges. There are only two different kinds of interactions. Edges $\langle s_u t_u \rangle$ between incoming and outgoing nodes of the same unitary have

$$\text{weight}(\langle s_u t_u \rangle) = \text{wg}(\tau_u \sigma_u^{-1}, q^2) \tag{4.26}$$

where $\text{wg}(\pi, q^2)$ is the Weingarten function. The Weingarten function arises from performing the expectations over the Haar measure in evaluation of the expressions for $Z_{k,\emptyset}$ and $Z_{k,A}$, and one formula for it is given in the Supplementary Information in Equation (5.19). Note that there exist permutations π for which $\text{wg}(\pi, q^2) < 0$, so the overall weight of a configuration can be negative and our stat mech model would only correspond to a physical model with complex-valued energy.

Meanwhile, edges $\langle s_{u_1} t_{u_2} \rangle$ connecting nodes of successive unitaries u_1 and u_2 (resp. edges $\langle s_u x_a \rangle$ connecting outgoing nodes to auxiliary nodes) have weight that depends only on the number of cycles in the product of the permutations assigned to

each of the nodes.

$$\text{weight}(\langle s_{u_1} t_{u_2} \rangle) = q^{C(\sigma_{u_1} \tau_{u_2}^{-1})} \tag{4.27}$$

$$\text{weight}(\langle s_u x_a \rangle) = q^{C(\sigma_u \chi_a^{-1})} \tag{4.28}$$

where $C(\pi)$ returns the number of cycles that make up the permutation π. This weight function becomes more complicated when weak measurements are applied in between gates u_1 and u_2, a generalization we discuss further in Section 5.1 of the Supplementary Information.

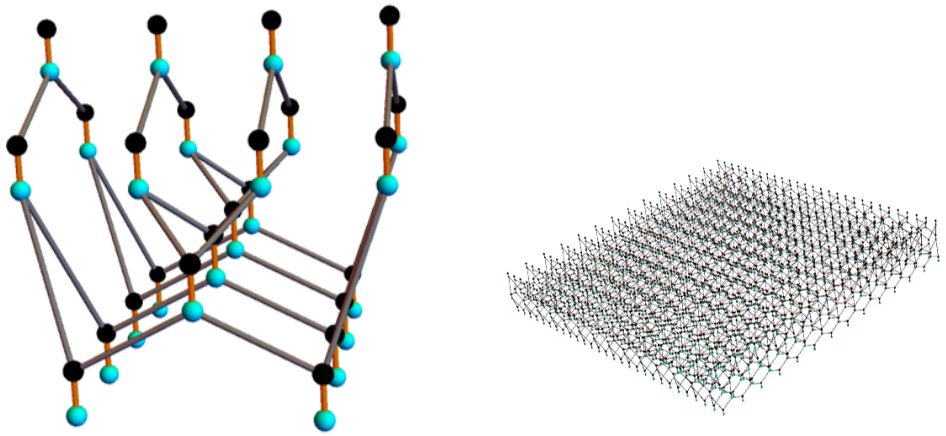
The final piece of this prescription is setting the value χ_a for each of the auxiliary nodes x_a at the end of the circuit, which can be seen as fixing the boundary conditions for the stat mech model. These nodes are fixed to the same value for each term in the sum and depend on whether we are calculating $\mathbb{E}_U(Z_{k,\emptyset})$ or $\mathbb{E}_U(Z_{k,A})$, and whether the qudit a is in the region A. For $\mathbb{E}_U(Z_{k,\emptyset})$, the value χ_a is fixed to the identity permutation e for every a. Meanwhile, for $\mathbb{E}_U(Z_{k,A})$, we "twist" the boundary conditions and change χ_a to be the k-cycle $(1 \dots k)$ if a is in A, leaving $\chi_a = e$ if a is in the complement of A.

4.5.4 Special case of $k = 2$

When $k = 2$, the symmetric group S_k has only 2 elements, identity (denoted by e) and swap (denoted by (12) in cycle notation), so the quantities $\mathbb{E}_U(Z_{2,\emptyset})$ and $\mathbb{E}_U(Z_{2,A})$ map to partition functions of Ising-like classical stat mech models where each node takes on one of two values. Furthermore, in the $k = 2$ case with no measurements, it was shown in [NVH18; Key+18] (see also [ZN19; Hun19]) that one can get rid of all negative terms in the partition function by decimating half of the nodes, i.e. explicitly performing the sum over the values of the incoming nodes $\{\tau_u\}_u$ in Eq. (4.25). This continues to be true even when there are measurements in between unitaries in the circuit, as discussed in the Supplementary Information. However, the decimation causes the two-body interactions to become three-body interactions between any three nodes $s_{u_1}, s_{u_2}, s_{u_3}$ when unitary u_3 succeeds unitaries u_1 and u_2 and shares a qudit with each. The lack of negative weights for $k = 2$ is convenient because it allows one to view the system as a classical spin model at a real temperature and can therefore be analyzed with well-studied numerical techniques like Monte Carlo sampling.

4.5.5 Mapping applied to general 2D circuits

We now apply the mapping directly to a depth-d circuit acting on a $\sqrt{n} \times \sqrt{n}$ lattice of qudits consisting of nearest-neighbor two-qudit Haar-random gates. This is the relevant case for the algorithms presented in this paper. In this section, we will assume for concreteness that the first unitary layer includes gates that act on qudits at gridpoints (i, j) and $(i, j + 1)$ for all odd i and all j, the second layer on (i, j) and $(i, j + 1)$ for all even i and all j, the third layer on (i, j) and $(i + 1, j)$ for all i and all odd j, and the fourth layer on (i, j) and $(i + 1, j)$ for all i and all even j. Subsequent layers then cycle through these four orientations.



(a) Depth-4 circuit on 4×4 lattice (b) Depth-5 circuit on 28×28 lattice

Figure 4-7: The graph produced by the stat mech mapping on shallow 2D circuits. (a) A close up view of the graph reveals that the degree of most nodes is three, similar to the honeycomb lattice. (b) A far-away view reveals that globally the graph looks like a two dimensional slab of thickness roughly d.

4.5.5.1 The classical stat mech model

Replacing the unitaries in the circuit diagram with pairs of nodes and connecting them as described previously yields a graph embedded in three dimensions. The nodes in this graph still have degree three, so locally the graph looks similar to the honeycomb lattice (the lattice that arises from a 1+1D circuit as discussed in [NVH18; Hun19; BCA20; Jia+20] and in Chapter 5), but globally the nodes form a 3D lattice that can be viewed roughly as a $\sqrt{n} \times \sqrt{n} \times d$ slab, although the details of how these nodes connect is not straightforward to visualize. We have included pictures of the graph in Figure 4-7.

Recall that edges between nodes originating from the same unitary are assigned a weight equal to the Weingarten function and edges between successive unitaries follow the interaction $\text{weight}(\langle s_{u_1} t_{u_2} \rangle) = q^{C(\sigma_{u_1} \tau_{u_2}^{-1})}$. For $k = 2$ this amounts to a ferromagnetic Ising interaction where

$$\text{weight}(\langle s_{u_1} t_{u_2} \rangle) = \begin{cases} q^2 & \text{if } \sigma_{u_1} \tau_{u_2} = e \\ q & \text{if } \sigma_{u_1} \tau_{u_2} = (12). \end{cases} \tag{4.29}$$

To analyze the output state, we will divide the n qudits into three groups A, B, and C. We suppose that after the d unitary layers have been performed, a projective measurement is performed on the qudits in region B. Qudits in regions A and C are left unmeasured and we wish to calculate quantities like $\mathbb{E}_U(Z_{k,\emptyset})$ and $\mathbb{E}_U(Z_{k,A})$. The mapping calls for us to introduce an auxiliary node for each unmeasured qudit in the circuit, i.e. an auxiliary node for qudits in regions A and C. For $\mathbb{E}_U(Z_{k,\emptyset})$ all of the auxiliary nodes are set to identity e, while for $\mathbb{E}_U(Z_{k,A})$, the auxiliary nodes for region A are set to the k-cycle $(1 \dots k)$.

4.5.5.2 Eliminating negative weights via decimation when $k = 2$

The quantities $\mathbb{E}_U(Z_{k,\emptyset/A})$ are now given by classical partition functions on this graph with appropriate boundary conditions for the auxiliary nodes in regions A and C. We wish to understand whether this stat mech model is ordered or disordered. We are faced with the issue that the Weingarten function can take negative values and thus some configurations over this graph could have negative weight. For $k = 2$, as previously discussed, we can rectify this by decimating all the incoming nodes. The resulting graph has half as many nodes and interactions between groups of three adjacent nodes s_{u_1}, s_{u_2}, and s_{u_3}, whenever unitary u_3 acts after u_1 and u_2. There is a simple formula for the weights:

$$\text{weight}(\langle s_{u_1} s_{u_2} s_{u_3} \rangle) = \begin{cases} 1 & \text{if } \sigma_{u_1} = \sigma_{u_2} = \sigma_{u_3} \\ \frac{1}{q+q^{-1}} & \text{if } \sigma_{u_2} \neq \sigma_{u_3} \\ 0 & \text{if } \sigma_{u_1} \neq \sigma_{u_2} = \sigma_{u_3}. \end{cases} \tag{4.30}$$

Now, all the weights are non-negative. Moreover, the largest weight occurs when all the nodes agree, indicating a generally ferromagnetic interaction between the trio of nodes. If either σ_{u_1} or σ_{u_2} disagrees with the other two values, the weight is reduced by a factor of $q + q^{-1}$. When σ_{u_3} disagrees, the weight is 0; these configurations are forbidden and contribute nothing to the partition function.

Given an assignment of e or (12) to each node σ_u, we can associate a pattern of domain walls, that is, a set of edges connecting nodes with disagreeing values. These domain walls partition the 2D slab into contiguous domains of adjacent nodes all given the same value.

4.5.5.3 Allowed domain wall configurations and disorder-order phase transitions

Using this observation, we can understand the kinds of domain wall structures that will appear in configurations that contribute non-zero weight. Recall that the stat mech model occupies a 2D slab of constant thickness in the direction of time, which we orient vertically. In this setting, domain wall structures are membrane-like since the graph is embedded in 3D. Membranes that have upward curvature, shaped like a bowl, are not allowed, because somewhere there would need to be an interaction where the upper node disagrees with the two below it, a situation that leads to 0 weight as in Eq. (4.30). On the other hand, cylindrically shaped domain wall membranes do not have this issue, nor do dome-shaped membranes with downward curvature. The weight of a configuration is reduced by a factor of $q+q^{-1}$ for each unit of domain wall, an effect that acts to minimize the domain wall size when drawing samples from the thermal distribution (energy minimization). On the other hand, larger domain walls have more configurational entropy — there are many ways to cut through the graph with a cylindrically shaped membrane — an effect that acts to bring out more domain walls in samples from the thermal distribution (entropy maximization). The question is, which of these effects dominates? For a certain

setting of the depth d (slab thickness) and local dimension q, is there long-range order, or is there exponential decay of correlations indicating disorder? Generally speaking, increasing depth magnifies both effects: cylindrical domain wall membranes must be longer — meaning larger energy — when the depth is larger; however, longer cylinders also have more ways of propagating through the graph. Meanwhile, increasing q only magnifies the energetic effect since it increases the interaction strength and thus the energy cost of a domain wall unit but leaves the configurational entropy unchanged.

Thus, in the limit of large q we expect the energetic effect to win out and the system to be ordered for any fixed circuit depth d and any circuit architecture. What about small q? Physically speaking, q must be an integer at least 2 since it represents the local Hilbert space dimension of the qudit. However, the statistical mechanical model itself requires no such restriction, and we can allow q to vary continuously in the region $[1, \infty)$. Then for $q \to 1$, the energy cost of one unit of domain wall becomes minimal (but it does not vanish). Depending on the exact circuit architecture and the depth of the circuit, the system may experience a phase transition into the disordered phase once q falls below some critical threshold q_c. The depth-3 circuit with brickwork architecture that we present later in Section 4.5.6 provides an example of such a transition. It is disordered when $q = 2$ and experiences a phase transition as q increases to the ordered phase at a transition point we estimate to be roughly $q_c \approx 6$.

When q is fixed and d is varied, it is less clear what to expect. Suppose for small d, the system is disordered. Then increasing d will amplify both the energetic and entropic effects, but likely not in equal proportions. If the amplification of the energetic effect is stronger with increasing depth, then we expect to transition from the disordered phase to the ordered phase at some critical value of the depth d_c. Without a better handle on the behavior of the stat mech model, we cannot definitively determine if and when this depth-driven phase transition will happen.

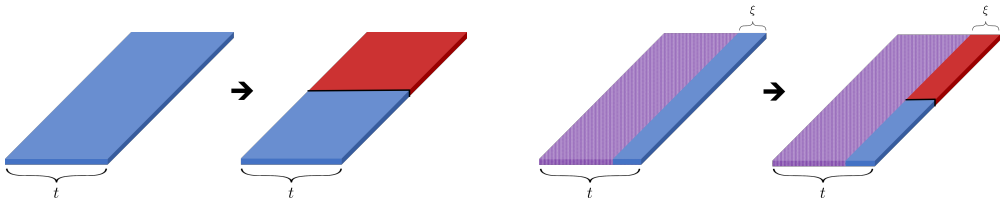
However, we have other reasons to believe that there should be a depth-driven phase transition. In particular, we now provide an intuitive argument for why a disorder-order transition in the parameter q should imply a disorder-order transition in the parameter d. Consider fixed d, and another fixed integer $r \geq 1$ such that $d/r \gg 1$. We may group together $r \times r$ patches of qudits to form a "supersite" with local dimension q^{r^2}. Similarly, we may consider a "superlayer" of $O(r)$ consecutive unitary layers. Since $O(r)$ layers is sufficient to implement an approximate unitary k-design on a $r \times r$ patch of qudits (taking $k = O(1)$) [HM18], we intuitively take each superlayer to implement a Haar-random unitary between pairs of neighboring supersites. Thus, a depth-d circuit acting on qudits of local dimension q is roughly equivalent to a depth-$O(d/r)$ circuit acting on qudits of local dimension q^{r^2} in the supersite picture. If for a fixed d, we observe a disorder-order phase transition for increasing q, then for fixed q and fixed d/r, we should also observe a disorder-order phase transition with increasing r. Equivalently, we should see a transition for fixed q and increasing d. This logic is not perfect because superlayers do not exactly map to layers of Haar-random two-qudit gates between neighboring supersites, but nonetheless we take it as reason to expect a depth-driven phase transition.

4.5.5.4 Efficiency of SEBD algorithm from stat mech

The efficiency of the SEBD algorithm relies on the error incurred during the MPS compression being small. If the inverse error has a polynomial relationship (or better) with the bond dimension of truncation, then the algorithm's time complexity is polynomial (or better) in the inverse error and the number of qudits. This will be the case if the MPS prior to truncation satisfies an area law for the k-Rényi entropy for some $0 < k < 1$. The stat mech mapping is unable to probe these values of k. However, we hypothesize that the behavior of larger values of k is indicative of the behavior for $k < 1$ since the examples where the k-Rényi entropy with $k \geq 1$ satisfies an area law but efficient MPS truncation is not possible require contrived spectrums of Schmidt coefficients. Although some physical processes give rise to situations where the von Neumann and k-Rényi entropies with $k > 1$ exhibit different behavior (see e.g. [Hua19], which showed that for random 1D circuits without measurements but with the unitaries chosen to commute with some conserved quantity, after time t the entropy is $O(t)$ for $k = 1$ but $O(\sqrt{t \log t})$ for $k > 1$), the numerical evidence we gave in Section 4.4, where the scaling of all the k-Rényi entropies appears to be the same, suggests our case is not one of these situations.

Previously, we discussed how for 1D circuits with alternating unitary and weak measurement dynamics, there has been substantial numerical evidence for a phase transition from an area-law phase to a volume-law phase as the parameters of the circuit are changed. There has also been analytic work [BCA20; Jia+20] on this model using the stat mech mapping, and in Section 5.1, we use a similar approach to analyze 1D circuits with a different form of weak measurement, inspired by the CHR problem discussed earlier, and show there is a q-driven phase transition from a disordered phase to an ordered phase. The SEBD algorithm simulating a 2D circuit of constant depth made from Haar-random gates may be viewed as a system with very similar dynamics — an alternation between entanglement-creating unitary gates and entanglement-destroying weak measurements. However, none of the unitary-and-measurement models that have been previously studied capture the exact dynamics of SEBD, one reason being that SEBD tracks the evolution of several columns of qudits at once (recall it must include all qudits within the lightcone of the first column). The Haar-random unitaries create entanglement within these columns of qudits, but not in the exact way that entanglement is created by Haar-random nearest-neighbor gates acting on a single column. Nonetheless, we expect the story to be the same for the dynamics of SEBD since the main findings of studies of these unitary-and-measurement models have been quite robust to variations in which unitary ensembles and which measurements are being implemented; we expect that varying parameters of the circuit architecture like q and d can lead to entanglement phase transitions, and thus transitions in computational complexity.

Indeed, the discussion from the previous section suggests precisely this fact. When we apply the stat mech mapping directly to 2D circuits instead of to 1D unitary-and-measurement models, we expect disorder-order phase transitions as both q and d are varied. To make the connection to entanglement entropy explicit here, we note that after t steps of the SEBD algorithm, all $\sqrt{n}$ qudits in the first t columns of the



(a) Ordered phase (b) Disordered phase

Figure 4-8: The stat mech mapping yields nodes arranged within a roughly $\sqrt{n} \times t \times d$ prism. (a) In the ordered phase, twisting the boundary conditions at the right boundary introduces a domain wall between the two phases (indicated by red and blue) that propagates through the bulk for a total area of $O(td)$. (b) In the disordered phase, boundary conditions introduce bias that is noticeable only within a constant $O(\xi)$ distance of the boundary, and the domain wall membrane introduced by twisting the boundary conditions is quickly washed out by the bulk disorder (dotted purple). The total area is $O(\xi d)$.

$\sqrt{n} \times \sqrt{n}$ lattice have been measured, and we have an MPS representation of the state on columns $t+1$ through $t+r$, where $r = O(d)$ is the radius of the lightcone (which depends on circuit architecture, but cannot be larger than d). To calculate the entropy of the MPS, we take the region A to be the top half of these r columns, and region C to be the bottom half. Region B consists of the first t columns, which experience projective measurements. The prescription for computing $\tilde{S}_2(A)$ calls for determining the free energy cost of twisting the boundary conditions in region A, which creates a domain wall along the $A : C$ border. If the bulk is in the ordered phase, then this domain wall membrane originating at the $A : C$ boundary will penetrate through the graph a distance of t, leading to a domain wall area of $O(td)$. If the bulk is in the disordered phase, it will only penetrate a constant distance, on the order of the correlation length ξ of the disordered stat mech model, before being washed out by the disorder, leading to a domain wall area of only $O(\xi d)$. This is the key observation that connects order-disorder to the quasi-entropy; the observation is inspired by a similar transition for random tensor networks (as opposed to random quantum circuits), studied in [Vas+19]. The typical domain wall configurations before and after twisting boundary conditions in the ordered and disordered phases is reflected in the cartoon in Figure 4-8. As elaborated upon in Section 5.1 of the Supplementary Information, we expect there to be a correspondence between the scaling of the domain wall size and the free energy cost after twisting the boundary conditions of the stat mech model.

This implies that the quasi-entropy $\tilde{S}_2$ is in the area (resp. volume) law phase when the classical stat mech model is in the disordered (resp. ordered) phase. Heuristically we might expect the runtime of the `SEBD` algorithm to scale like $\text{poly}(n) \exp\left(O(\tilde{S}_2)\right)$, suggesting that the disorder-to-order transition is accompanied by an efficient-to-inefficient transition in the complexity of the `SEBD` algorithm. Furthermore, near the transition point within the volume-law phase, the quasi-entropy scales linearly with

system size but with a small constant prefactor, suggesting that the SEBD runtime, though exponential, could be considerably better than previously known exponential-time techniques.

4.5.6 Depth-3 2D circuits with brickwork architecture

Now, we turn our attention specifically to the depth-3 brickwork architecture that we also numerically simulated. In this architecture, three layers of two-qudit gates are performed on a 2D lattice of qudits as shown in Fig. 4-9(a). Note that this architecture was also introduced in Section 4.3; the architecture we consider here is exactly the "extended brickwork architecture" of that section with the extension parameter r fixed to be one.

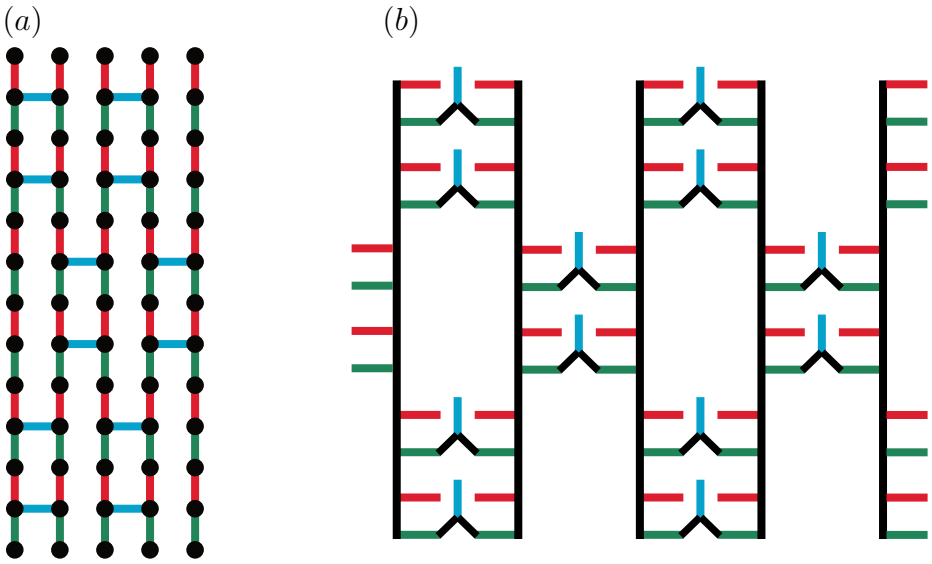


Figure 4-9: (a) Brickwork architecture. Qudits lie at location of black dots. Three layers of two-qudit gates act between nearest-neighbor qudits — first qudits linked by a vertical red edge, then vertical green, then horizontal blue. In our SEBD simulation of this circuit architecture, we sweep from left to right. (b) Result of stat mech mapping applied to brickwork architecture depicted in (a). Nodes are implied to lie at the endpoints of each edge. Red, green, and blue edges carry Weingarten weight. Black edges carry weight given by $\text{weight}(\langle s_{u_1} t_{u_2} \rangle) = q^{C(\sigma_{u_1} \tau_{u_2}^{-1})}$

As previously discussed in Section 4.3, this structure is known to be universal in the sense that one may simulate any quantum circuit using a brickwork circuit (with polynomial overhead in the number of qudits) by judiciously choosing which two-qudit gates to perform and performing adaptive measurements [BFK09]. Thus, it is hard to exactly sample or compute the output probabilities of brickwork circuits in the worst case assuming the polynomial hierarchy does not collapse, and we expect neither the SEBD algorithm nor the Patching algorithm to be efficient. However, we now give evidence that these algorithms are efficient in the "average-case," where each

two-qudit gate is Haar random, by considering the order/disorder properties of the stat mech model that the brickwork architecture maps to.

4.5.6.1 Stat mech mapping for general k.

The stat mech mapping proceeds as previously discussed for 2D circuits, but we will see that the brickwork architecture allows us to make some important simplifications. Each gate in the circuit is replaced by a pair of nodes, which are connected with an edge. Then, the outgoing nodes of the first (red) layer are connected to the incoming nodes of the second (green) layer, and the outgoing nodes of the second (green) layer are connected to the incoming nodes of the third (blue) layer. The resulting graph is shown in Fig. 4-9(b). Edges connecting incoming and outgoing nodes of the same layer are shown in color (red, green, blue) and carry weight equal to the Weingarten function. Edges connecting subsequent layers are black. These edges carry weight given by $\text{weight}(\langle s_{u_1} t_{u_2} \rangle) = q^{C(\sigma_u \tau_{u_2}^{-1})}$.

To perform the full mapping, we would also add a layer of auxiliary nodes for any unmeasured qudits and connect them to the third layer. However, we are interested primarily in the bulk order-disorder properties of the sytem and suppose that all the qudits, except perhaps those at the boundary of the 2D system, will be measured after the third layer, so we need not consider auxiliary nodes.

Looking at Fig. 4-9(b), we see that some of the nodes have degree 1 and connect to the rest of the graph via a (red or blue) Weingarten link. We can immediately decimate these nodes from the graph. For any τ, we have [Gu13]

$$\sum_{\sigma \in S_k} \text{wg}(\tau\sigma^{-1}, q^2) = \sum_{\sigma \in S_k} \text{wg}(\sigma, q^2) = \frac{(q^2-1)!}{(k+q^2-1)!} \tag{4.31}$$

which is independent of τ, so decimating these spins merely contributes the above constant to the total weight. This constant will appear in both the numerator and denominator of quantities like $\mathbb{E}_U(Z_{k,A}) / \mathbb{E}_U(Z_{k,\emptyset})$, and we ignore them henceforth. The remaining graph can be straightened out, yielding Fig. 4-10(a). The fact that Fig. 4-10(a) is a graph embedded in a plane that includes only two-body interactions is one upshot of studying the brickwork architecture, as it makes the analysis more straightforward and the stat mech model easier to visualize. This property and the fact that the brickwork architecture is universal for MBQC constitute the primary reasons we studied this architecture in the first place. Architectures with larger depth would lead to stat mech models that cannot be straightforwardly collapsed onto a single plane while maintaining the two-body nature of the interactions.

4.5.6.2 Simplifications when $k = 2$.

As in previous examples, we examine the $k = 2$ case. In this case we might as well decimate all the degree-2 nodes in the graph in Fig. 4-10(a). This yields a graph with entirely degree-3 nodes, as shown in Fig. 4-10(b). The graph has two kinds of links, both carrying standard Ising interactions. The vertical blue links have weights given

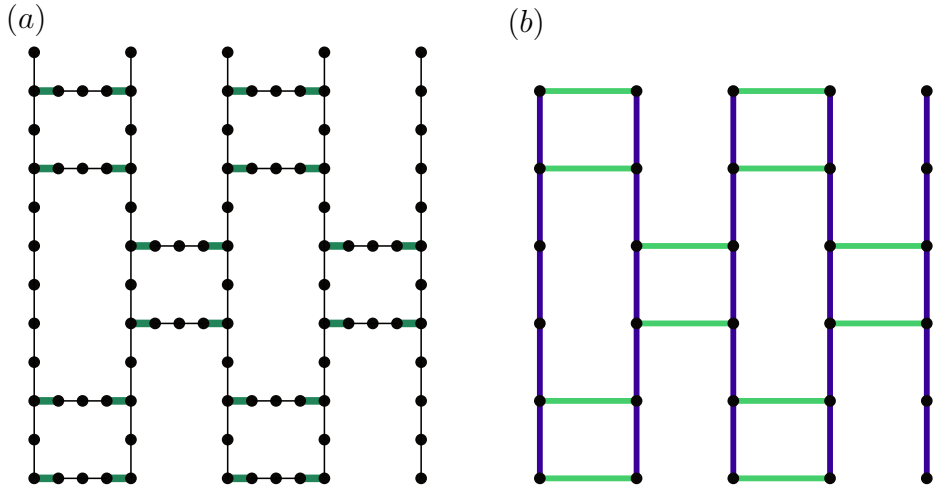


Figure 4-10: (a) The graph that results from decimating degree-1 nodes in Fig. 4-9(b). Each thin black link carries weight equal to the function $q^{C(\sigma\tau^{-1})}$ while each thick green link carries weight equal to $\mathrm{wg}(\sigma\tau^{-1}, q^2)$. (b) The graph that results from decimating nodes of the graph in (a). For $k = 2$, both the horizontal light green and the vertical blue links are ferromagnetic, but have different strengths.

by

$$\mathrm{weight}(\langle s_u s_{u'} \rangle) = \begin{cases} q^2(q^2+1) & \text{if } \sigma_u \sigma_{u'} = e \\ q^2(2q) & \text{if } \sigma_u \sigma_{u'} = (12) \end{cases} \tag{4.32}$$

while the horizontal light green links have weights given by

$$\mathrm{weight}(\langle s_u s_{u'} \rangle) = \begin{cases} \frac{1}{q^2(q^4+1)^2}\left(q^6+q^4-4q^3+q^2+1\right) & \text{if } \sigma_u \sigma_{u'} = e \\ \frac{1}{q^2(q^4+1)^2}\left(2q^5-2q^4-2q^2+2q\right) & \text{if } \sigma_u \sigma_{u'} = (12) \end{cases} \tag{4.33}$$

Both of these interactions are ferromagnetic and become stronger as q increases. We may think of the model as the square lattice Ising model for which $1/2$ of the links carry a ferromagnetic interaction of one strength, $1/4$ of the links carry ferromagnetic interactions of another strength, and the final $1/4$ of the links have no interaction at all. The energy functional can be written

$$E/(kT) = -J_{\mathrm{vert}} \sum_{\langle ij \rangle} s_i s_j - J_{\mathrm{horiz}} \sum_{\langle ij \rangle} s_i s_j \tag{4.34}$$

where s_i take on values in $\{+1, -1\}$. For $q = 2$ we have $J_{\mathrm{vert}} = \log(5/4)/2 = 0.112$ and $J_{\mathrm{horiz}} = \log(53/28)/2 = 0.319$. Both of these values are weaker than the critical interaction strength for the square lattice Ising model of $J_{\mathrm{square}} = \log\left(1+\sqrt{2}\right)/2 = 0.441$. This indicates that the graph generated by the stat mech mapping on 2D circuits of depth 3 with brickwork architecture is in the disordered phase when $q = 2$.

This remains true for $q = 3$. For $q = 4$, $J_{\text{horiz}} = 0.500 > J_{\text{square}}$, but $J_{\text{vert}} = 0.377 < J_{\text{square}}$. Recall that $1/4$ of the links can be thought to have $J = 0$ since they are missing. Taking this into account, the value of J averaged over all the links remains below J_{square} for $q = 5$, and slightly exceeds it for $q = 6$.

This indicates that when we run `SEBD` on these uniform depth-3 circuits with Haar-random gates, the quasi-entropy satisfies $\tilde{S}_2 = O(1)$ (independent of the number of qudits n) when $q = 2$ or $q = 3$ (and probably also for $q = 4$ and $q = 5$). We take this as evidence that `SEBD` would be efficient for these circuits.

4.6 Future work and open questions

Our work yields several natural follow-up questions and places for potential future work. We list some here.

1. Can ideas from our work also be used to simulate *noisy* 2D quantum circuits? Roughly, we expect that increasing noise in the circuit corresponds to decreasing the interaction strength in the corresponding stat mech model, pushing the model closer toward the disordered phase, which is (heuristically) associated with efficiency of our algorithms. We therefore suspect that if noise is incorporated, there will be a 3-dimensional phase diagram depending on circuit depth, qudit dimension, and noise strength. As the noise is increased, our algorithms may therefore be able to simulate larger depths and qudit dimensions than in the noiseless case.

2. Can one approximately simulate random 2D circuits of arbitrary depth? This is the relevant case for Google's quantum computational supremacy experiment [Aru+19]. Assuming Conjecture 2, our algorithms are not efficient once the depth exceeds some constant, but it is not clear if this difference in apparent complexity for shallow vs. deep circuits is simply an artifact of our simulation method, or if it is inherent to the problem itself.

3. Our algorithms are well defined for all 2D circuits, not only random 2D circuits. Are they also efficient for other kinds of unitary evolution at shallow depths, for example evolution by a fixed local 2D Hamiltonian for a short amount of time?

4. Can we rigorously prove Conjecture 1? One way to make progress on this goal would be to find a worst-case-hard uniform circuit family for which it would be possible to perform the analytic continuation of quasi-entropies $\tilde{S}_k$ in the $k \to 1$ limit using the mapping to stat mech models.

5. Can we give numerical evidence for Conjecture 2, which claims that our algorithms undergo computational phase transitions? This would require numerically simulating our algorithms for circuit families with increasing local Hilbert space dimension and increasing depth and finding evidence that the algorithms eventually become inefficient.

6. How precisely does the stat mech mapping inform the efficiency of our algorithms? Is the correlation length of the stat mech model associated with the runtime of our simulation algorithms? How well does the phase transition point in the stat mech model (and accompanying phase transition in quasi-entropies) predict the computational phase transition point in the simulation algorithms? If such questions are answered, it may be possible to predict the efficiency and runtime of the simulation algorithms for an arbitrary (and possibly noisy) random circuit distribution via Monte Carlo studies of the associated stat mech model. In this way, the performance of the algorithms could be studied even when direct numerical simulation is not feasible.

7. In the regime where `SEBD` is inefficient, i.e., when the effective 1D dynamics it simulates are on the volume-law side of the entanglement phase transition, is `SEBD` still better than previously known exponential-time methods? Intuitively, we expect this to be the case close to the transition point.

8. Can `SEBD` and/or `Patching` be generalized to simulate shallow circuits in three or higher dimensions? For `SEBD` the natural approach would be to use a PEPS (higher dimensional generalization of MPS) and simulate action of unitary gates and measurements, but PEPS cannot be efficiently contracted or truncated exactly in the same way as MPS.

Chapter 5

Classical Algorithms for Random Shallow 2D Quantum Circuits, II: Details and Derivations

5.1 General description and justification of the stat mech mapping

In the main text, we described how one maps a random quantum circuit to a classical stat mech model. Our description assumed that the circuit consists of Haar-random unitary gates and that at the end of the circuit, some subset of the qudits experience projective measurements. We only described how to perform the mapping and not the background that is used to justify it. Here, we present (and justify) a generalized formalism that allows for weak measurements to be inserted in between the Haar-random unitary gates; then we apply this formalism to a model of 1D random quantum circuits with weak measurements inspired by the `CHR` problem introduced in the main text. The problem of 1D circuits interspersed with weak measurements was previously studied using this approach in [BCA20; Jia+20] (however, we analyze a different weak measurement).

5.1.1 Generalized mapping procedure

5.1.1.1 Setup.

Let our system consist of n qudits of local dimension q. The circuits we consider are specified by a sequence of pairs of qudits (indicating where unitary gates are applied) and single-qudit weak measurements; this sequence can be assembled into a quantum circuit diagram. The single-qudit measurements are each described by a set $\mathcal{M}$ of measurement operators along with a probability distribution μ over the set $\mathcal{M}$. These sets are normalized such that $\mathrm{tr}\big(M^\dagger M\big)$ is constant for all $M \in \mathcal{M}$ and $\mathbb{E}_{M \leftarrow \mu} M^\dagger M = \mathbb{I}_q$ where $\mathbb{I}_q$ is the $q \times q$ identity matrix. Thus we have $\mathrm{tr}\big(M^\dagger M\big) = q$ for all M. The introduction of a probability measure over $\mathcal{M}$ in our notation, which

was also used in [Jia+20], is not conventional, but it is equivalent to the standard formulation and will be important for later definitions.

When a measurement is performed, if the state of the system at the time of measurement is σ, the probability of measuring the outcome associated with operator M is $\mu(M)\operatorname{tr}\big(M\sigma M^\dagger\big)$ (Born rule for quantum measurements). For a fixed outcome M, the quantity $\operatorname{tr}\big(M\sigma M^\dagger\big)$ is a function of σ that we refer to as the *relative likelihood* of obtaining the outcome M on the state σ, since it gives the ratio of the probability of obtaining outcome M in the state σ to the probability of obtaining outcome M in the maximally mixed state $\frac{1}{q}\mathbb{I}_q$. After obtaining outcome M, the state is updated by the rule $\sigma \to M\sigma M^\dagger/\operatorname{tr}\big(M\sigma M^\dagger\big)$. Thus a pure initial state remains pure throughout the evolution. For notational convenience and without loss of generality, we will assume that for each u, the uth unitary is immediately followed by single-qudit measurements $(\mathcal{M}_u, \mu_u)$ and $(\mathcal{M}'_u, \mu'_u)$ on the qudits $a_u, a'_u \in [n]$ that are acted upon by the unitary, respectively; in the case no measurement is performed, we may simply take $\mathcal{M}_u$ to consist solely of the identity operator, and in the case that more than one measurement is performed, we may multiply together the sets of measurement operators and their corresponding probability distributions to form a single set describing the overall weak measurement.

Thus, the (non-normalized) output state of the circuit with l unitaries acting on the initial state $|1\dots1\rangle$ can be expressed as

$$\rho = E\,|1\dots1\rangle\,\langle1\dots1|\,E^\dagger \tag{5.1}$$

with

$$E = (M'_l M_l U_l)\dots(M'_2 M_2 U_2)(M'_1 M_1 U_1) \tag{5.2}$$

where each unitary U_u is chosen from the Haar measure over unitaries acting on qudits a_u and a'_u, while M_u and M'_u are the measurement operators associated with the measurement outcome obtained upon performing a measurement on qudits a_u and a'_u, respectively, following application of unitary U_u.

5.1.1.2 Goal.

As discussed in the main text, the objective of the stat mech mapping is to learn something about the entanglement entropy for the output state ρ on some subset A of the qudits. The k-Rényi entanglement entropy for the state ρ on region A is defined as

$$S_k(A)_\rho = \frac{1}{1-k}\log\left(\frac{Z_{k,A}}{Z_{k,\emptyset}}\right) \tag{5.3}$$

where

$$Z_{k,\emptyset} = \operatorname{tr}(\rho)^k \tag{5.4}$$

$$Z_{k,A} = \operatorname{tr}\big(\rho_A^k\big). \tag{5.5}$$

and ρ_A is the reduced density matrix of ρ on region A. The von Neumann entropy $S(A)_\rho = -\operatorname{tr}\left(\frac{\rho_A}{\operatorname{tr}(\rho)} \log\left(\frac{\rho_A}{\operatorname{tr}(\rho)}\right)\right)$ represents the $k \to 1$ limit.

For the purposes of assessing the efficiency of our algorithms, we would like to be able to calculate the average value of the k-Rényi entropy over the random choice of the unitaries in the circuit and for measurement outcomes drawn randomly according to the Born rule. Mathematically, we let the notation $\mathbb{E}_U(Q)$ represent the expectation of a quantity Q when the unitaries of the circuit are drawn uniformly at random from the Haar measure and the measurement outcomes are drawn at random from the distribution over their respective sets of measurement operators

$$\mathop{\mathbb{E}}_{U}(Q) = \mathop{\mathbb{E}}_{M_1 \leftarrow \mu_1} \mathop{\mathbb{E}}_{M_1' \leftarrow \mu_1'} \dots \mathop{\mathbb{E}}_{M_l \leftarrow \mu_1} \mathop{\mathbb{E}}_{M_l' \leftarrow \mu_1'} \int_{U(q^2)} dU_1 \dots \int_{U(q^2)} dU_l \; Q \tag{5.6}$$

Here $\int_{U(q^2)}$ denotes integration over the Haar measure of the unitary group with dimension q^2. To take into account the Born rule, a certain choice of unitaries and measurement outcomes leading to the output state ρ, as in Eq. (5.1), should be weighted by $\operatorname{tr}(\rho)$, i.e. the product of the relative likelihoods of all the measurement outcomes. Thus, the relevant average k-Rényi entropy values are given by

$$\langle S_k(A)_\rho \rangle := \frac{\mathbb{E}_U(\operatorname{tr}(\rho) S_k(A)_\rho)}{\mathbb{E}_U(\operatorname{tr}(\rho))} \tag{5.7}$$

$$= \frac{1}{1-k} \frac{\mathbb{E}_U\left(\operatorname{tr}(\rho) \log \frac{Z_{k,A}}{Z_{k,\emptyset}}\right)}{\mathbb{E}_U(\operatorname{tr}(\rho))}. \tag{5.8}$$

However, the quantity naturally computed by the stat mech model is not $\langle S_k(A)_\rho \rangle$, but rather the "quasi-entropy" given by

$$\tilde{S}_k(A) := \frac{1}{1-k} \log\left(\frac{\mathbb{E}_U(\operatorname{tr}(\rho)^k \frac{Z_{k,A}}{Z_{k,\emptyset}})}{\mathbb{E}_U(\operatorname{tr}(\rho)^k)}\right) \tag{5.9}$$

$$= \frac{1}{1-k} \log\left(\frac{\mathbb{E}_U(Z_{k,A})}{\mathbb{E}_U(Z_{k,\emptyset})}\right) \tag{5.10}$$

$$= \frac{F_{k,\emptyset} - F_{k,A}}{1-k} \tag{5.11}$$

where $F_{k,\emptyset/A} := -\log\big(\mathbb{E}_U(Z_{k,\emptyset/A})\big)$. When clear, we abbreviate $\tilde{S}_k(A)$ by $\tilde{S}_k$ and $\langle S_k(A)_\rho \rangle$ by $\langle S_k \rangle$. It is apparent that $\langle S_k \rangle$ is not equal to the quasi-entropy $\tilde{S}_k$: the definition of $\langle S_k \rangle$ weights circuit instances by $\operatorname{tr}(\rho)$ and takes the log before the expectation, while the definition of $\tilde{S}_k$ weights by $\operatorname{tr}(\rho)^k$ and takes the log afterward. Indeed, it is possible for $\tilde{S}_k$ to be smaller than some constant independent of L (area law), while $\langle S_k \rangle$ scales extensively with L (volume law) due to fluctuations of the random variable $Z_{k,A}$ away from its average value toward 0.

Importantly, though, $\tilde{S}_k \to \langle S_k \rangle$ as $k \to 1$; in this limit both quantities approach the expected observed von Neumann entropy $\langle S \rangle$ of the circuit output. This conclu-

sion is justified by L'Hospital's rule and noting that $Z_{k,A}/Z_{k,\emptyset} \to 1$ as $k \to 1$. We will see that the stat mech mapping can only be applied for integers $k \geq 2$, so unfortunately it does not allow for direct access to formula $\tilde{S}_k$ in this limit. As discussed in the main text, we still take $\tilde{S}_k$ to be an informative proxy for $\langle S_k \rangle$ and generally for the entanglement properties of the system.

5.1.1.3 Generalized interaction weights.

The method of forming an interaction graph from the random quantum circuit diagram is discussed in the main text and captured by the example in Figure 4-6. The quantities $\mathbb{E}_U(Z_{k,\emptyset})$ and $\mathbb{E}_U(Z_{k,A})$ are then given by a partition function on this graph, as in Eq. (4.25). In the main text, we gave equations for the edge weights for edges $\langle s_u t_u \rangle$, $\langle s_{u_1} t_{u_2} \rangle$, and $\langle s_u x_a \rangle$ in Eqs. (4.26), (4.27), and (4.28). Here we first generalize these to the case where we allow weak measurements, and justify the formulas in the next subsection. The former equation, which gives the interaction weight between the incoming node t_u and the outgoing node s_u for the same unitary u, is unchanged, and still reads

$$\text{weight}(\langle s_u t_u \rangle) = \text{wg}(\tau_u \sigma_u^{-1}, q^2) \tag{5.12}$$

where $\text{wg}(\pi, q^2)$ is the Weingarten function. The Weingarten function arises from performing the integrals over the Haar measure in Eq. (5.6), and one formula for it is given in the next subsection in Equation (5.19). Note that there exist permutations π for which $\text{wg}(\pi, q^2) < 0$, so the overall weight of a configuration can be negative and our stat mech model would only correspond to a physical model with complex-valued energy.

Meanwhile, the weight of interactions between the outgoing node for one unitary u_1 and the incoming node for a successive unitary u_2 (or between a unitary u and an auxiliary node a) must be updated to account for the case that a weak measurement occurs in between the unitaries. This weight, generalizing Eqs. (4.27) and (4.28), is given by

$$\text{weight}(\langle s_{u_1} t_{u_2} \rangle) = \mathop{\mathbb{E}}_{M \leftarrow \mu} \text{tr}\Big((M^\dagger M)^{\otimes k} W_{\sigma_{u_1} \tau_{u_2}^{-1}} \Big) \tag{5.13}$$

$$\text{weight}(\langle s_u x_a \rangle) = \mathop{\mathbb{E}}_{M \leftarrow \mu} \text{tr}\big((M^\dagger M)^{\otimes k} W_{\sigma_u \chi_a^{-1}} \big) \tag{5.14}$$

where W_π is the operator acting on a k-fold tensor product space that performs the permutation π of the registers. Later, in Appendix 5.2, we will be interested in expressing entropies of the *classical* output distribution of the circuit in terms of partition functions and to handle this case we will update Eq. (5.14). Note that the quantity $\text{tr}\big(X^{\otimes k} W_\pi\big)$ is equal for all π with the same cycle structure, which corresponds to some partition $\lambda = (\lambda_1, \ldots, \lambda_r)$ of k, where $\sum_i \lambda_i = k$ and $\lambda_1 \geq \ldots \geq \lambda_r > 0$. Then we have

$$\text{tr}\big(X^{\otimes k} W_\pi\big) = \prod_{i=1}^{r} \text{tr}\big(X^{\lambda_i}\big) \tag{5.15}$$

This formula allows us to simplify the weight formulas (5.13) and (5.14) in a few special cases. If no measurement is made, then $\mathcal{M} = \{I\}$ and $\text{weight}(\langle s_{u_1} t_{u_2} \rangle) = q^{C(\sigma_{u_1} \tau_{u_2}^{-1})}$, where $C(\pi)$ is the number of cycles r in the permutation π, recovering the weight equations (4.27) and (4.28) from the main text. On the other hand, if a projective measurement onto one of the q basis states is made, then $\mathcal{M} = \{\sqrt{q}\Pi_m\}_{m=1}^{q}$ and μ is the uniform distribution, where $\Pi_m = |m\rangle \langle m|$. Since in this case $\text{tr}\big((M^\dagger M)^w\big) = q^w$ for any power w and any $M \in \mathcal{M}$, we have $\text{weight}(\langle s_{u_1} t_{u_2} \rangle) = q^{k-1}$ for any pair σ_{u_1}, τ_{u_2}.

5.1.1.4 Justification of stat mech mapping

In this subsection, our goal is to provide a justification for (1) the mapping procedure that allows $\mathbb{E}_U(Z_{k,\emptyset})$ and $\mathbb{E}_U(Z_{k,A})$ to be expressed as a partition function as in Eq. (4.25) of the main text and (2) the formulas for the interaction weights for this partition function given in the previous subsection in Eqs. (5.12), (5.13), and (5.14).

To begin, for any integer $k \geq 2$, we rewrite $Z_{k,\emptyset/A}$ from Eqs. (5.4) and (5.5) as

$$Z_{k,\emptyset} = \text{tr}[(\rho \otimes \ldots \otimes \rho)] \tag{5.16}$$

$$Z_{k,A} = \text{tr}\Big[(\rho \otimes \ldots \otimes \rho) W_{(1\ldots k)}^{(A)}\Big] \tag{5.17}$$

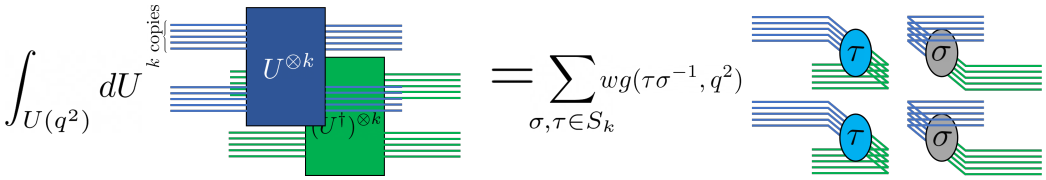
where each trace includes k copies of ρ and $W_{(1\ldots k)}^{(A)}$ is the linear operator that performs a k-cycle permutation, denoted $(1 \ldots k)$ in cycle notation, of the copies for qudits within region A while leaving the copies of the qudits outside of A unpermuted. When $k = 2$ there are two copies of ρ and $W_{(12)}^{(A)}$ is the swap operator for qudits in A.

After substituting Eq. (5.1) for each copy of ρ that appears in the equations above, we obtain an expression with k copies of each unitary U_u and k copies of its adjoint $U_u^\dagger$, as well as k copies each of M_u, $M_u^\dagger$, M'_u, and ${M'_u}^\dagger$. Taking the expectation $\mathbb{E}_U(Z_{k,\emptyset/A})$ introduces integrals over U_u and expectations over M_u and M'_u drawn from distributions μ_u and μ'_u, for each u. To perform the integrals, we rely on techniques for integration over the Haar measure, invoking the formula [Col03; CŚ06]

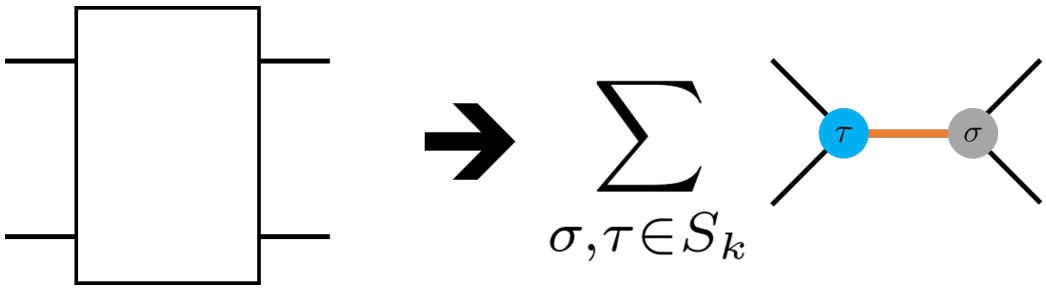
$$\begin{aligned} &\int_{U(q^2)} dU \; U_{i_1 j_1} \ldots U_{i_k j_k} U^\dagger_{i'_1 j'_1} \ldots U^\dagger_{i'_k j'_k} \\ &= \sum_{\sigma,\tau \in S_k} \delta_\sigma(\vec{i}, \vec{j}') \delta_\tau(\vec{i}', \vec{j}) \, \text{wg}(\tau\sigma^{-1}, q^2) \end{aligned} \tag{5.18}$$

where on the left hand side U_{ij} is the (i,j) matrix element of the unitary U and on the right hand side S_k is the symmetric group, $\delta_\sigma(\vec{i}, \vec{j}')$ is shorthand for $\prod_{a=1}^{k} \delta_{i_a j'_{\sigma(a)}}$ and $\text{wg}(\tau\sigma^{-1}, q^2)$ is the Weingarten function, which can be defined in several ways, for example by the following expansion [Col03; CŚ06] over irreducible characters of the symmetric group S_k

$$\text{wg}(\pi, q^2) = \frac{1}{(q^2)!^2} \sum_\lambda \frac{\chi^\lambda(e)^2}{s_{\lambda,q^2}(1)} \chi^\lambda(\pi) \tag{5.19}$$



(a) Haar integration formula applied to k copies of two-qudit gate



(b) Mapping of unitary in circuit diagram

Figure 5-1: (a) Graphical depiction of Haar integration formula given in Eq. (5.18). (b) Haar integration formula allows us to replace Haar-random unitaries from circuit diagram with sums over configurations on a graph with nodes taking values in S_k, and edges between graphs contributing a factor to the weight of a configuration.

where the sum is over all partitions λ of the integer k, χ^λ is the irreducible character of S_k associated with the partition λ, e is the identity permutation, and $s_{\lambda,q^2}(1)$ is the Schur polynomial evaluated at 1 which is equal to the dimension of the representation of $U(q^2)$ associated with λ. Note that there exist permutations π for which $\mathrm{wg}(\pi, q^2)$ is negative.

In words, formula (5.18) states that Haar integration can be performed by summing over all ways of pairing up the incoming index for each of the k copies of U with an outgoing index of a copy of $U^\dagger$, and the incoming index of each copy of $U^\dagger$ with an outgoing index of a copy of U. The permutations $\sigma, \tau \in S_k$ encode which copies are paired with each other, and each permutation pair (σ, τ) is weighted by $\mathrm{wg}(\tau\sigma^{-1}, q^2)$ in the sum. It is helpful to think of this formula graphically, as in Figure 5-1, where we have depicted how the indices pair up after integration over a two-qudit Haar-random unitary.

By applying this formula to all of the Haar-random gates, all of the integrals are eliminated and the tensor network representation of $Z_{k,\emptyset/A}$ can be expressed as a weighted sum over many networks, where each network in this sum corresponds to some choice of (σ_u, τ_u) for every unitary u in the original circuit and some choice of M_u and M'_u from $\mathcal{M}_u$ and $\mathcal{M}'_u$. Furthermore, each network in this weighted sum is itself composed of many disjoint parts that can be individually evaluated. We can see this by observing that when unitary u_2 succeeds unitary u_1 and shares a qudit, Haar integration forces the k tensor indices representing that qudit at that place in the circuit diagram to pair up with the k dual indices for the qudit at the same place, according to some permutation. This happens both at the output of unitary u_1 (corresponding to permutation σ_{u_1}) and at the input of unitary u_2 (corresponding

to permutation τ_{u_2}) yielding a set of closed loops in the tensor network diagram. If the weak measurement acting on that qudit between unitaries u_1 and u_2 is M, then k copies of M and k copies of $M^\dagger$ appear among these loops. An example of such a subdiagram is shown in Figure 5-2.

This observation justifies the partition function Eq. (4.25), as we have expressed $\mathbb{E}_U(Z_{k,\emptyset/A})$ as a weighted sum, with each term labelled by pairs of permutations at the locations of each unitary, where the weight is given by a product of factors that depend only on two of these permutations. These factors are the weights given by Eqs. (5.12), (5.13), and (5.14). Eq. (5.12) accounts for the factor $\mathrm{wg}(\tau\sigma^{-1}, q^2)$ in Eq. (5.18). Meanwhile, we can derive Eq. (5.13) by performing the expectation in Figure 5-2. We can graphically see that each term in the expansion of this expectation is given by $\mu(M)\,\mathrm{tr}\left(W_\sigma M^{\otimes k} W_{\tau^{-1}} (M^\dagger)^{\otimes k}\right)$, and then simply note that W_π commutes with $X^{\otimes k}$ for any X.

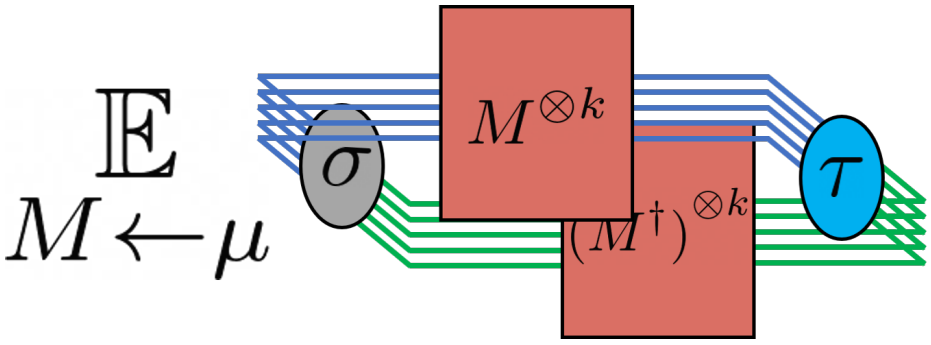


Figure 5-2: Disjoint part that forms tensor network representation of $\mathbb{E}_U(Z_{k,\emptyset/A})$ after performing integrals over Haar-random gates. The weight given by Eqs. (5.13) and (5.14) is derived by evaluating this diagram and taking the expectation with M drawn from $\mathcal{M}$ according to the distribution μ.

Our final task is to justify the introduction of the auxiliary nodes and corresponding weights in Eq. (5.14). The tr in the definition of $Z_{k,\emptyset}$ implies that the indices of the qudits at the circuit output are paired up with their dual indices without permutation. This creates a disjoint closed diagram for each qudit at the circuit output. To evaluate it, we may use the same formula as Eq. (5.13) taking $\tau_{u_2} = e$, the identity permutation. This is equivalent to introducing auxiliary nodes, as we have done, that are fixed to e for all qudits and across all terms in the partition function. The same follows for $Z_{k,A}$ with the exception that the operator $W^{(A)}_{(1\ldots k)}$ is applied to the circuit output, which permutes the output indices of any qudit $a \in A$ prior to connecting them with their dual indices. This is equivalent to introducing an auxiliary node and fixing it to the value $(1\ldots k)$.

There is no need to introduce auxiliary nodes at the beginning of the circuit because we are assuming the circuit acts on the pure product state $|1\ldots1\rangle\langle1\ldots1|$. Thus, the k copies of the index that feeds into the first unitary of the circuit are forced to be 0 and regardless of the permutation value of the incoming node for that unitary, this part of the circuit will contribute a factor of 1. If we had considered

circuits that act initially on the maximally mixed state, we could have handled this by introducing a layer of auxiliary nodes at the beginning of the circuit and fixing their value to e.

5.1.2 Mapping applied to 1D circuits with weak measurements

In Section 4.2.3, we discussed the connection between the effective 1D dynamics of our SEBD algorithm and previous work (originating from [LCF18; Cha+19; SRN19]) on 1D Haar-random circuits with some form of measurements in between each layer of unitaries.

In this subsection, we apply the stat mech mapping to the 1D with weak measurement model and explain the connection between the area-law-to-volume-law transition that has been observed in numerical simulations and the disorder-to-order thermal transition in the classical stat mech model, which occurs at a non-zero critical temperature T_c. This analysis was first performed in [BCA20] and independently in [Jia+20]. The results presented in this section are essentially a reproduction of their analysis but for a different weak measurement, chosen to be relevant for the dynamics of the SEBD algorithm acting on the CHR problem. We include this analysis for two purposes: first, to shed light on the behavior of SEBD acting on CHR, and second, to serve as a more complete example of the stat mech mapping in action, complementing the more heuristic analysis we give in Section 4.5 of the main text.

5.1.2.1 Mapping to the honeycomb lattice.

Let us assume our circuit has n qudits of local dimension q arranged on a line with open boundary conditions. A circuit of depth d acts on the qudits where each layer consists of nearest-neighbor two-qudit Haar-random unitaries. In between each layer of unitaries, a weak measurement is performed on every qudit, described by the set $\mathcal{M}$ of measurement operators and a probability distribution μ over $\mathcal{M}$. The first step of the stat mech mapping is to replace each Haar-random unitary with a pair of nodes and connect these nodes according to the order of the unitaries acting on the qudits. The second step is to introduce a new auxiliary node for each qudit and connect each outgoing node within the final layer of unitaries to the corresponding pair of auxiliary nodes. The resulting graph is the honeycomb lattice, as shown in Figure 5-3(b). We now review what the interactions are on this graph. The horizontal links in Figure 5-3(b) host interactions that contribute a weight equal to the Weingarten function. When $k = 2$, the interaction depends only on if the pair of nodes agree ($\sigma_u \tau_u^{-1} = e$) or if they disagree ($\sigma_u \tau_u^{-1} = (12)$). In this case the interactions are given explicitly by

$$\text{weight}(\langle s_u t_u \rangle) = \text{wg}(\sigma_u \tau_u^{-1}, q^2) \tag{5.20}$$

$$= \begin{cases} \frac{1}{q^4-1} & \text{if } \sigma_u \tau_u^{-1} = e \\ -\frac{1}{q^2(q^4-1)} & \text{if } \sigma_u \tau_u^{-1} = (12). \end{cases} \tag{5.21}$$

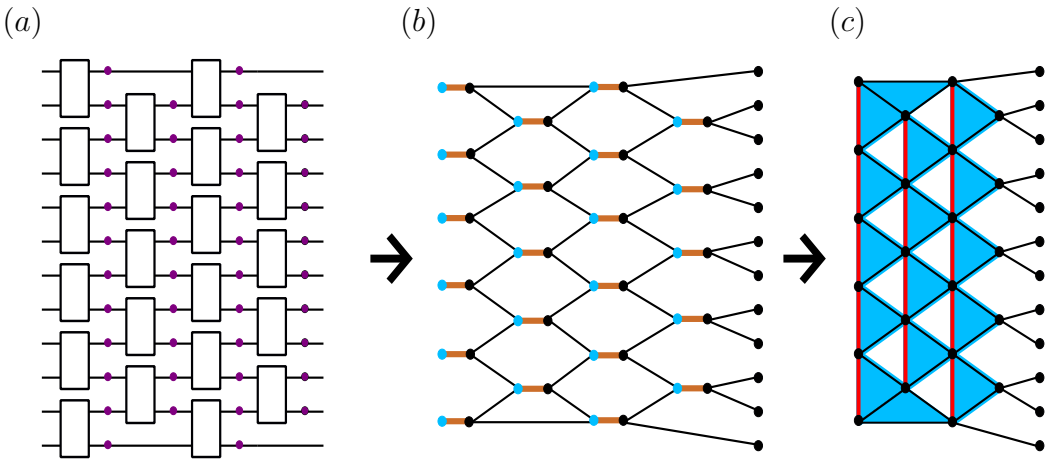


Figure 5-3: Summary of series of maps for Haar-random 1D circuits with weak measurements. (a) The quantum circuit diagram for the unitary plus weak measurement model consists of layers of Haar-random two-qudit gates followed by layers of weak measurements on every qudit, indicated by purple dots. (b) The stat mech mapping results in a model on the honeycomb lattice, where horizontal orange links have weight given by the Weingarten function and diagonal black links have weight that depends on the weak measurement. Blue dots and black dots represent incoming and outgoing nodes, respectively. (c) By decimating the incoming (blue) nodes in the honeycomb lattice, we reduce the number of nodes by half and generate a model with three-body interactions living on rightward-pointing triangles, shaded in blue. When $k = 2$ the weights are all positive, and the three-body interaction can be decomposed into an anti-ferromagnetic interaction along vertical (red) links and ferromagnetic interactions along diagonal (dark blue) links.

Meanwhile, the diagonally oriented links in Figure 5-3(b) host interactions that depend on the details of the weak measurement being applied in between each layer of unitaries, which we now define.

5.1.2.2 Weak measurement and diagonal weights.

The weak measurement we choose is given as follows. First, for a fixed $q \times q$ unitary matrix U, define

$$M_U^{(m)} := \sqrt{q} \cdot \mathrm{diag}(U_{m,\cdot}) \tag{5.22}$$

that is, the $q \times q$ matrix whose diagonal entries are given by the mth row of U, scaled by a factor of $\sqrt{q}$, and whose off-diagonal entries are 0. Define the probability distribution μ_U to be the uniform distribution over the set $\mathcal{M}_U = \{M_U^{(m)}\}_{m=1}^q$. We can see that $(\mathcal{M}_U, \mu_U)$ forms a valid weak measurement since

$$\sum_{m=1}^{q} \mu_U(m)(M_U^{(m)})^\dagger M_U^{(m)} = \sum_{m=1}^{q} \mathrm{diag}(|U_{m,\cdot}|^2) = \mathbb{I}_q \tag{5.23}$$

where the last equality follows from the fact that the sum of the squared norms of the entries within a column of a unitary matrix is 1. When $U = \mathbb{I}_q$, the measurement operator $M_U^{(m)}$ is a projector onto the mth basis state (scaled by a factor of $\sqrt{q}$), and the weak measurement is simply a projective measurement onto the computational basis.

The weak measurement that we consider for our analysis will be a mixture of the weak measurement $(\mathcal{M}_U, \mu_U)$ for different U. Formally, we take $\mathcal{M} = \cup_{U \in U(q)} \mathcal{M}_U$. We let the distribution μ over $\mathcal{M}$ be the distribution resulting from drawing U according to the Haar measure, and then drawing M from $\mathcal{M}_U$ uniformly at random.

This weak measurement is seen to exactly reproduce the weak measurement of `SEBD` acting on `CHR` in Algorithm 4 when $q = 2$, where the measurement operators were the diagonal matrices

$$M^{(1)} := \begin{pmatrix} \cos(\theta/2) & 0 \\ 0 & e^{-i\phi}\sin(\theta/2) \end{pmatrix} \tag{5.24a}$$

$$M^{(2)} := \begin{pmatrix} \sin(\theta/2) & 0 \\ 0 & e^{i\phi}\cos(\theta/2) \end{pmatrix}. \tag{5.24b}$$

with angles (θ, ϕ) drawn according to the Haar measure on the sphere. Indeed, even for $q \neq 2$, this weak measurement arises from a natural generalization of the `CHR` problem, where one makes Haar-random measurements on a cluster state of higher local dimension, which is created by applying a generalized Hadamard gate to each qudit followed by a generalized CZ gate on each pair of neighboring qudits on the 2D lattice.

To compute the weights on the edges of the stat mech model for $k = 2$, we apply the formula in Eqs. (5.13) and (5.14).

$$\begin{aligned} &\text{weight}(\langle s_{u_1} t_{u_2} \rangle) \\ &= \int_{U(q)} dU \sum_{m=1}^{q} \frac{1}{q} \operatorname{tr}\left(\left((M_U^{(m)})^\dagger M_U^{(m)} \right)^{\otimes 2} W_{\sigma_{u_1}\tau_{u_2}^{-1}} \right) \\ &= \int_{U(q)} dU\, q \sum_{m=1}^{q} \begin{cases} \operatorname{tr}\left(\operatorname{diag}(|U_{m,\cdot}|^2)\right)^2 & \text{if } \sigma_{u_1}\tau_{u_2}^{-1} = e \\ \operatorname{tr}(\operatorname{diag}\left(|U_{m,\cdot}|^4\right)) & \text{if } \sigma_{u_1}\tau_{u_2}^{-1} = (12) \end{cases} \\ &= \begin{cases} q^2 & \text{if } \sigma_{u_1}\tau_{u_2}^{-1} = e \\ q^2 \cdot w & \text{if } \sigma_{u_1}\tau_{u_2}^{-1} = (12) \end{cases} \end{aligned} \tag{5.25}$$

where

$$w := \int_{U(q)} dU \sum_m \frac{1}{q} \operatorname{tr}\big(\operatorname{diag}(|U_{m,\cdot}|^4)\big) \tag{5.26}$$

$$= q \int_{U(q)} dU |U_{0,0}|^4 \tag{5.27}$$

$$= q \sum_{\sigma,\tau \in S_2} \operatorname{wg}(\sigma\tau^{-1}, q) \tag{5.28}$$

$$= 2q \sum_{\sigma \in S_2} \operatorname{wg}(\sigma, q) \tag{5.29}$$

$$= 2q \left(\frac{1}{q^2-1} - \frac{1}{q(q^2-1)} \right) \tag{5.30}$$

$$= \frac{2}{q+1}, \tag{5.31}$$

where in the second line we have invoked the Haar integration formula that appears in Eq. (5.18), and then substituted the explicit values for the Weingarten function when $k = 2$. The formula for weight$(\langle s_u x_a \rangle)$ is given similarly.

We can see that for all $q > 1$, the weight is larger when the values of the nodes agree than when they disagree, indicating a ferromagnetic Ising interaction. Indeed, the interaction for $k = 2$ will be ferromagnetic regardless of what weak measurement M is made since $\operatorname{tr}\big(M^\dagger M\big)^2 \geq \operatorname{tr}\big((M^\dagger M)^2\big)$ holds for all M. Furthermore, for our choice of weak measurement, the ferromagnetic Ising interaction becomes stronger as q increases.

5.1.2.3 Eliminating negative weights via decimation when $k = 2$.

The possibility of a negative weight on the horizontal edges of the honeycomb lattice in Figure 5-3(b) appears to impede further progress in the analysis since the classical model cannot be viewed as a physical system with real interaction energies at a real temperature. As discussed in the main text, for $k = 2$, this problem may be circumvented by decimating half of the spins; that is, we explicitly perform the sum over $\{\tau_u\}_u$ in the partition function in Eq. (4.25), yielding a new stat mech model involving only the outgoing nodes s_u. Since the decimated incoming nodes (except for those in the first layer) each have three neighbors, all three of which are undecimated outgoing nodes, the new model will have a three-body interaction between each such trio of nodes.

We may furthermore observe that, for our choice of weak measurement when $k = 2$, the three-body weight may be re-expressed as the product of three two-body weights acting on the three edges of the triangle. Below we give formulas for the two-body weights; our formulas are a unique decomposition of the three-body interaction up to a shifting of overall constant factors from one link to another. Thus, via decimation we have moved from the honeycomb lattice with two-body interactions to the triangular lattice with two-body interactions, as illustrated in Figure 5-3(c). There are two kinds

of two-body interactions on this triangular lattice. Vertically oriented links between nodes s_{u_1} and s_{u_2} host anti-ferromagnetic interactions

$$\text{weight}(\langle s_{u_1} s_{u_2} \rangle) \tag{5.32}$$

$$= \begin{cases} \frac{1}{q^4-1} & \text{if } \sigma_{u_1}\sigma_{u_2} = e \\ \frac{w}{1+q^2}\left((q^2-w^2)(q^2w^2-1)\right)^{-1/2} & \text{if } \sigma_{u_1}\sigma_{u_2} = (12) \end{cases} \tag{5.33}$$

and diagonally oriented links host ferromagnetic interactions, where

$$\text{weight}(\langle s_{u_1} s_{u_2} \rangle) = \begin{cases} q\sqrt{q^2-w^2} & \text{if } \sigma_{u_1}\sigma_{u_2} = e \\ q\sqrt{w^2q^2-1} & \text{if } \sigma_{u_1}\sigma_{u_2} = (12). \end{cases} \tag{5.34}$$

For all values of the measurement strength p, the ferromagnetic interactions are stronger than the anti-ferromagnetic interaction.

5.1.2.4 Phase diagram.

The model described above for $k = 2$ is exactly the anisotropic Ising model on the triangular lattice. In general this model may be described by its energy functional

$$E/kT = -J_1 \sum_{\langle ij \rangle_1} g_i g_j - J_2 \sum_{\langle ij \rangle_2} g_i g_j - J_3 \sum_{\langle ij \rangle_3} g_i g_j \tag{5.35}$$

where $g_i \in \{+1, -1\}$ are Ising spin variables and the three sums are over links along each of the three triangular axes. This model has been studied and its phase diagram is well understood [Hou50; Ste70]. In the setting where along two of the axes the interaction strength is equal in magnitude and ferromagnetic, while along the third axis it is weaker in magnitude and antiferromagnetic, the model is known to experience a phase transition as the temperature is varied. At high temperatures, it is in the disordered phase; in other words, samples drawn from the thermal distribution exhibit exponentially decaying correlations between spin values σ_u with a constant correlation length of ξ. At low temperatures, it is in an ordered phase where samples exhibit long-range correlation. At the critical point, the interaction strengths satisfy the equation [Hou50; Ste70]

$$\sinh(2J_1)\sinh(2J_2) + \sinh(2J_2)\sinh(2J_3) + \sinh(2J_1)\sinh(2J_3) = 1. \tag{5.36}$$

For us, parameter q plays the role of the temperature, and the interaction strengths, derived from Eqs. (5.32) and (5.34), are given by

$$J_1 = J_2 = \frac{1}{4}\log\left(\frac{q^2 - w^2}{w^2q^2 - 1}\right), \tag{5.37}$$

$$J_3 = -\frac{1}{2}\log\left(\frac{w(q^2-1)}{\sqrt{(q^2-w^2)(q^2w^2-1)}}\right). \tag{5.38}$$

Using these equations, we can solve for the critical point, and we find it to be $q_c = 3.249$. Only integer values of q correspond to valid quantum circuits, so we conclude that the model is disordered when $q = 2$ or $q = 3$ and ordered when $q \geq 4$. We plot this one dimensional phase diagram in Figure 5-4.

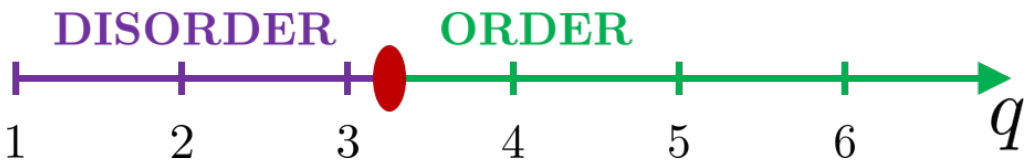


Figure 5-4: Phase diagram showing for which values of q the anisotropic Ising model on the triangular lattice is ordered and disordered. The critical point, indicated by the red dot, occurs at $q_c = 3.249$.

5.1.2.5 Connection between (dis)order and scaling of entanglement entropy.

We expect the scaling of the quantity $\tilde{S}_2 = F_{2,A} - F_{2,\emptyset} = -\log(\mathbb{E}_U(Z_{2,A})/\mathbb{E}_U(Z_{2,\emptyset}))$ to be related to the order or disorder of the model by the following argument. For $\mathbb{E}_U(Z_{2,\emptyset})$, the auxiliary spins are all set to $\chi_a = e$, biasing the bulk spins nearby to prefer e over (12). For $\mathbb{E}_U(Z_{2,A})$, the spins within the region A are twisted so that $\chi_a = (12)$, introducing a domain wall at the boundary. In the ordered phase, the bias introduced at the boundary extends throughout the whole bulk since there is no decay of correlation with distance. The domain wall at the boundary in the calculation of $\mathbb{E}_U(Z_{2,A})$ forces the bulk to separate into two regions with distinct phases separated by a domain wall that cuts through the bulk. The domain wall has length of order $\min(|A|, d)$ where $|A|$ is the number of sites in region A and d is the depth. In the calculation of $\mathbb{E}_U(Z_{2,\emptyset})$, there is no domain wall. The addition of one additional unit of domain wall within a configuration leads the weight of the configuration to decrease by a constant factor, so in the ordered phase we expect $-\log(\mathbb{E}_U(Z_{2,A})/\mathbb{E}_U(Z_{2,\emptyset})) = O(\min(|A|, d))$. Meanwhile, in the disordered phase, there is a natural length scale ξ that boundary effects will penetrate into the bulk. The domain wall at the boundary due to twisted boundary conditions will be washed out by the bulk disorder after a distance on the order of $\xi = O(1)$. Thus we expect $-\log(\mathbb{E}_U(Z_{2,A})/\mathbb{E}_U(Z_{2,\emptyset})) = O(1)$. A cartoon illustrating this logic appears in Figure 4-8 of the main text. For further discussion of the connection between order-disorder

properties of the stat mech model and entropic properties of the underlying quantum objects, see [Vas+19; BCA20; Jia+20].

This logic suggests that, if we take the scaling of $\tilde{S}_2$ to be a good proxy for the scaling of $\langle S_2 \rangle$, the disorder-to-order phase transition in the classical model would be accompanied by an area-law-to-volume-law phase transition in the Rényi-2 entropy of the output of random circuits.

5.1.2.6 Relationship to numerical simulation of SEBD on CHR.

In Section 4.2.3, with fixed $q = 2$, it was established that the effective dynamics of SEBD running on CHR are alternating layers of entangling two-qubit CZ gates and weak measurements on every qubit of a 1D line, where the form of the weak measurement is given explicitly. The dynamics we have studied in this section use the same weak measurement, but choose the two-qubit entangling gates to be Haar-random. We have established that the quasi-2-entropy $\tilde{S}_2$ satisfies an area law for this process when $q = 2$, and the statement remains true for $q = 3$ when the weak measurement corresponds to a natural generalization of the CHR problem to larger local dimension. For $q = 4$, it is no longer true; the dynamics of $\tilde{S}_2$ satisfy a volume law.

Due to the similarity between the dynamics studied in this section and that of SEBD running on CHR, our conclusion provides a partial explanation for the numerical observation presented in Section 4.4 that the average entanglement entropy $\langle S_k \rangle$ satisfies an area law when SEBD runs on CHR for $q = 2$ and various values of k.

5.1.2.7 Additional observations appearing in previous work.

The above analysis is essentially a restatement of what appears in recent works by Bao, Choi, and Altman [BCA20] and separately Jian, You, Vasseur, and Ludwig [Jia+20], except that here we analyzed a different weak measurement. In particular, [BCA20] considered the case where a projective measurement occurs with some probability p on each qudit after each layer of unitaries, and otherwise there is no measurement. They made the observation that we describe above that the $k = 2$ mapping can be written as a 2-body anisotropic Ising model on the triangular lattice with an exact solution. Both of these papers went beyond what we have presented here to analyze the $q \to \infty$ limit directly, where they observed that the stat mech model becomes a standard ferromagnetic Potts model on the square lattice for all integers k. For $k = 2$ this is exactly the square lattice Ising model and indeed, we can see from Eq. (5.38) that when $q \to \infty$, $J_3 \to 0$; the anti-ferromagnetic links along one axis vanish leaving a square lattice with exclusively ferromagnetic interactions. The fact that the model becomes tractable for all integers $k \geq 2$ allows these papers to invoke analytic continuation and make sense of the $k \to 1$ limit, where the quasi-entropy $\tilde{S}_k$ exactly becomes the expected von Neumann entropy $\langle S \rangle$.

5.1.3 Patching

We now describe a second algorithm for sampling from the output distributions and computing output probabilities of 2D quantum circuits acting on qudits of local dimension q. While the SEBD algorithm described in the previous section is efficient if the corresponding effective 1D dynamics can be efficiently simulated with TEBD, the algorithm of this section is efficient if the circuit depth d and local dimension q are constant and the conditional mutual information (CMI) of the classical output distribution is exponentially decaying in a sense that we make precise below. In Section 4.5 we will give evidence that the output distribution of sufficiently shallow random 2D circuits acting on qudits of sufficiently small dimension satisfies such a property with high probability, and the property is not satisfied if the circuit depth or local dimension exceeds some critical constant value.

The algorithm we describe is an adaptation and simplification of the Gibbs state preparation algorithm of [BK19]. In that paper, the authors essentially showed that a quantum Gibbs state defined on a lattice can be prepared by a quasipolynomial time quantum algorithm, if the Gibbs state satisfies two properties: (1) exponential decay of correlations and (2) exponentially decaying quantum conditional mutual information for shielded regions. Our situation is simpler than the one considered in that paper, due to the fact that sufficiently separated regions of the lattice are causally disconnected as a result of the fact that the circuit inducing the distribution is constant-depth and therefore has a constant-radius lightcone. The structure of our algorithm is very similar to theirs, except we can make some simplifications and substantial improvements as a result of the constant-radius lightcone and the fact that we are sampling from a classical distribution rather than a quantum Gibbs state.

Before we describe the algorithm, we set some notation. Let Λ denote the set of all qudits of a $L_1 \times L_2$ rectangular grid (assume $L_1 \leq L_2 \leq \text{poly}(L_1)$). If A and B are two subsets of qudits of Λ, we define $\text{dist}(A, B) := \min_{i \in A, j \in B} \text{dist}(i, j)$, where $\text{dist}(i, j)$ is the distance between sites i and j as measured by the ∞-norm. There are two primary facts that our algorithm relies on. First, if the circuit has depth d, any two sets of qudits separated by a distance greater than $2d$ have non-overlapping lightcones. Hence, if A and B are two lattice regions separated by distance at least $2d$, and ρ is the quantum state output by the circuit (before measurement), it holds that $\rho_{AB} = \rho_A \otimes \rho_B$ and therefore $\mathcal{D}_{AB} = \mathcal{D}_A \otimes \mathcal{D}_B$ if $\mathcal{D} = \sum_{\mathbf{x}} \mathcal{D}(\mathbf{x}) \, |\mathbf{x}\rangle\langle\mathbf{x}|$ is the classical output distribution of the circuit and (for example) $\mathcal{D}_A$ denotes the marginal of $\mathcal{D}$ on subregion A. (Note that our notation is slightly different in this section – we now use subscripts on $\mathcal{D}$ to denote marginals, and the dependence of $\mathcal{D}$ on the circuit instance is left implicit.) Second, if the classical CMI $I(X : Z|Y)_p$ of three random variables with joint distribution p_{XYZ} is small, then p_{XYZ} is close to the distribution $p_{X|Y} p_Y p_{Z|Y}$ corresponding to a Markov chain $X - Y - Z$. We state this more formally as the following lemma, which follows from the Pinsker inequality.

Lemma 29 (see e.g. [CT91]). *Let X, Y, Z be discrete random variables, and let p_{XYZ} denote their joint distribution. Then*

$$I(X:Z|Y)_p \geq \frac{1}{2\ln 2}\|p_{XYZ} - p_{X|Y}p_Y p_{Z|Y}\|_1^2.$$

Following [BK19], we also formally define a notion of CMI decay.

Definition 16 (Markov property). *Let p denote a probability distribution supported on Λ. Then p is said to satisfy the* $\delta(l)$-Markov condition *if, for any tripartition of a subregion X of the lattice into subregions $X = A \cup B \cup C$ such that* $\mathrm{dist}(A, C) \geq l$, *we have*

$$I(A:C|B)_p \leq \delta(l). \tag{5.39}$$

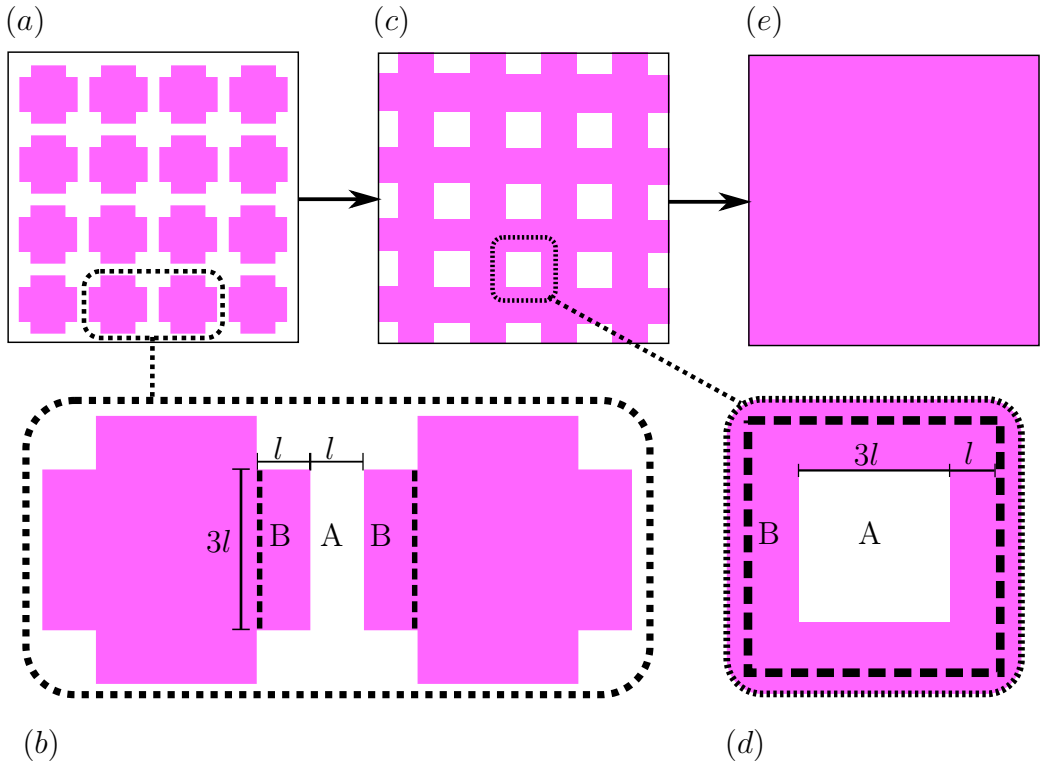


Figure 5-5: `Patching`. Pink represents marginals of the output distribution that have been approximately sampled, while white represents unsampled regions. In (a), the algorithm has sampled from disconnected patches. Figure (b) depicts how the algorithm transitions from configuration (a) to (c). Namely, the algorithm generates a sample from the conditional distribution on A, conditioned on the configuration of region B. Similarly, figure (d) depicts how the "holes" of configuration (c) are filled in. The end result is shown in (e), an approximate sample from the global distribution on the full lattice.

Intuitively, our algorithm works by first sampling from the marginal distributions of spatially separated patches on the lattice, and then stitching the patches together

to approximately obtain a sample from the global distribution. For a $O(1)$-depth circuit whose output distribution has exponentially decaying CMI, the efficiency of this procedure is guaranteed by the two facts above. We now show this more formally.

Theorem 12. *Suppose C is a 2-local quantum circuit of depth d defined on a 2D rectangular grid Λ of $n = L_1 \times L_2$ qudits, and let $\mathcal{D}(\mathbf{x}) := |\langle \mathbf{x}| C |1\rangle^{\otimes n}|^2$ denote its output distribution. Then if $\mathcal{D}$ satisfies the $\delta(l)$-Markov condition, for any integer $l > 2d$* `Patching` *with a length-scale parameter l runs in time $nq^{O(dl)}$ and samples from some distribution $\mathcal{D}'$ that satisfies $\|\mathcal{D}' - \mathcal{D}\|_1 \leq O(1)(n/l^2)\sqrt{\delta(l)}$.*

In particular, if $d = O(1)$, $q = O(1)$, and $\mathcal{D}$ is $\mathrm{poly}(n)e^{-\Omega(l)}$-Markov, then for any polynomial $r(n)$, for some choice of lengthscale parameter `Patching` *runs in time $\mathrm{poly}(n)$ and samples from a distribution that is $1/r(n)$-close to $\mathcal{D}$ in total variation distance.*

Proof. The algorithm proceeds in three steps, illustrated in Figure 5-5. First, for each square subregion R_i shaded in Figure 5-5(a) with $i \in [O(n/l^2)]$, sample from $\mathcal{D}_{R_i}$, the marginal distribution of $\mathcal{D}$ on subregion R_i. To do this, first restrict to the qudits and gates in the lightcone of R_i. Sampling from the output distribution on R_i produced by this restricted version of the circuit is equivalent to sampling from the marginal on R_i of the true distribution produced by the full circuit. Since $l > 2d$, this restriction of the circuit is contained in a sublattice of dimensions $O(l) \times O(l)$. Using standard tensor network methods [MS08], sampling from the output distribution of this restricted circuit on R_i can be performed in time $q^{O(dl)}$. Since there are $O(n/l^2)$ patches, this step can be performed in time $nq^{O(dl)}$. After performing this step, we have prepared the state $\mathcal{D}_{R_1} \otimes \cdots \otimes \mathcal{D}_{R_k} = \mathcal{D}_{R_1,\ldots,R_k}$ where the equality holds because the patches are separated by $l > 2d$ and are therefore mutually independent.

In the second step, we apply "recovery maps" to approximately prepare a sample from the larger, connected lattice subregion S shaded in Figure 5-5(c). The prescription for these recovery maps is given in Figure 5-5(b). Referring to this figure, a recovery map $\mathcal{R}_{B\to AB}$ is applied to generate a sample from subregion A, conditioned on the state of region B. Explicitly, the mapping is given by linearly extending the map $\mathcal{R}_{B\to AB}(|b\rangle\langle b|_B) = \sum_a \mathcal{D}_{A|B}(a|b)\,|a\rangle\langle a|_A \otimes |b\rangle\langle b|_B$. Note that, for a tripartite distribution $\mathcal{D}_{ABC}$, $\mathcal{R}_{B\to AB}(\mathcal{D}_{BC}) = \mathcal{D}_{A|B}\mathcal{D}_B\mathcal{D}_{C|B}$. To implement this recovery map, one can again restrict to gates in the lightcone of region AB and utilize standard tensor network simulation algorithms to generate a sample from the marginal distribution on A, conditioned on the (previously sampled) state of B. The time complexity for this step is again $q^{O(dl)}$. After applying this and $O(n/l^2)$ similar recovery maps, we obtain a sample from a distribution $\mathcal{D}'_S$. By Lemma 29, the triangle inequality, and Definition 16, the error of this step is bounded as

$$\|\mathcal{D}'_S - \mathcal{D}_S\|_1 \leq O(1)(n/l^2)\sqrt{\delta(l)} = O(1)(n/l^2)\sqrt{\delta(l)}. \tag{5.40}$$

Note that the fact that the errors caused by recovery maps acting on disjoint regions accumulate at most linearly has been referred to previously [BK19] as the "union property" for recovery maps. The final step is very similar to the previous step. We now apply recovery maps, described by Figure 5-5(d), to fill in the "holes"

of the subregion S and approximately obtain a sample from the full distribution $\mathcal{D} = \mathcal{D}_\Lambda$. By a similar analysis, we find that the error incurred in this step is again $O(1)(n/l^2)\sqrt{\delta(l)}$, and therefore the procedure samples from a distribution $\mathcal{D}'_\Lambda$ for which $\|\mathcal{D}'_\Lambda - \mathcal{D}_\Lambda\|_1 \leq O(1)(n/l^2)\sqrt{\delta(l)}$.

The second paragraph of the theorem follows immediately by choosing a suitable $l = \Theta(\log n)$. □

A straightforward application of Markov's inequality implies that a polynomial-time algorithm for sampling with error $1/\operatorname{poly}(n)$ succeeds with high probability over a random circuit instance if the output distribution CMI is exponentially decaying in expectation. We formalize this as the following corollary.

Corollary 6. *Let $\mathcal{C}$ be a random circuit distribution. Define $\mathcal{C}$ to be $\delta(l)$-Markov if, for any tripartition of a subregion X of the lattice into subregions $X = A \cup B \cup C$ such that dist$(A, C) \geq l$, we have*

$$\langle I(A : C|B)_{\mathcal{D}} \rangle \leq \delta(l) \tag{5.41}$$

where the angle brackets denote an average over circuit realizations and $\mathcal{D}$ is the associated classical output distribution. Then if $d = O(1), q = O(1)$, and $\mathcal{C}$ is $\operatorname{poly}(n)e^{-\Omega(l)}$-Markov, then for any polynomials $r(n)$ and $s(n)$, `Patching` can run in time $\operatorname{poly}(n)$ and, with probability $1 - 1/s(n)$ over the random circuit realization, sample from a distribution that is $1/r(n)$-close to the true output distribution in variational distance.

Thus, proving that some uniform worst-case-hard circuit family $\mathcal{C}$ is $\operatorname{poly}(n)e^{-\Omega(l)}$-Markov provides another route to proving the part of Conjecture 1 about sampling with small total variation distance error. In Section 4.5, we will give analytical evidence that if $\mathcal{C}$ is a random circuit distribution of sufficiently low depth and small qudit dimension, then $\mathcal{C}$ is indeed $\operatorname{poly}(n)e^{-\Omega(l)}$-Markov, and if the depth or qudit dimension becomes sufficiently large, then $\mathcal{C}$ is not $\operatorname{poly}(n)f(l)$-Markov for any $f(l) = o(1)$, supporting Conjecture 2, which states that our algorithms exhibit computational phase transitions.

Finally, we note that `Patching` can also be used to estimate specific output probabilities of a random circuit instance C with high probability if C is drawn from a distribution $\mathcal{C}$ that is $\operatorname{poly}(n)e^{-\Omega(l)}$-Markov. This shows that the Markov condition could also be used to prove the second part of Conjecture 1 regarding computing output probabilities with small error. This is similar to how `SEBD` can also be used to compute output probabilities, as discussed in Section 4.2.2.

Lemma 30. *Let $\mathcal{C}$ be a circuit distribution over constant depth d and constant qudit dimension q 2D circuits on n qudits which is $\operatorname{poly}(n)e^{-\Omega(l)}$-Markov and invariant under application of a final layer of arbitrary single-qudit gates. Then for a circuit instance C drawn from $\mathcal{C}$ and a fixed $\mathbf{x} \in [q]^n$, a variant of `Patching` can be used to output a number $\mathcal{D}'(\mathbf{x})$ in time $\operatorname{poly}(n)$ that satisfies*

$$|\mathcal{D}'(\mathbf{x}) - \mathcal{D}(\mathbf{x})| \leq q^{-n}/r(n) \tag{5.42}$$

with probability $1 - 1/s(n)$ *for any polynomials* $r(n)$ *and* $s(n)$*, where* $\mathcal{D}$ *is the output distribution associated with* C*.*

Proof. With probability $1 - 1/\operatorname{poly}(n)$ over the circuit instance C, `Patching` with some choice of lengthscale $l = \Theta(\log n)$ efficiently samples from a distribution $\mathcal{D}'_C$ that is $1/\operatorname{poly}(n)$-close in variational distance to $\mathcal{D}_C$ for any choice of polynomials. Hence, for an output probability $\mathbf{y}$ chosen uniformly at random and a circuit C drawn from $\mathcal{C}$, it holds that

$$\mathop{\mathbb{E}}_{\mathbf{y}} \mathop{\mathbb{E}}_{C} |\mathcal{D}'(\mathbf{y}) - \mathcal{D}(\mathbf{y})| \leq q^{-n}/\operatorname{poly}(n) \tag{5.43}$$

if $l = c \log n$ and c is a sufficiently large constant. By a nearly identical argument to that used in the proof of Corollary 3, due to the invariance of $\mathcal{C}$ under application of a final layer of single qudit gates, for some fixed $\mathbf{x} \in [q]^n$ we also have

$$\mathop{\mathbb{E}}_{C} |\mathcal{D}'(\mathbf{x}) - \mathcal{D}(\mathbf{x})| \leq q^{-n}/\operatorname{poly}(n) \tag{5.44}$$

for any choice of polynomial. Finally, it is straightforward to see that an instance of `Patching` that samples from $\mathcal{D}'$ can also be used to exactly compute $\mathcal{D}'(\mathbf{x})$ for any $\mathbf{x}$. (To do this, the algorithm computes conditional probabilities via tensor network contractions as before, except instead of using these conditional probabilities to sample, it simply multiplies them together similarly to how `SEBD` can be used to compute output probabilities.) Applying Markov's inequality completes the proof. □

5.2 Efficiency of `Patching` algorithm from stat mech

We now study the predictions of the stat mech model for the fate of the `Patching` algorithm we introduced in Section 5.1.3. To do so, we in turn study the predictions of the stat mech model for entropic properties of the *classical* output distribution, as `Patching` is efficient if the CMI of the classical output distribution is exponentially decaying with respect to shielded regions.

We have previously applied the stat mech model to study expected entropies of quantum states. However, we now wish to study expected entropies of the classical output distribution. To this end, we now consider the non-unitary quantum circuit consisting of the original, unitary circuit followed by a layer of dephasing channels applied to every qudit. The resulting mixed state is classical (i.e., diagonal in the computational basis) and is exactly equal to the output distribution we want to study. That is, the state after application of the dephasing channels is $\sum_{\mathbf{x}} \mathcal{D}(\mathbf{x}) |\mathbf{x}\rangle\langle\mathbf{x}|$ where $\mathcal{D}$ is the output distribution of the circuit. Note that the application of the dephasing channel is not described in the formalism we have discussed previously, but is easily incorporated. In particular, we need to compute the weights between the auxiliary node x_a and the corresponding outgoing node s_u associated with the unitary u that is the last in the circuit to act on qudit a. We may update Eq. (5.14) (whose original form was derived in Appendix 5.1.1.4) and compute the following,

letting $|\Phi_k\rangle \equiv \left(\sum_{i=1}^{q} |i\rangle \otimes |i\rangle\right)^{\otimes k}$.

$$\begin{aligned} \text{weight}(\langle s_u x_a \rangle) &= \langle \Phi_k | (I \otimes W_{\sigma_u^{-1}}) \left(\sum_{i=1}^{q} |i\rangle\langle i| \otimes |i\rangle\langle i| \right)^{\otimes k} (I \otimes W_{\chi_a}) |\Phi_k\rangle \\ &= \sum_{i_1,\ldots,i_k} \langle i_1, \ldots, i_k | W_{\sigma_u^{-1}} | i_1, \ldots, i_k \rangle \ \langle i_1, \ldots, i_k | W_{\chi_a} | i_1, \ldots, i_k \rangle \,. \end{aligned} \tag{5.45}$$

We therefore see that $\text{weight}(\langle s_u x_a \rangle)$ in this setting is exactly equal to the number of k-tuples of indices $(i_1, \ldots, i_k)$ with $i_j \in [q]$ that are invariant under both permutation operators $\sigma_u, \chi_a \in S_k$ acting as $\sigma_u \cdot (i_1, \ldots, i_k) = (i_{\sigma(1)}, \ldots, i_{\sigma(k)})$. In fact, for our purposes, the auxiliary spin χ_a will either be set to the identity e or to the k-cycle permutation $(1 \ldots k)$. In the former case, the weight reduces to $\text{tr}(W_{\sigma_u}) = q^{C(\sigma_u)}$. In the latter case, since the only tuples that are invariant under application of the cycle permutation $(1 \ldots k)$ are the q tuples of the form $(x, x, \ldots, x)$ for $x \in [q]$, the weight is simply q for all σ_u. Summarizing,

$$\text{weight}(\langle s_u x_a \rangle) = \begin{cases} q^{C(\sigma_u)}, & \chi_a = e \\ q, & \chi_a = (1 \ldots k). \end{cases} \tag{5.46}$$

From these expressions, we may immediately note the following facts. First, flipping some auxiliary spin from e to $(1 \ldots k)$ cannot increase the weight of a configuration, and hence such a flip corresponds to an increase in free energy. Second, if an auxiliary spin is in the $(1 \ldots k)$ configuration, then the auxiliary spin may be effectively removed from the system since in this case the contribution of the auxiliary spin to the weight of a configuration is constant across all configurations.

With these modified weights, we may now compute "quasi-entropies" $\tilde{S}_k(X)$ as before, where now in the $k \to 1$ limit $\tilde{S}_k(X)$ approaches the expected Shannon entropy of the marginal of the output distribution on subregion X, $\langle S(X)_{\mathcal{D}} \rangle$, where the average is over random circuit instances.

5.2.1 Disordered stat mech model suggests `Patching` is successful.

We consider the quasi-CMI defined by

$$\tilde{I}_2(A : C|B) := \tilde{S}_2(AB) + \tilde{S}_2(BC) - \tilde{S}_2(B) - \tilde{S}_2(ABC), \tag{5.47}$$

where all quasi-entropies are taken with respect to the collection of classical output distributions that arise from the quantum circuit architecture. This definition is in analogy to the definition of CMI as $I(A : C|B) = S(AB) + S(BC) - S(B) - S(ABC)$ [CT91]. Note that we may define the quasi-k-CMI $\tilde{I}_k(A : C|B)$ analogously for any nonnegative k, and it holds that $\langle I(A : C|B)_{\mathcal{D}} \rangle = \lim_{k \to 1} \tilde{I}_k(A : C|B)$ where the angle brackets denote an expectation over random circuit instances.

Recalling that $\tilde{S}_2(X) = F_{2,X} - F_{2,\emptyset}$, we may rewrite the quasi-2-CMI as

$$\tilde{I}_2(A:C|B) = (F_{2,AB} - F_{2,B}) - (F_{2,ABC} - F_{2,BC}). \tag{5.48}$$

In stat mech language, the quasi-CMI is essentially the difference in free energy costs of twisting the boundary condition of subregion A in the case where (1) no other spins have boundary conditions, and the case where (2) subregion C also has an imposed boundary condition.

Now, consider some random circuit family $\mathcal{C}$ with associated stat mech model that is in the disordered phase for $k = 2$. For any subregion X of qudits, and partition of X into subregions $X = A \cup B \cup C$, we expect this difference between free energy costs will decay exponentially with the separation between A and C as

$$\tilde{I}_2(A:C|B) \leq \text{poly}(n,q)e^{-\text{dist}(A,C)/\xi} \tag{5.49}$$

where ξ is a correlation length. This is because in the disordered phase of the stat mech model, information about the boundary of region C will be exponentially attenuated as the distance from region C grows. If we take $\tilde{I}_2(A:C|B)$ as a proxy for the average CMI of the output distribution, $\langle I(A:C|B)_{\mathcal{D}} \rangle$, we conclude that the random circuit family $\mathcal{C}$ is $\text{poly}(n,q)e^{-\Theta(l)}$-Markov as defined in Section 5.1.3. The results of that section then show that `Patching` can be used to efficiency sample from the output distribution and estimate output probabilities with high precision and high probability. We take this exponential decay of quasi-2-CMI as evidence that the average CMI also decays exponentially, and therefore that `Patching` is successful. Recall from that main text that the (worst-case-hard) depth-3 brickwork architecture's associated stat mech model is disordered; we therefore expect `Patching` to be capable of efficiently simulating this architecture.

5.2.2 Ordered stat mech model suggests `Patching` is unsuccessful.

We first obtain exact, closed form results in the zero-temperature limit of the stat mech model, which corresponds to the $q \to \infty$ limit. However, we expect that qualitatively similar results hold outside of this limit.

As before, consider the stat mech model obtained by applying dephasing channels to all qudits after the application of all gates. Consider some connected, strict subset A of qudits on the original grid. Suppose we are interested in the quasi-entropy $\tilde{S}_k(A) = (F_{k,A} - F_{k,\emptyset})/(k-1)$ of the output distribution on this region. This quantity is given by the free energy cost of twisting the boundary conditions (auxiliary spins) associated with region A from e to $(1 \dots k)$. The auxiliary spins associated with qudits in the complement of A are fixed to be in the identity permutation configuration, e. For both sets of boundary conditions, all non-auxiliary spins will order in the configuration e. This is because the configuration e maximizes the weights in Equation (5.46) for spins connected to auxiliary spins in the configuration e, and the weight of a spin connected to an auxiliary spin in the configuration $(1 \dots k)$ is

independent of that spin's configuration. Hence, regardless of the configuration of the auxiliary spins, all bulk spins are in the identity permutation configuration in the $q \to \infty$ limit of infinitely strong couplings.

Therefore, twisting a single auxiliary spin from e to $(1 \dots k)$ results in a reduction of the total weight by a factor of $q/q^{C(e)} = q/q^k = q^{1-k}$, corresponding to a free energy increase of $(k-1)\log(q)$. We therefore compute

$$\tilde{S}_k(A) = \frac{F_{k,A} - F_{k,\emptyset}}{k-1} = |A| \log(q). \tag{5.50}$$

Note that this result is exact in the $q \to \infty$ limit. Notably, we find that all integer quasi-entropies are equal in this limit, and so we may trivially perform the analytic continuation to the von Neumann (i.e. Shannon) entropy:

$$\langle S(A) \rangle = \lim_{k \to 1} |A| \log(q) = |A| \log(q). \tag{5.51}$$

Hence, in the $q \to \infty$ limit, the entropy of a strict subregion of the output distribution is maximal.

Now, let X denote the set of *all* qudits. We want to compute $\langle S(X) \rangle$. We again proceed by computing the quasi-entropies:

$$\tilde{S}_k(X) = \frac{F_{k,X} - F_{k,\emptyset}}{k-1}.$$

As before, for each auxiliary spin associated with region X that we "twist", the weight of the configuration is decreased by a factor of q^{1-k} relative to the configuration in which all auxiliary spins are set to e. However, in this case, as opposed to our previous calculation, *all* of the auxiliary spins are twisted. Recall from Equation (5.46) that the weight between a twisted auxiliary spin and a bulk spin is independent of the value of the bulk spin. Hence, if all auxiliary spins are twisted, the lowest energy state in the bulk is no longer just the configuration in which all spins take the value e – in the absence of a symmetry-breaking boundary condition, there is now a global spin-flip symmetry and the ground space is $k!$-fold degenerate, consisting of all configurations in which all bulk spins are aligned. This symmetry contributes a factor of $k!$ to the partition function and $-\log(k!)$ to the free energy. We hence calculate

$$\tilde{S}_k(X) = |A| \log(q) - \frac{\log(k!)}{k-1}. \tag{5.52}$$

We now perform the analytic continuation to the Shannon entropy:

$$\langle S(X)\rangle = \lim_{k\to 1} \tilde{S}_k(X) \tag{5.53}$$

$$= |A|\log(q) - \lim_{k\to 1} \frac{\log(k!)}{k-1} \tag{5.54}$$

$$= |A|\log(q) - \frac{1-\gamma}{\ln(2)} \tag{5.55}$$

$$\approx |A|\log(q) - 0.61, \tag{5.56}$$

where $\gamma \approx 0.557$ denotes the Euler constant. The expected Shannon entropy of the output distribution is therefore $\frac{1-\gamma}{\ln(2)}$ less bits than maximal in the low-temperature limit, corresponding to $q \to \infty$.

From the above facts, we can immediately compute the expected CMI of the output distribution in this limit. Let (A, B, C) be any partition of the qudits. We have

$$\langle I(A:C|B)_{\mathcal{D}}\rangle \tag{5.57}$$

$$\equiv \langle S(AB)_{\mathcal{D}} + S(BC)_{\mathcal{D}} - S(B)_{\mathcal{D}} - S(ABC)_{\mathcal{D}}\rangle \tag{5.58}$$

$$= [(|A|+|B|)\log(q)] + [(|B|+|C|)\log(q)] \tag{5.59}$$

$$- [(|B|)\log(q)] - [(|A|+|B|+|C|)\log(q) - \frac{1-\gamma}{\ln(2)}] \tag{5.60}$$

$$= \frac{1-\gamma}{\ln(2)} \approx 0.61. \tag{5.61}$$

We therefore find that in this limit, the expected CMI of the classical output distribution approaches a constant equal to $\frac{1-\gamma}{\ln(2)}$. While this result was derived with respect to the *completely* ordered stat mech model, corresponding to $q \to \infty$, we expect similar behavior for ordered stat mech models in general. In particular, if X denotes the set of all qudits, in the case of an ordered k^{th}-order stat mech model, $\tilde{S}_k(X)$ will similarly receive an extra contribution corresponding to the global spin-flip symmetry, which will also be contributed to the corresponding quasi-CMI $\tilde{I}_k(A:C|B)_{\mathcal{D}}$. Hence, we do not expect the quasi-CMIs to decay when the corresponding stat mech model is in an ordered phase. We take this as evidence that the average CMI does not decay, and therefore that `Patching` is not successful in efficiently sampling from the output distribution with small error.

5.3 Relation to worst-to-average-case reductions based on truncated Taylor series

Recently, it was shown [Mov19] that for any constant-depth random circuit family with Haar-random gates acting on n qubits for which it is #P-hard to compute output probabilities in the worst case, there does not exist a poly(n)-time algorithm for computing the output probability of some arbitrary output string $\mathbf{x}$ up to additive

error $2^{-\tilde{\Theta}(n^3)}$ with high probability over the circuit realization, unless there exists a poly(n)-time randomized algorithm for computing a #P-hard function. (Note: in even more recent work using the same technique, the error robustness has been improved from $2^{-\tilde{\Theta}(n^3)}$ to $2^{-\Theta(n\log(n))}$ [Bou+21; KMM21].) Essentially, for Haar-random circuits, near-exact average-case computation of output probabilities is as hard as worst-case computation of output probabilities. Our complexity separation in Section 4.3 shows that the error tolerance for this hardness result cannot be improved to $2^{-n}/2^{n^c}$ for any $c < 1$.

This hardness result builds on and improves other prior work [Bou+19] on the average-case hardness of random circuit simulation. In particular, the original paper [Bou+19] uses a different interpolation scheme than that used in [Mov19] to perform the worst-to-average-case reduction. Interestingly, as discussed below, we find that the interpolation scheme of [Bou+19] cannot be used to prove hardness results about our algorithms' performance on a shallow random 2D quantum circuit possessing worst-case hardness for computing output probabilities; this essentially is a consequence of how `SEBD` and `Patching` exploit the unitarity of the circuit to be simulated. While this observation may be of technical interest for future work on worst-to-average-case reductions for quantum circuit simulations, the alternative interpolation scheme of [Mov19] does not suffer from this limitation.

While [Bou+19; Mov19] prove hardness results for the near-exact computation of output probabilites of random circuits, it is ultimately desirable to prove hardness for the Random Circuit Sampling (RCS) problem of sampling from the output distribution of a random circuit with small error in variational distance, as this is the computational task corresponding to the problem that the quantum computer solves. *A priori*, one might hope that such a result could be proved via such a worst-to-average-case reduction. In particular, it was pointed out in these works that improving the error tolerance of the hardness result to $2^{-n}/\operatorname{poly}(n)$ would be sufficient to prove hardness of RCS. Our work rules out such a proof strategy working in general by showing that even improving the error tolerance to $2^{-n}/2^{n^c}$ for any constant $c < 1$ is unachievable. In particular, any proof of the hardness of RCS should be sensitive to the depth and should not be applicable to the worst-case-hard shallow random circuit ensembles that admit approximate average-case classical simulations.

5.3.1 Implications for reductions based on truncated Taylor series

In this section, we discuss the relation between our algorithms (`SEBD` and `Patching`) applied to the computation of output probabilities and the recent result [Bou+19] on the hardness of average-case simulation of random circuits based on polynomial interpolation via truncated Taylor series. In particular, we discuss how this polynomial interpolation argument is insufficient to show that the task of even *exactly* computing output probabilities and sampling from the output distribution of a constant-depth Haar-random circuit instance with high probability using our algorithms is classically hard, even though these circuits possess worst-case hardness. We first briefly

review their technique before discussing a limitation in the robustness of the polynomial interpolation scheme. We then discuss how this robustness limitation makes the interpolation scheme inapplicable to our algorithms.

The main point is that our algorithms exploit unitarity (via the fact that gates outside of the lightcone of the qudits currently under consideration are ignored), but the hardness result of [Bou+19] holds with respect to circuit families that are non-unitary, albeit very close to unitary in some sense. Our algorithms are unable to simulate these slightly non-unitary circuits to the precision required for the worst-to-average case reduction, regardless of how well they can simulate Haar-random circuit families. While it is true that in this scheme there is an adjustable parameter K which, when increased, brings the non-unitary circuit family closer to approximating the true Haar-random family, increasing K also increases the degree of the interpolating polynomial. This makes the interpolation more sensitive to errors in such a way that, for any choice of K, the robustness that the interpolation can tolerate is not large enough to overcome the inherent errors that our algorithms make when trying to simulate these non-unitary families. The existence of simulation algorithms like `SEBD` and `Patching`, which exploit the unitarity of the circuit, may present an obstruction to applying worst-to-average-case reduction techniques that obtain a polynomial structure at the expense of unitarity. Note that, as discussed previously, a very recent alternative worst-to-average case reduction [Mov19] based on "Cayley paths" rather than truncated Taylor series does not suffer from this same limitation.

5.3.1.1 Background: truncated Haar-random circuit ensembles and polynomial interpolation

In this section, we give an overview (omitting some details) of the interpolation technique of [Bou+19] used to show their worst-to-average-case reduction, partially departing from their notation. Suppose U is a unitary operator. Then we define the θ-contracted and K-truncated version of U to be $U'(\theta, K) = U \sum_{k=0}^{K} \frac{(-\theta \ln U)^k}{k!}$. Note that $U'(\theta, \infty) = Ue^{-i\theta(-i \ln U)}$ is simply U pulled-back by angle θ towards the identity operator I. Note that $U'(0, \infty) = U$ and $U'(1, \infty) = I$. For $U'(\theta, K)$ for $K < \infty$, the operator that performs this pullback is then approximated by a Taylor series which is truncated at order K. If $K < \infty$, $U'(\theta, K)$ is (slightly) non-unitary.

Suppose C is some circuit family for which computing output probabilities up to error $2^{-\operatorname{poly}(n)}$ is classically hard. Now, for each gate G in C, multiply that gate by $H'(\theta, K)$ with H Haar-distributed and supported on the same qubits as G. This yields some distribution over non-unitary circuits that we call $\mathcal{D}(C, \theta, K)$. Note that if $\theta = 0$, $\mathcal{D}$ exactly becomes the Haar-random circuit distribution with the same architecture as C. When $\theta = 1$, the hard circuit C is recovered up to some small correction due to the truncation. If K is sufficiently large, we can assume that computing output probabilities for this slightly perturbed version of C is also classically hard.

Fix some circuit A drawn from $\mathcal{H}(C)$, the distribution over circuits with the same architecture as C with gates chosen according to the Haar measure. Let $A(C, \theta, K)$ denote the circuit obtained when the θ-pulled-back and K-truncated gates of A are multiplied with their corresponding gates in C. Note that $A(C, \theta, K)$ is distributed

as $\mathcal{D}(C, \theta, K)$. Define the quantity

$$p_0(A, \theta, K) := |\langle 0|A(C, \theta, K)|0\rangle|^2. \tag{5.62}$$

Assuming the circuit C has m gates, it is easy to verify that $p_0(A, \theta, K)$ may be represented as a polynomial in θ of degree $2mK$. Note also that $p_0(A, 1, \infty) = p_0(C)$, which is assumed to be classically hard to compute.

Now, assume that there exists some classical algorithm $\mathcal{A}$ and some $\epsilon = 1/\operatorname{poly}(n)$ such that, for some fixed $K \leq \operatorname{poly}(n)$ and for all $0 \leq \theta \leq \epsilon$, $\mathcal{A}$ can compute $p_0(A, \theta, K)$ up to additive error $\delta \leq 2^{-n^c}$ for some constant c, with probability $1 - 1/\operatorname{poly}(n)$ over $A(C, \theta, K) \sim \mathcal{D}(C, \theta, K)$. Then, $\mathcal{A}$ may evaluate $p_0(A, \theta, K)$ for $2mK + 1$ evenly spaced values of θ in the range $[0, \epsilon]$ (up to very small error), and construct an interpolating polynomial $q_0(A, \theta, K)$. By a result of Rakhmanov [Rak07], there is some interval $[a, b] \subset [0, \epsilon]$ such that $b - a \geq 1/\operatorname{poly}(n)$ and $|p_0(A, \theta, K) - q_0(A, \theta, K)| \leq 2^{-n^{c'}}$ for $\theta \in [a, b]$ where c' depends on c. One then invokes the following result of Paturi.

Lemma 31 ([Pat92])**.** *Let $p : \mathbb{R} \to \mathbb{R}$ be a real polynomial of degree d, and suppose $|p(x)| \leq \delta$ for all $|x| \leq \epsilon$. Then $|p(1)| \leq \delta e^{2d(1+1/\epsilon)}$.*

Applying this result, we find $|p_0(A, 1, K) - q_0(A, 1, K)| \leq 2^{-n^{c'}} e^{\operatorname{poly}(n,m,K)}$. If c is sufficiently large, then $|p_0(A, 1, K) - q_0(A, 1, K)| \leq 2^{-\operatorname{poly}(n)}$ and the quantity $q_0(A, 1, K)$ is hard to compute classically. But this would be a contradiction, because $q_0(A, 1, K)$ can be efficiency evaluated classically by performing the interpolation.

Hence, this argument shows that for some choice of K and a sufficiently large c depending on K, computing output probabilities of circuits in the truncated families $\mathcal{D}(C, \theta, K)$ with $\theta \leq 1/\operatorname{poly}(n)$ up to error 2^{-n^c} is hard (assuming standard hardness conjectures).

5.3.1.2 Limitation of the interpolation argument

The above argument shows that the average-case simulation of some family $\mathcal{D}(C, \theta, K)$ of non-unitary circuits which in some sense is close to the corresponding Haar-random circuit family to precision $2^{-\operatorname{poly}(n)}$ is classically hard, if simulating C is classically hard and the polynomial in the exponent is sufficiently large.

We now explain how, based on this argument, we are unable to conclude that exactly computing output probabilities of Haar-random circuits is classically hard.[1] In other words, suppose that with probability $1 - 1/\operatorname{poly}(n)$, some algorithm $\mathcal{A}$ can *exactly* compute output probabilities from the distribution $\mathcal{H}(C)$. We argue that a straightforward application of the above result based on Taylor series truncations and polynomial interpolation is insufficient to compute $p_0(C)$ with small error.

Consider some circuit realization A drawn from $\mathcal{H}(C)$, and assume that we can exactly compute its output probability $p_0(A)$. To use the argument of [Bou+19], we actually need to compute $p_0(A, \theta, K)$ for some fixed value of K and θ in some

[1] A simplified and slightly weaker version of our argument was also reported in [Mov19].

range $[0, \epsilon]$. We first find an upper bound for ϵ which must be satisfied for the interpolation to be guaranteed to succeed with high probability. To this end, we note that [Bou+19] the total variation distance between the distributions $\mathcal{D}(C, \theta, \infty)$ and $\mathcal{D}(C, 0, \infty)$ is bounded by $O(m\theta)$. Hence, if we try to use the algorithm $\mathcal{A}$ to estimate $p_0(A, \theta, \infty)$, the failure probability over random circuit instances could be as high as $O(m\theta)$. Therefore, since the θ values to be evaluated are uniformly spaced on the interval $[0, \epsilon]$, the union bound tells us that the probability that one of the $2mK+1$ values $p_0(A, \theta, K)$ is erroneously evaluated is bounded by $O(m^2K\epsilon)$. Hence, in order to ensure that all $2mK + 1$ points are correctly evaluated, we should take $\epsilon \leq O(1/m^2K)$.

Now, assume that we have chosen $\epsilon \leq O(1/m^2K)$ and all $2mK + 1$ points $p_0(A, \cdot, \infty)$ are correctly evaluated. Let θ be one of the evaluation points. We now must consider the error made by approximating the "probability" associated with the truncated version of the circuit with the probability associated with the untruncated version of the circuit, namely $|p_0(A, \theta, \infty) - p_0(A, \theta, K)|$. This error associated with the truncated Taylor series is upper bounded by $\delta \leq \frac{2^{O(nm)}}{K!}$ [Bou+19].

Plugging these values into Lemma 31, we find that if we use these values to try to interpolate to the classically hard-to-compute quantity $p_0(C, 1, K)$, the error bound guaranteed by Paturi's lemma is no better than $\frac{2^{O(nm)}}{K!} \exp\left(O(2mK(1 + m^2K))\right)$, which diverges in the limit $n \to \infty$ for any scaling of m and K. Hence, the technique of [Bou+19] is insufficient to show that exactly computing output probabilities of circuits drawn from the Haar-random circuit distribution $\mathcal{H}_C$ with high probability is hard.

Intuitively, the limitation arises because there is a tradeoff between the amount of truncation error incurred and the degree of the interpolating polynomial. As the parameter K is increased, the truncation error is suppressed, but the degree of the interpolating polynomial is increased, making the interpolation more sensitive to errors.

5.3.1.3 Inapplicability to `SEBD` and `Patching`

To summarize the findings above, the argument of [Bou+19] for the hardness of computing output probabilities of random circuits applies not directly to Haar-random circuit distributions, but rather to distributions over slightly non-unitary circuits that are exponentially close to the corresponding Haar distributions in some sense. We argued that the interpolation scheme cannot be straightforwardly applied to circuits that are truly Haar-random, and therefore it cannot be used to conclude that simulating truly Haar-random circuits, even exactly, is classically hard.

A priori, it is not obvious whether this limitation is a technical artifact or a more fundamental limitation of the interpolation scheme. In particular, one might imagine that if some algorithm $\mathcal{A}$ is capable of exactly simulating Haar-random circuit families, some modified version of the algorithm $\mathcal{A}'$ might be capable of simulating the associated truncated Haar-random circuit families, at least up to the precision needed for the interpolation argument to work. If this were the case, then the hardness argument *would* be applicable.

However, SEBD and Patching appear to be algorithms that *cannot* be straightforwardly used to efficiently simulate truncated Haar-random circuit families to the precision needed for the interpolation to work, even under the assumption that they can efficiently, exactly simulate Haar-random circuit families. This is because the efficiency of these algorithms crucially relies on the existence of a constant-radius lightcone for constant-depth circuits. The algorithm is able to ignore all qubits and gates outside of the lightcone of the sites currently being processed. However, the lightcone argument breaks down for non-unitary circuits. If the gates are non-unitary and we want to perform an exact simulation, we are left with using Markov-Shi or some other general-purpose tensor network contraction algorithm, with a running time of $2^{O(d\sqrt{n})}$ for a depth-d circuit on a square grid of n qubits.

Consider what happens if one tries to use one of these algorithms to compute output "probabilities" for a slightly non-unitary circuit coming from a truncated Haar-random distribution $\mathcal{D}(C, \theta, K)$, and then use these computed values to interpolate to the hard-to-compute value $p_0(C, 1, K)$ via the interpolating polynomial of degree $2mK$ proposed in [Bou+19]. Even without any other sources of error, when one of these algorithms ignores gates outside of the current lightcone, it is essentially approximating each gate outside the lightcone as unitary. This causes an incurred error bounded by $2^{O(nm)}/K!$ for the computed output probability. Then, by an argument essentially identical to the one appearing in the previous section, one finds that this error incurred just from neglecting gates outside the lightcone is already large enough to exceed the error permitted for the polynomial interpolation to be valid. We conclude that this worst-to-average-case reduction based on truncated Taylor series expansions cannot be used to conclude that it is hard for SEBD or Patching to exactly simulate worst-case hard shallow Haar-random circuits with high probability.

5.4 Deferred proofs

Lemma 21. *Let ϵ_i denote the sum of the squares of all singular values discarded in the compression during iteration i of the simulation of a circuit C with output distribution $\mathcal{D}_C$ by SEBD with no bond dimension cutoff, and let Λ denote the sum of all singular values discarded over the course of the algorithm. Then the distribution $\mathcal{D}'_C$ sampled from by SEBD satisfies*

$$\frac{1}{2}\|\mathcal{D}'_C - \mathcal{D}_C\|_1 \leq \mathbb{E}\sum_{i=1}^{L_2}\sqrt{2\epsilon_i} \leq \sqrt{2}\,\mathbb{E}\,\Lambda, \tag{4.4}$$

where the expectations are over the random measurement outcomes.

Proof. We rely upon a well-known fact about the error caused by truncating the bond dimension of a MPS, which we state in Lemma 32.

Lemma 32 (follows from [VC06])**.** *Suppose the MPS $|\psi\rangle$ is compressed via truncation of small singular values, and ϵ is the sum of the squares of the discarded singular*

values. Then if $\left|\psi^{(t)}\right\rangle$ *is the truncated version of the MPS after normalization,*

$$\||\psi\rangle\langle\psi| - |\psi^{(t)}\rangle\langle\psi^{(t)}|\|_1 \leq \sqrt{8\epsilon}. \tag{5.63}$$

The second inequality follows from the fact that $\sqrt{\sum_i x_i^2} \leq \sum_i x_i$ for $x_i \geq 0$. To prove the first inequality, we start by considering the version of the algorithm with no truncation, which we have argued samples exactly from $\mathcal{D}$. Let $\mathcal{N}_t$ denote the TPCP map corresponding to the application of gates that have come into the lightcone of `column` t and the measurement of `column` t. That is,

$$\mathcal{N}_t(\rho) = \sum_{\mathbf{x}_t} \Pi_t^{\mathbf{x}_t} V_t \rho V_t^\dagger \Pi_t^{\mathbf{x}_t}, \tag{5.64}$$

where $\mathbf{x}_t$ indexes (classical) outcome strings of `column` t. Note that $\mathcal{N}_t(\rho)$ is a classical-quantum state for which the sites corresponding to the first t columns are classical, and the quantum register consists of sites which are in the lightcone of `column` t but not in the first t columns. Define $\rho_t := \mathcal{N}_{t-1}(\rho_{t-1})$ and $\rho_1 := |1\rangle\langle 1|^{\otimes L_1}_{\texttt{column 1}}$, so that ρ_{L_2+1} is a classical state exactly corresponding to output strings on the $L_1 \times L_2$ grid distributed according to $\mathcal{D}$.

Now consider the "truncated" version of the algorithm, which is defined similarly except we use σ_t to denote the state of the algorithm immediately after the truncation at the beginning of iteration t. That is, we define

$$\sigma_t := (T_t \circ \mathcal{N}_{t-1})(\sigma_{t-1}), \tag{5.65}$$

where T_t denotes the mapping corresponding to the MPS truncation and subsequent renormalization at the beginning of iteration t, and we define $\sigma_1 := T_1(\rho_1) = \rho_1$ (there is no truncation at the beginning of the first iteration since the initial state is a product state).

We now have

$$\begin{aligned}
\|\mathcal{D}_C - \mathcal{D}'_C\|_1 &= \|\rho_{L_2+1} - \sigma_{L_2+1}\|_1 && (5.66)\\
&\leq \|\rho_{L_2+1} - \mathcal{N}_{L_2}(\sigma_{L_2})\|_1 + \|\mathcal{N}_{L_2}(\sigma_{L_2}) - \sigma_{L_1+1}\|_1 && (5.67)\\
&\leq \|\rho_{L_2} - \sigma_{L_2}\|_1 + \|\mathcal{N}_{L_2}(\sigma_{L_2}) - \sigma_{L_2+1}\|_1, && (5.68)
\end{aligned}$$

where the first inequality follows from the triangle inequality, and the second from contractivity of TPCP maps. Applying this inequality recursively yields

$$\begin{aligned}
\|\mathcal{D}_C - \mathcal{D}'_C\|_1 &\leq \sum_{i=1}^{L_2} \|\mathcal{N}_i(\sigma_i) - \sigma_{i+1}\|_1 && (5.69)\\
&= \sum_{i=1}^{L_2-1} \|\mathcal{N}_i(\sigma_i) - (T_{i+1} \circ \mathcal{N}_i)(\sigma_i)\|_1 && (5.70)
\end{aligned}$$

where we also used the fact that no truncation occurs after $\mathcal{N}_{L_2}$ is applied (i.e. T_{L_2+1} acts as the identity). Now, note that $\|\mathcal{N}_i(\sigma_i) - (T_{i+1} \circ \mathcal{N}_i)(\sigma_i)\|_1$ is ex-

actly the expected error in 1-norm caused by the truncation in iteration $i+1$. (This is true because of the following fact about classical-quantum states: $\left\| \mathbb{E}_i |i\rangle\langle i|_C \otimes (|\psi_i\rangle\langle\psi_i|_Q - |\phi_i\rangle\langle\phi_i|_Q) \right\|_1 = \mathbb{E}_i \left\| |\psi_i\rangle\langle\psi_i| - |\phi_i\rangle\langle\phi_i| \right\|_1$ where $\{|i\rangle_C\}_i$ is an orthonormal basis for the Hilbert space associated with register C.) By Lemma 32, this quantity is bounded by $\mathbb{E}\sqrt{8\epsilon_{i+1}}$. Substituting this bound into the summation yields the desired inequality. □

Lemma 25. *Let $\lambda_1 \geq \lambda_2 \geq \cdots$ denote the half-chain Schmidt values after at least $n/2$ iterations of the toy model process. Then with probability at least $1-\delta$ the half-chain Schmidt values indexed by $i \geq i^* = \exp\Big(\Theta(\sqrt{\log(n/\delta)})\Big)$ obey the asymptotic scaling*

$$\lambda_i \propto \exp\big(-\Theta(\log^2(i))\big). \tag{4.14}$$

Furthermore, upon truncating the smallest Schmidt coefficients up to a truncation error of ϵ, with probability at least $1-\delta$, the half-chain Schmidt rank r of the post-truncation state obeys the scaling

$$r \leq \exp\Big(\Theta\Big(\sqrt{\log(n/\epsilon\delta)}\Big)\Big). \tag{4.15}$$

Proof. Suppose that an EPR pair is measured $2t$ times, corresponding to each of the two qubits being measured t times. A calculation shows that the probability of obtaining s M_1 outcomes is given by a mixture of two binomial distributions. Letting S be the random variable denoting the number of M_1 outcomes, we find

$$\Pr[S=s] = \frac{1}{2}\Pr\big[B_{2t,\sin^2(\theta/2)} = s\big] + \frac{1}{2}\Pr\big[B_{2t,\cos^2(\theta/2)} = s\big], \tag{5.71}$$

where $B_{n,p}$ denotes a binomial random variable associated with n trials and success probability p. If after the $2t$ measurements we obtain outcome M_1 s times, the post-measurement state is given by (up to normalization)

$$|00\rangle + \tan^{2(t-s)}(\theta/2)\,|11\rangle\,. \tag{5.72}$$

Note that s can be assumed to be generated by sampling from either $B_{2t,\sin^2(\theta/2)}$ or $B_{2t,\cos^2(\theta/2)}$ with probability $1/2$ each. In the former case, the post-measurement state may be written

$$\begin{aligned} &|00\rangle + \tan^{2\left(t - B_{2t,\sin^2(\theta/2)}\right)}(\theta/2)\,|11\rangle \\ &\qquad = |00\rangle + \tan^{2t\cos(\theta) - 2X_{2t,\sin^2(\theta/2)}}(\theta/2)\,|11\rangle \end{aligned} \tag{5.73}$$

where we have defined the random variable $X_{2t,\sin^2(\theta/2)}$ via $B_{n,p} = np + X_{n,p}$. That is, the random variable $X_{n,p}$ is distributed as a binomial distribution shifted by its mean. Now, defining $\gamma := (\tan(\theta/2))^{2\cos(\theta)}$ and $X'_{n,p} = X_{n,p}/\cos(\theta)$, we may write the post-measurement state as

$$|00\rangle + \gamma^{t - X'_{2t,\sin^2(\theta/2)}}\,|11\rangle\,. \tag{5.74}$$

We assume WLOG that $0 < \theta < \pi/2$, so that $0 < \gamma < 1$. Similarly, if s is drawn from $B_{2t,\cos^2(\theta/2)}$, then the post-measurement state is given by

$$|00\rangle + \gamma^{-t-X'_{2t,\cos^2(\theta/2)}} |11\rangle . \tag{5.75}$$

Note that, under a relabeling of basis states $0 \leftrightarrow 1$, the post-measurement state in this case is

$$|00\rangle + \gamma^{t-X'_{2t,\sin^2(\theta/2)}} |11\rangle , \tag{5.76}$$

where we used the fact that $-X'_{2t,\cos^2(\theta/2)}$ is distributed identically to $X'_{2t,\sin^2(\theta/2)}$. Since we will be interested in studying the entanglement spectrum of this process, which is invariant under such local basis changes, we may assume WLOG that the random post-measurement state after $2t$ measurements is given by $|00\rangle + \gamma^{t-X'_{2t,\sin^2(\theta/2)}} |11\rangle$.

We can then model the final state as

$$\bigotimes_t |00\rangle + \gamma^{t-X'_{2t,\sin^2(\theta/2)}} |11\rangle \tag{5.77}$$

up to normalization. This allows an estimate of the tradeoff between rank, truncation error, and associated probability of success.

Let $Q(\ell)$ denote the number of "strict partitions" of ℓ, i.e. the number of ways of writing $\ell = t_1 + t_2 + \dots$ for positive integers $t_1 < t_2 < \dots$. Precise asymptotics are known for $Q(\ell)$ (see `https://oeis.org/A000009` and [Ayo63]):

$$Q(\ell) = \exp\Big(\Theta(\sqrt{\ell})\Big). \tag{5.78}$$

By expanding Equation (5.77) as a superposition over computational basis states, we obtain the unnormalized Schmidt coefficients $\tilde{\lambda}_1 \geq \tilde{\lambda}_2 \geq \cdots$; each coefficient in the expansion gives an unnormalized Schmidt coefficient. There are $Q(\ell)$ unnormalized Schmidt coefficients that are distributed as $\gamma^{\ell - X'_{2\ell,\sin^2(\theta/2)}}$, where we used the fact that $X'_{t_1,\sin^2(\theta/2)} + X'_{t_2,\sin^2(\theta/2)}$ is distributed as $X'_{t_1+t_2,\sin^2(\theta/2)}$. We say that these $Q(\ell)$ coefficients live in sector ℓ. For a fixed probability p, let $K_{\ell,p}$ denote the smallest positive integer for which, with probability at least $1-p$, all sector-ℓ coefficients lie in the range $[\gamma^{\ell+K_{\ell,p}}, \gamma^{\ell-K_{\ell,p}}]$. By the union bound, to upper bound $K_{\ell,p}$ it suffices to find an integer a for which

$$\Pr\Big[\Big|X'_{2\ell,\sin^2(\theta/2)}\Big| \geq a\Big] \leq \frac{p}{Q(\ell)} = p\exp\Big(-\Theta(\sqrt{\ell})\Big). \tag{5.79}$$

By Hoeffding's inequality, we have $\Pr\Big[\Big|X'_{2\ell,\sin^2(\theta/2)}\Big| \geq a\Big] \leq \exp(-\Theta(a^2/\ell))$; this yields the bound

$$K_{\ell,p} \leq \Theta\left(\sqrt{\ell\log(1/p) + \ell\sqrt{\ell}}\right). \tag{5.80}$$

Furthermore, note that since there are $\Theta(n^2)$ sectors, by the union bound, with probability at least $1-\delta$, for each sector j, all coefficients lie in the range $[\gamma^{j+K_{j,p}}, \gamma^{j-K_{j,p}}]$ if

we take p to be $p = \delta/\Theta(n^2)$. We make this choice of p and assume for the remainder of the argument that all coefficients of sector j lie in the given range, which is true with probability at least $1 - \delta$. We also note the following fact which will be used below: if ℓ and p are related as $\ell \geq \Theta(\log(1/p))$, then $K_{\ell,p} = O(\ell)$.

Still working with the unnormalized state of Equation (5.77), we now study the scaling between the Schmidt index i and corresponding coefficient $\tilde{\lambda}_i$ for i in the regime $i \geq \exp\Big(\Theta(\sqrt{\log(1/p)})\Big)$. Note that $\tilde{\lambda}_i = \gamma^\ell$ for some integer ℓ. We first lower bound ℓ. Note that the lower bound is achieved if, for each sector j, all coefficients in that sector are equal to $\gamma^{j-K_{j,p}}$. In this case, the exponent ℓ is equal to $\ell' - K_{\ell',p}$, where ℓ' is the smallest integer such that

$$i \leq \sum_{j=1}^{\ell'} Q(\ell') = \exp\Big(\Theta(\sqrt{\ell'})\Big). \tag{5.81}$$

Rearranging, we see that $\ell' = \Theta(\log^2(i)) \geq \Theta(\log(1/p))$, and hence $\ell = \Theta(\log^2(i))$ since $\ell' - K_{\ell',p} = \Theta(\ell')$. Similarly , an upper bound on ℓ is achieved if, for each sector j, all coefficients in that sector are equal to $\gamma^{j+K_{j,p}}$. In this case, ℓ is equal to $\ell' + K_{\ell',p}$, where ℓ' is defined as above. This yields a matching upper bound for ℓ of $\Theta(\log^2(i))$. We therefore have the scaling $\ell = \Theta(\log^2(i))$, which, using the fact that $\tilde{\lambda}_i = \gamma^\ell$ yields

$$\tilde{\lambda}_i = \exp\big(\Theta(-\log^2(i))\big),\ i \geq \exp\Big(\Theta(\sqrt{\log(1/p)})\Big). \tag{5.82}$$

Noting that λ_i is proportional to $\tilde{\lambda}_i$ via $\lambda_i = \frac{1}{N}\tilde{\lambda}_i$ with $N = \sqrt{\sum_i \tilde{\lambda}_i^2}$, this shows the first statement of the lemma.

Now, suppose that for some $i \geq i^* = \exp\Big(\Theta(\sqrt{\log(1/p)})\Big)$, we truncate all Schmidt coefficients with index $\geq i$. The incurred truncation error is

$$\epsilon = \sum_{j \geq i} \lambda_j^2 < \sum_{j \geq i} \tilde{\lambda}_j^2 = \exp\big(-\Theta(\log^2(i))\big) \tag{5.83}$$

where the inequality holds because the unnormalized state has norm strictly greater than one (i.e. $N > 1$). Rearranging, this becomes

$$i \leq \exp\Big(\Theta(\sqrt{\log(1/\epsilon)})\Big). \tag{5.84}$$

Hence, if we truncate the state at the end of the process up to a truncation error of ϵ, the rank r of the post-truncation state is bounded by

$$r \leq \max\left(\exp\Big(\Theta(\sqrt{\log(1/\epsilon)})\Big), \exp\Big(\Theta(\sqrt{\log(1/p)})\Big)\right) \tag{5.85}$$

$$= \exp\left(\Theta\left(\sqrt{\log\left(\frac{n}{\epsilon \cdot \delta}\right)}\right)\right) \tag{5.86}$$

as desired, where we used the relation $p = \delta/\Theta(n^2)$. □

Lemma 27. *Suppose a 1D random circuit C is applied to qubits $\{1,\dots,n\}$ consisting of a layer of 2-qubit Haar-random gates acting on qubits $(k,k+1)$ for odd $k \in \{1,\dots,n-1\}$, followed by a layer of 2-qubit Haar-random gates acting on qubits $(k,k+1)$ for even $k \in \{1,\dots,n-1\}$. Suppose the qubits of region $B := \{i,i+1,\dots,j\}$ for $j \geq i$ are measured in the computational basis, and the outcome b is obtained. Then, letting $|\psi_b\rangle$ denote the post-measurement pure state on the unmeasured qubits, and letting $A := \{1,2,\dots,i-1\}$ denote the qubits to the left of B,*

$$\mathbb{E}\, S(A)_{\psi_b} \leq c^{|B|} \tag{4.16}$$

for some universal constant $c < 1$, where the expectation is over measurement outcomes and choice of random circuit C.

Proof. We will use a smaller technical lemma, which we state and prove below.

Lemma 33. *Let $|\psi\rangle_{AB}$ be some state on subsystems A and B with subsystem B a qubit, and let $|H\rangle_{CD}$ be some two-qubit Haar-random state on subsystems C and D. Suppose a Haar-random two-qubit gate U is applied to subsystems B and C. If subsystem B is measured in the computational basis and outcome b is obtained, then the von Neumann entropy of the post-measurement state $|\psi_b\rangle_{ABCD}$ in subsystem A satisfies*

$$\mathop{\mathbb{E}}_{b,H,U} S(A)_{\psi_b} \leq c \cdot S(A)_{\psi} \tag{5.87}$$

for some constant $c < 1$, where the expectation is over the random measurement outcome, the random state $|H\rangle_{CD}$, and the Haar-random unitary U.

Proof. Consider the Schmidt decomposition $|\psi\rangle_{AB} = \sqrt{p}\,|e_1\rangle_A\,|f_1\rangle_B + \sqrt{1-p}\,|e_2\rangle_A\,|f_2\rangle_B$ where we assume WLOG that $p \geq 1/2$. We also assume that $p < 1$, because the statement is trivially true for any value of c when $p = 1$. Note that the entanglement entropy of this state is simply $S(A)_\psi = H_2(p)$ where $H_2(p) := -p\log p - (1-p)\log(1-p)$ is the binary entropy function. Let $M_0 := (\Pi_0 \otimes I)U$ and $M_1 := (\Pi_1 \otimes I)U$ denote the measurement operators acting on subsystems B and C, where Π_i denotes the projector onto the computational basis state $|i\rangle$ and U is the Haar-random unitary applied to subsystems B and C. Let X denote a random variable equal to 1 with probability p and equal to 2 with probability $1-p$. Let Y denote the measurement outcome of $\{M_0, M_1\}$ when applied to the state $|e_X\rangle_A\,|f_X\rangle_B\,|H\rangle_{C,D}$. The probability of obtaining measurement outcome b on the original state is simply $\Pr(Y=b)$, and the post-measurement state after obtaining outcome b is

$$\begin{aligned}
&\frac{1}{\sqrt{\Pr(Y=b)}}\Big(\sqrt{p\cdot\Pr(Y=b|X=1)}\,|e_1\rangle_A\,|b\rangle_B\,|\phi_{b,1}\rangle_{C,D} \\
&\qquad\qquad +\sqrt{(1-p)\cdot\Pr(Y=b|X=2)}\,|e_2\rangle_A\,|b\rangle_B\,|\phi_{b,2}\rangle_{C,D}\Big) \\
&= \sqrt{\Pr(X=1|Y=b)}\,|e_1\rangle_A\,|b\rangle_B\,|\phi_{b,1}\rangle_{C,D} + \sqrt{\Pr(X=2|Y=b)}\,|e_2\rangle_A\,|b\rangle_B\,|\phi_{b,2}\rangle_{C,D}
\end{aligned} \tag{5.88}$$

where $|\phi_{b,j}\rangle_{C,D}$ are normalized states on subsystems C and D. Define

$$\epsilon := \min_b |\langle \phi_{b,1} | \phi_{b,2} \rangle|^2. \tag{5.89}$$

Letting $\rho_{A,b}$ denote the reduced density matrix on subsystem A of the post-measurement state after obtaining measurement outcome b, the maximal eigenvalue of this matrix is lower bounded as $\lambda_{\max}(\rho_{A,b}) \geq \Pr(X = 1|Y = b) + \epsilon \Pr(X = 2|Y = b)$. (To see this, observe that the reduced density matrix on CD is $\sigma = \Pr(X = 1|Y = b) |\phi_{b,1}\rangle\langle\phi_{b,1}| + \Pr(X = 2|Y = b) |\phi_{b,2}\rangle\langle\phi_{b,2}|$, and the maximal eigenvalue is lower bounded as $\lambda_{\max}(\rho_{A,b}) = \lambda_{\max}(\sigma) \geq \langle \phi_{b,1} | \sigma | \phi_{b,1} \rangle \geq \Pr(X = 1|Y = b) + \epsilon \Pr(X = 2|Y = b)$). Furthermore, note that

$$\mathop{\mathbb{E}}_Y \lambda_{\max}(\rho_{A,Y}) \geq \mathop{\mathbb{E}}_Y [\Pr(X = 1|Y) + \epsilon \Pr(X = 2|Y)] \tag{5.90}$$

$$= p + \epsilon(1 - p). \tag{5.91}$$

Now, using concavity of the binary entropy function, we have

$$\mathop{\mathbb{E}}_Y S(A)_{\psi_Y} = \mathop{\mathbb{E}}_Y H_2(\lambda_{\max}(\rho_{A,Y})) \tag{5.92}$$

$$\leq H_2(\mathop{\mathbb{E}}_Y \lambda_{\max}(\rho_{A,Y})) \tag{5.93}$$

$$\leq H_2(p + \epsilon(1 - p)). \tag{5.94}$$

Consider the ratio $r(p, \epsilon) := \frac{H_2(p+\epsilon(1-p))}{H_2(p)}$. We want to argue that for any $\epsilon > 0$, $r(p, \epsilon)$ is bounded away from one on the interval $p \in [1/2, 1)$. This statement is clearly true for any p bounded away from one since H_2 is monotonically decreasing on the interval $[1/2, 1)$. Furthermore, it is straightforward to show $\lim_{p\to 1} r(p, \epsilon) = 1 - \epsilon$. Hence, we have

$$\frac{\mathbb{E}_Y S(A)_{\psi_Y}}{S(A)_\psi} \leq r(p, \epsilon) \leq c(\epsilon) \tag{5.95}$$

where $c(\epsilon) < 1$ unless $\epsilon = 0$. We now average both sides over the choice of Haar-random state on CD as well as the Haar-random unitary U acting on BC. Since the event $\epsilon > 0$ occurs with nonzero probability (in fact, with probability one), we have the strict inequality $\mathbb{E}_{H,U}[c(\epsilon)] := c < 1$, from which the desired inequality follows. □

We may assume that $i \neq 0$ and $j \neq n$, as in these cases we trivially have $S(\rho_A(b)) = 0$. The post-measurement state may be constructed as follows. Apply all gates in the lightcone of qubit i, then measure qubit i. Now apply all gates in the lightcone of qubit $i + 1$ not previously applied, then measure qubit $i + 1$. Assume that qubits are introduced only when they come into the lightcone under consideration. Iterate until all qubits in region B have been measured. Finally, apply any gates that have not yet been applied. It is straightforward to verify that this is equivalent to applying all gates of the circuit before performing the measurement of region B, in the sense that the measurement statistics are the same, and the post-measurement state given some outcome b is the same.

By Lemma 33, after the first iteration we are left with the state $|\psi\rangle_{LR}|b_{i_1}\rangle_{i_1}$, such that $\mathbb{E}S(L)_\psi \leq c$ for some constant $c < 1$. In all iterations, we let L denote the current subsystem to the left of the measured qubits, and R denote the subsystem to the right of the measured qubits. Now consider the second iteration. Depending on whether i was even or odd, R may consist of one or two qubits immediately after the measurement of i. In the former case, we may apply Lemma 33 again, obtaining $\mathbb{E}S(L)_\psi \leq c^2$ after the measurement of qubit $i+1$, and obtaining a two-qubit subsystem to the right of the measured qubits. In the latter case, as a consequence of concavity of von Neumann entropy, we have $\mathbb{E}S(L)_\psi \leq c$ after measurement, and are left with a one-qubit subsystem to the right of the measured qubits. Iterating this process, after all qubits of subregion B have been measured, we are left with some state $|\psi\rangle_{LR}$ such that $\mathbb{E}\, S(L)_\psi \leq c^{|B|/2} \leq c'^{|B|}$ where $c' = \sqrt{c} < 1$. Finally, local unitary gates are applied to $|\psi\rangle_{LR}$ to obtain the final post-measurement state on the entire chain. Since each unitary is applied to only the left of region B or only the right of region B, the entanglement entropy across the (A, A^c) cut is unaffected by these gates, and remains bounded by $c^{|B|}$ in expectation. □

Lemma 28. *Let C be an instance of* `Brickwork`(L, r, v). *Then, with probability at least $1 - 2^{-\Theta(r)}$ over the circuit instance,* `SEBD` *running with maximal bond dimension cutoff $D = \Theta(1)$ and truncation error parameter $\epsilon = 2^{-\Theta(r)}$ can be used to (1) sample from the output distribution of C up to error $n2^{-\Theta(r)}$ in variational distance and (2) compute the output probability of an arbitrary output string up to additive error $n2^{-\Theta(r)}/2^n$ in runtime $\Theta(n)$.*

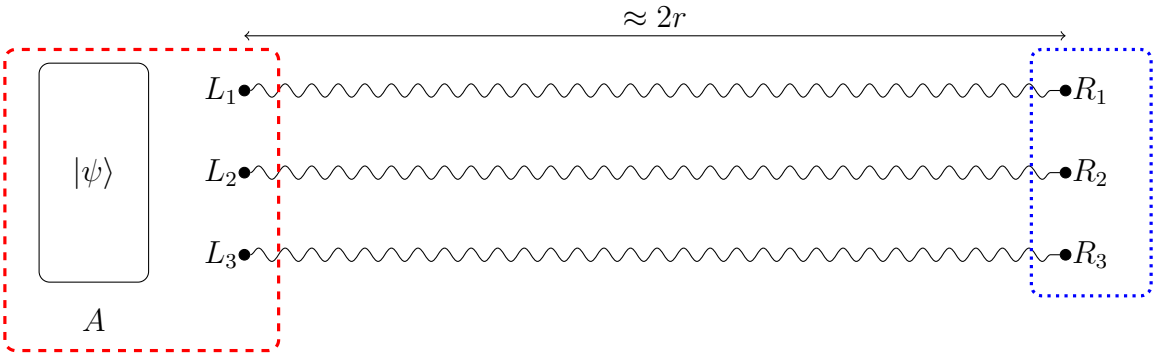


Figure 5-6: Illustration of the state after the qubits of columns $i, i+1, \ldots, j$ have been measured, but before gates in the lightcone of registers A and L have been performed. In each row i, we are left with a post-measurement bipartite state $|\phi_i\rangle_{L_iR_i}$ depicted by a wavy line. The expected entanglement entropy $S(L_i)_{\phi_i}$ decays exponentially in r. The final state of interest $|\psi'\rangle$ is obtained by applying local unitaries to the qubits in the dashed red box before measuring all of these qubits in the computational basis, inducing the final state $|\psi'\rangle$ on $R = R_1 \cup \cdots \cup R_L$. By concavity of the von Neumann entropy, the expected entanglement entropy of $|\psi'\rangle$ across the cut defined by the dotted blue box is upper bounded by the entanglement entropy across this cut before the unitaries and measurements in the dashed red box are performed.

Proof. Suppose the state stored by SEBD immediately before entering into a 1-local region is $|\psi\rangle_A$, defined on register A. After another $O(r)$ iterations of SEBD, just before the end of the 1-local region, denote the new one-dimensional state stored by SEBD as $|\psi'\rangle$. Note that $|\psi'\rangle$ is a random state, depending on both the random choices of gates in the 1-local region and the random measurement outcomes. We now bound the expected entanglement entropy of $|\psi'\rangle$ across an arbitrary cut.

To this end, we observe that the random final state $|\psi'\rangle$ may be equivalently generated as follows. Instead of iterating SEBD as usual for $O(r)$ iterations, we first introduce a contiguous block of qubits that lie in the 1-local region. In particular, for all rows, we introduce all qubits that lie in columns $\{i, i+1, \ldots, j\}$. Here, i is chosen to be the leftmost column such that the lightcone of column i does not contain qubits in register A. Similarly, j is chosen to be the rightmost column such that the lightcone of qubits in column j does not contain vertical gates. Note that $|i - j| = \Theta(r)$.

We next apply all gates in the lightcone of the qubits of columns $\{i, i+1, \ldots, j\}$, before measuring these qubits in the computational basis. Note that in this step, we are effectively performing a set of L one-dimensional depth-2 Haar-random circuits, and then measuring $\Theta(r)$ intermediate qubits for each of the L instances. For each instance, we are left with a (generically entangled) pure state between a "left" and "right" subsystem, as illustrated in Figure 5-6. Let L_i (R_i) denote the left (right) subsystem associated with row i, and let $|\phi_i\rangle_{L_i R_i}$ denote the associated post-measurement pure state on these subsystems. By Lemma 27, it follows that the expected entanglement entropy for any 1D instance obeys $\mathbb{E}\, S(L_i)_{\phi_i} \leq 2^{-\Theta(r)}$ where the expectation is over random circuit instance and measurement outcomes.

The next step is to apply all gates in the lightcone of the qubits of registers A and $L := \cup_i L_i$ before measuring these registers, inducing a (random) 1D post-measurement pure state on subsystem $R := \cup_i R_i$. It is straightforward to verify that the distribution of the random 1D pure state $|\psi'\rangle_R$ obtained via this procedure is identical to that obtained from repeatedly iterating SEBD through column j[2]. Indeed, the procedures are identical up to performing commuting gates and commuting measurements in different orders, which does not affect the measurement statistics or post-measurement states.

Our goal is now to bound the entanglement entropy $S(R_1 R_2 \ldots R_k)_{\psi'}$ in expectation across an arbitrary cut of the post-measurement 1D state. Such a bound follows from the concavity of the von Neumann entropy. Let $\rho_{R_1,\ldots,R_k}$ denote the reduced density matrix on these subsystems before the measurements on A and L are performed. Let $\rho^x_{R_1,\ldots,R_k}$ denote the reduced density matrix on these subsystems after the measurements on A and L are performed and the outcome x is obtained; note that the final state ψ' implicitly depends on x. Now, letting $\Pr[x]$ denote the probability of obtaining outcome x, we have the relation $\sum_x \Pr[x] \rho^x_{R_1 \cdots R_k} = \rho_{R_1 \cdots R_k}$. To see this, observe that for any set of measurement operators $\{M^x\}_x$ satisfying $\sum_x M^{x\dagger} M^x = I$, we have $\rho_{R_1 \cdots R_k} = \mathrm{tr}_{\setminus R_1 \cdots R_k}(|\psi'\rangle\langle\psi'|) =$

[2]Strictly speaking, we are actually studying a version of SEBD that only performs the MPS compression step at the end of a 1-local region. Since 1-local operations cannot increase the bond dimension of the associated MPS, the algorithm can forego the compression steps during the 1-local regions without incurring a bond dimension increase.

$\sum_x \mathrm{tr}_{\setminus R_1 \cdots R_k} \left(M^x \left| \psi' \right\rangle \! \left\langle \psi' \right| M^{x\dagger} \right) = \sum_x \Pr[x] \frac{\mathrm{tr}_{\setminus R_1 \cdots R_k} \left(M^x |\psi'\rangle\langle\psi'| M^{x\dagger} \right)}{\mathrm{tr} \left(M^x |\psi'\rangle\langle\psi'| M^{x\dagger} \right)} = \sum_x \Pr[x] \rho^x_{R_1 \cdots R_k}$. Now,

$$\begin{aligned}
&\sum_x \Pr[x] S(R_1 \ldots R_k)_{\psi'} \\
&= \sum_x \Pr[x] S(\rho^x_{R_1,\ldots,R_k}) && (5.96) \\
&\leq S\left(\sum_x \Pr[x] \rho^x_{R_1,\ldots,R_k} \right) && (5.97) \\
&= S(\rho_{R_1,\ldots,R_k}) && (5.98) \\
&= \sum_{i=1}^k S(R_i)_{\phi_i} && (5.99)
\end{aligned}$$

where the first line follows by definition, the second line follows from concavity of the von Neumann entropy, the third line uses the relation we discussed previously, and in the final line we used the fact that $\rho_{R_1,\ldots,R_k}$ is a product state. Hence, we see that for any fixed set of gates and for any outcomes of the measurements of qubits in columns $i, i+1, \ldots, j$, the expected entanglement entropy of the final 1D state ψ' on R across any cut is bounded by the entropy across that cut before the measurements on subregions A and L. Taking the expectations of both sides of this result with respect to the random gates and measurement outcomes of the qubits in columns $i, i+1, \ldots, j$, we finally obtain

$$\mathbb{E}\, S(R_1 \ldots R_k)_{\psi'} \leq L 2^{-\Theta(r)} \tag{5.100}$$

where we used the fact that $k < L$ and $\mathbb{E}\, S(R_i)_{\phi_i} \leq 2^{-\Theta(r)}$. We now use the fact that the largest eigenvalue $\lambda_{\max}(R_1 \cdots R_k)$ of the reduced density matrix is lower bounded as $\lambda_{\max}(R_1 \cdots R_k)_{\psi'} \geq 2^{-S(R_1 \cdots R_k)_{\psi'}}$; this follows from the fact that Shannon entropy upper bounds min-entropy. Using this inequality as well as Jensen's inequality, we have the bound

$$\begin{aligned}
\mathbb{E}\, \lambda_{\max}(R_1 \cdots R_k) &\geq \mathbb{E}\, 2^{-S(R_1 \cdots R_k)_{\psi'}} && (5.101) \\
&\geq 2^{-\mathbb{E}\, S(R_1 \cdots R_k)_{\psi'}} && (5.102) \\
&\geq 2^{-L 2^{-\Theta(r)}} && (5.103) \\
&\geq 1 - L 2^{-\Theta(r)}. && (5.104)
\end{aligned}$$

Therefore, if we truncate all but the largest Schmidt coefficient across the $R_k : R_{k+1}$ cut, we incur an expected truncation error upper bounded by $L 2^{-\Theta(r)}$. Hence, by Markov's inequality, we incur a truncation error upper bounded by $L 2^{-\Theta(r)}$ with probability at least $1 - 2^{-\Theta(r)}$.

Therefore, if we run `SEBD` using a *per bond* truncation error of $\epsilon = L 2^{-\Theta(r)}$ and a maximum bond dimension cutoff of $D = O(1)$, the failure probability will be upper

bounded by $Lv2^{-\Theta(r)}$; here we used the union bound to upper bound the probability that any of the $O(Lv)$ bonds over the course of the algorithm becomes larger than the cutoff D. Hence, by Corollary 1, for at least $1 - 2^{-\Theta(r)}$ fraction of random circuit instances, `SEBD` can sample from the output distribution with variational distance error $Lv2^{-\Theta(r)} < n2^{-\Theta(r)}$. Similarly, by Corollary 3, for at least $1 - 2^{-\Theta(r)}$ fraction of circuit instances, `SEBD` can compute the probability of the all-zeros output string up to additive error $n2^{-\Theta(r)}/2^n$.

Since the runtime of `SEBD` is $O(nD^3)$ when acting on qubits as discussed previously, and D is chosen to be constant for the version of the algorithm used here, the runtime is $O(n)$. □

Appendix A

Background on stochastic gradient and mirror descent

In this section, we review some relevant preliminaries pertaining to convex optimization and stochastic descent algorithms. Much of the material in this section follows the review [Bub15].

A.1 Gradient descent

We first describe the *projected gradient descent* scheme for minimizing a convex differentiable function f on some compact convex subset $\mathcal{X} \subset \mathbb{R}^n$. Starting from some initial point $\mathbf{x}_1 \in \mathcal{X}$, iterate the following procedure:

$$\mathbf{x}_{t+1} = \Pi_{\mathcal{X}}(\mathbf{x}_t - \eta_t \nabla f(\mathbf{x}_t))$$

where $\eta_t > 0$ is the stepsize at iteration t, and $\Pi_{\mathcal{X}}$ is the Euclidean projection onto $\mathcal{X}$, $\Pi_{\mathcal{X}}(\mathbf{x}) = \operatorname{argmin}_{\mathbf{y}\in\mathcal{X}} \|\mathbf{x} - \mathbf{y}\|_2$. The intuition for this strategy is clear: the vector $-\nabla f(\mathbf{x}_t)$ points in the direction of steepest decrease of f at $\mathbf{x}_t$, and in each iteration we take a step of size η_t in this direction and then project back into $\mathcal{X}$. It is sometimes helpful to think of gradient descent in an alternative, *proximal* picture. Namely, Eq. A.1 is equivalent to

$$\mathbf{x}_{t+1} = \operatorname*{argmin}_{\mathbf{x}\in\mathcal{X}} \left[f(\mathbf{x}_t) + \nabla f(\mathbf{x}_t)^\top (\mathbf{x} - \mathbf{x}_t) + \frac{1}{2\eta_t} \|\mathbf{x} - \mathbf{x}_t\|_2^2 \right].$$

Intuitively, the point $\mathbf{x}_{t+1}$ is chosen to minimize $f(\mathbf{x}_t) + \nabla f(\mathbf{x}_t)^\top (\mathbf{x} - \mathbf{x}_t)$, a linearization of f around $\mathbf{x}_t$, while not making the regularization term $\frac{1}{2\eta_t}\|\mathbf{x} - \mathbf{x}_t\|_2^2$ too big.

The following result about projected gradient descent is well-known. Recall that a differentiable function f is L-Lipschitz with respect to $\|\cdot\|$ if $\|\nabla f(\mathbf{x})\|_* \leq L$ for all $\mathbf{x} \in \mathcal{X}$, where $\|\cdot\|_*$ denotes the dual norm.

Theorem 13. *If the convex function f is L-Lipschitz w.r.t. the Euclidean norm, and $\mathcal{X}$ is contained in a Euclidean ball of radius R_2, then projected gradient descent with*

stepsize $\eta = \frac{R_2}{L\sqrt{T}}$ *satisfies*

$$f\left(\frac{1}{T}\sum_{s=1}^{T}\mathbf{x}_s\right) - f(\mathbf{x}^*) \leq \frac{R_2 L}{\sqrt{T}}$$

where $\mathbf{x}^*$ *is a minimizer of* f *on* $\mathcal{X}$.

Note that this implies that $\frac{R_2^2 L^2}{\epsilon^2}$ iterations are sufficient for some desired precision ϵ. We now define strong convexity.

Definition 17 (Strong convexity)**.** *The function* $f : \mathcal{X} \to \mathbb{R}$ *is* λ-strongly convex *with respect to arbitrary norm* $\|\cdot\|$ *for* $\lambda > 0$ *if for all* $\mathbf{x}, \mathbf{y} \in \mathcal{X}$, $f(\mathbf{y}) \geq f(\mathbf{x}) + \nabla f(\mathbf{x})^T(\mathbf{y} - \mathbf{x}) + \frac{\lambda}{2}\|\mathbf{x} - \mathbf{y}\|^2$.

Note that if f is twice differentiable, then f is λ-strongly convex with respect to $\|\cdot\|_2$ if and only if the eigenvalues of the Hessians of f are all at least λ. Recall that f is convex if and only if the Hessians of f are all positive semidefinite. In general, f is λ-strongly convex w.r.t. arbitrary norm $\|\cdot\|$ if and only if $\forall \mathbf{x} \in \mathcal{X}, \mathbf{h} \in \mathbb{R}^p$, it holds that $\mathbf{h}^\top \nabla^2 f(\mathbf{x})\mathbf{h} \geq \lambda \|\mathbf{h}\|^2$.

The following result about projected gradient descent for strongly convex functions is known.

Theorem 14. *Let* f *be* λ_2*-strongly convex and* L*-Lipschitz on* $\mathcal{X}$*, w.r.t. the Euclidean norm. Then projected gradient descent with* $\eta_s = \frac{2}{\lambda_2(s+1)}$ *satisfies*

$$f\left(\sum_{s=1}^{T}\frac{2s}{T(T+1)}\mathbf{x}_s\right) - f(\mathbf{x}^*) \leq \frac{2L^2}{\lambda_2(T+1)}.$$

Note that this result implies that $O\left(\frac{L^2}{\lambda_2 \epsilon}\right)$ iterations are sufficient to optimize f to error ϵ.

It turns out that if one does gradient descent with noisy, unbiased estimates of the gradient $\hat{\mathbf{g}}(\mathbf{x})$ instead of the true gradient $\nabla f(\mathbf{x})$, the above results are qualitatively unchanged. We refer to projected gradient descent with stochastic gradient estimates as *stochastic gradient descent* (SGD). We now state some results formally.

Theorem 15. *Let* f *be convex on* $\mathcal{X}$*, which is contained in a Euclidean ball of radius* R_2*. Assume we have access to a stochastic gradient oracle, which upon input of* $\mathbf{x} \in \mathcal{X}$*, returns a random vector* $\hat{\mathbf{g}}(\mathbf{x})$ *such that* $\mathbb{E}\hat{\mathbf{g}}(\mathbf{x}) = \nabla f(\mathbf{x})$ *and* $\mathbb{E}\|\hat{\mathbf{g}}(\mathbf{x})\|_2^2 \leq G_2^2$*. Then SGD with* $\eta = \frac{R_2}{G_2\sqrt{T}}$ *satisfies*

$$f\left(\frac{1}{T}\sum_{s=1}^{T}\mathbf{x}_s\right) - f(\mathbf{x}^*) \leq \frac{R_2 G_2}{\sqrt{T}}.$$

Theorem 16. *Let* f *be* λ_2*-strongly convex on* $\mathcal{X}$ *w.r.t. the Euclidean norm. Assume we have access to a stochastic gradient oracle, which upon input of* $\mathbf{x} \in \mathcal{X}$*, returns a*

random vector $\hat{\mathbf{g}}(\mathbf{x})$ *such that* $\mathbb{E}\hat{\mathbf{g}}(\mathbf{x}) = \nabla f(\mathbf{x})$ *and* $\mathbb{E}\|\hat{\mathbf{g}}(\mathbf{x})\|_2^2 \leq G_2^2$. *Then SGD with step sizes* $\eta_s = \frac{2}{\lambda_2(s+1)}$ *satisfies*

$$f\left(\sum_{s=1}^{T} \frac{2s}{T(T+1)}\mathbf{x}_s\right) - f(\mathbf{x}^*) \leq \frac{2G_2^2}{\lambda_2(T+1)}.$$

A.2 Mirror descent

A reflection on gradient descent shows that the gradient descent procedure defined above in fact only makes sense when we are working in Euclidean space. For example, an iteration of gradient descent (Eq. A.1) involves adding the vectors $\mathbf{x}_t$ and $\eta_t \nabla f(\mathbf{x}_t)$. When the problem is defined in Euclidean space, $\nabla f(\mathbf{x}_t)$ may be considered as living in the same space by the Riesz representation theorem. However, if (for example) the objective function f is defined on an l_1 space, then the gradient $\nabla f(\mathbf{x})$ lives in the dual l_∞ space, and hence adding these vectors is not even formally well-defined.

Mirror descent may be viewed as a generalization of gradient descent to non-Euclidean geometries. To gain intuition for why we might want to do this, recall that minimizing a function that is L-Lipschitz in the Euclidean norm to precision ϵ requires $O\left(\frac{L^2}{\epsilon^2}\right)$ iterations using the above projected gradient descent bound. Note that this expression does not have any explicit dependence on the dimension p. However, if the parameter L has a dependence on p, then the convergence rate could depend on p implicitly. Consider for example a situation in which we know that all partial derivatives of f are bounded by 1, so that $\|\nabla f(\mathbf{x})\|_\infty \leq 1$ for all $\mathbf{x}$ in the domain. Then it follows that we can bound $L \leq \sqrt{p}$, and so we obtain an upper bound of $O\left(\frac{p}{\epsilon^2}\right)$ for gradient descent, which has a linear dependence on p. But notice that under this assumption, we have a much stronger bound on the ∞-norm of the gradient. In particular, $\|\nabla f\|_\infty \leq 1$, so the Lipschitz constant is only 1 with respect to this geometry. If we could somehow work in an l_1 geometry so that $\|\nabla f(\mathbf{x})\|_\infty$ is the relevant quantity instead of $\|\nabla f(\mathbf{x})\|_2$, then perhaps we could achieve a stronger upper bound on the convergence rate. Indeed, this is possible with mirror descent.

In the remainder of this section, we review basic aspects of mirror descent and its stochastic variant. We begin by fixing an arbitrary norm $\|\cdot\|$ on $\mathbb{R}^p$, and a compact convex set $\mathcal{X} \subset \mathbb{R}^p$. Recall that the dual norm is defined by $\|\mathbf{g}\|_* = \sup_{\mathbf{x}\in\mathbb{R}^p:\|\mathbf{x}\|\leq 1} \mathbf{g}^\top \mathbf{x}$. Let $\mathcal{D} \subset \mathbb{R}^p$ be a convex open set such that $\mathcal{X} \subset \overline{\mathcal{D}}$, where $\overline{\mathcal{D}}$ is the closure of $\mathcal{D}$. Let Φ be a real-valued function on $\mathcal{D}$. Call Φ a *mirror map* (sometimes also called a potential function or distance-generating function) if it satisfies the following technical properties: it is differentiable, strictly convex, $\nabla\Phi(\mathcal{D}) = \mathbb{R}^p$, and $\lim_{\mathbf{x}\to\partial\mathcal{D}} \|\nabla\Phi(\mathbf{x})\| = \infty$. Intuitively, the mirror map Φ may be thought of as a distance-generating function appropriate to the geometry of the problem. For reference, an appropriate choice of Φ for a Euclidean geometry is $\Phi(\mathbf{x}) = \frac{1}{2}\|\mathbf{x}\|_2^2$, with $\mathcal{D}$ chosen to be $\mathbb{R}^p$. For an l_1 geometry where $\mathcal{X}$ is the unit simplex (so points may be interpreted as probability vectors), an appropriate choice of Φ is the negative entropy, $\Phi(\mathbf{x}) = \sum_{i=1}^p \mathbf{x}_i \log \mathbf{x}_i$, defined on the positive orthant $\mathcal{D} = \mathbb{R}^p_{++}$.

We may associate to the mirror map Φ its *Bregman divergence*,

$$D_\Phi(\mathbf{x}, \mathbf{y}) = \Phi(\mathbf{x}) - \big[\Phi(\mathbf{y}) + \nabla\Phi(\mathbf{y})^\top(\mathbf{x} - \mathbf{y})\big].$$

The quantity $D_\Phi(\mathbf{x}, \mathbf{y})$ may be thought of as a distance measure between $\mathbf{x}$ and $\mathbf{y}$, generated by Φ. If $\Phi(\mathbf{x}) = \frac{1}{2}\|\mathbf{x}\|_2^2$, then $D_\Phi(\mathbf{x}, \mathbf{y}) = \frac{1}{2}\|\mathbf{x} - \mathbf{y}\|_2^2$. If $\Phi(\mathbf{x}) = \sum_{i=1}^p \mathbf{x}_i \log \mathbf{x}_i$, then $D_\Phi(\mathbf{x}, \mathbf{y}) = D_{\mathrm{KL}}(\mathbf{x}, \mathbf{y})$, the (generalized) KL divergence between $\mathbf{x}$ and $\mathbf{y}$. We now define the notion of a projection onto the feasible set $\mathcal{X}$ with respect to the Bregman divergence D_Φ:

$$\Pi_{\mathcal{X}}^\Phi(\mathbf{y}) = \operatorname*{argmin}_{\mathbf{x} \in \mathcal{X} \cap \mathcal{D}} D_\Phi(\mathbf{x}, \mathbf{y}).$$

We are now ready to define the mirror descent procedure, with stepsize η. Let $\mathbf{x}_1 = \operatorname{argmin}_{\mathbf{x} \in \mathcal{X} \cap \mathcal{D}} \Phi(\mathbf{x})$. Then mirror descent is defined by the following iteration. For $t \geq 1$, let $\mathbf{y}_{t+1} \in \mathcal{D}$ and $\mathbf{x}_{t+1} \in \mathcal{X}$ be such that

$$\nabla\Phi(\mathbf{y}_{t+1}) = \nabla\Phi(\mathbf{x}_t) - \eta\nabla f(\mathbf{x}_t)$$

and

$$\mathbf{x}_{t+1} \in \Pi_{\mathcal{X}}^\Phi(\mathbf{y}_{t+1}).$$

In other words, we first move to a "dual space" via the mirror map Φ, then do the gradient descent step in the dual space, then move back to the original space again via the mirror map. The resulting point, $\mathbf{y}_{t+1}$, may lie outside the feasible set $\mathcal{X}$, so we then project back to $\mathcal{X}$ via the Bregman divergence generated by Φ. We also note that a step of mirror descent can be equivalently described in the following proximal picture, which makes the relation to gradient descent clearer.

$$\mathbf{x}_{t+1} = \operatorname*{argmin}_{\mathbf{x} \in \mathcal{X} \cap \mathcal{D}} \left[f(\mathbf{x}_t) + \nabla f(\mathbf{x}_t)^\top(\mathbf{x} - \mathbf{x}_t) + \frac{1}{\eta} D_\Phi(\mathbf{x}, \mathbf{x}_t) \right].$$

The following convergence rate can be proven for mirror descent.

Theorem 17. *If Φ is ρ-strongly convex on $\mathcal{X} \cap \mathcal{D}$ with respect to $\|\cdot\|$, $R^2 := \sup_{\mathbf{x} \in \mathcal{X} \cap \mathcal{D}} [\Phi(\mathbf{x}) - \Phi(\mathbf{x}_1)]$, f is convex, and f is L-Lipschitz with respect to $\|\cdot\|$, then mirror descent with $\eta = \frac{R}{L}\sqrt{\frac{2}{T}}$ satisfies*

$$f\left(\frac{1}{T}\sum_{s=1}^T \mathbf{x}_s\right) - f(\mathbf{x}^*) \leq RL\sqrt{\frac{2}{\rho T}}.$$

We now record a mirror map Φ that is appropriate for an l_1 setup, extracted from [Nem+09]. Namely, assuming $\mathcal{X} \subset \mathbb{R}^p$, take

$$\Phi(\mathbf{x}) = (e \ln p) \sum_{i=1}^p |\mathbf{x}_i|^{1 + \frac{1}{\ln p}}, \ p \geq 3.$$

Whenever $\mathcal{X}$ is contained in an l_1-ball of radius 1 centered at the origin, we have $R^2 = e \ln p$ and $\rho \geq 1$. These assumptions on $\mathcal{X}$ can always be achieved by shifting and scaling $\mathcal{X}$. We now state a mirror descent bound for an l_1 geometry.

Theorem 18. *If the convex function f is L-Lipschitz with respect to the norm $\|\cdot\|_1$, and $\mathcal{X}$ is contained in a 1-ball of radius R_1, then projected mirror descent with an appropriate choice of mirror map and stepsizes satisfies*

$$f\left(\frac{1}{T}\sum_{s=1}^{T}\mathbf{x}_s\right) - f(\mathbf{x}^*) \leq R_1 L\sqrt{\frac{2e\ln p}{T}}$$

Finally, we comment on stochastic mirror descent (specializing to the l_1 setup case). In the stochastic setting, one is given access to a oracle which, upon input $\mathbf{x} \in \mathcal{X}$, outputs a random variable $\hat{\mathbf{g}}(\mathbf{x})$ such that $\mathbb{E}\,\hat{\mathbf{g}}(\mathbf{x}) = \nabla f(\mathbf{x})$ and $\mathbb{E}\,\|\hat{\mathbf{g}}(\mathbf{x})\|_\infty^2 \leq G_\infty^2$. The iteration for stochastic mirror descent is identical to that of mirror descent, except the gradients are replaced by their stochastic estimates, just as for the case of stochastic gradient descent. As for the Euclidean case of SGD, the upper bound obtained for SMD is qualitatively very similar to that of the noiseless version.

Theorem 19. *Assume the convex function f is contained on a 1-ball of radius R_1. Assume we have access to a stochastic gradient oracle, which upon input of $\mathbf{x} \in \mathcal{X}$, returns a random vector $\hat{\mathbf{g}}(\mathbf{x})$ such that $\mathbb{E}\hat{\mathbf{g}}(\mathbf{x}) = \nabla f(\mathbf{x})$ and $\mathbb{E}\|\hat{\mathbf{g}}(\mathbf{x})\|_\infty^2 \leq G_\infty^2$. Then SMD with appropriate stepsize satisfies*

$$f\left(\frac{1}{T}\sum_{s=1}^{T}\mathbf{x}_s\right) - f(\mathbf{x}^*) \leq R_1 G_\infty\sqrt{\frac{2e\ln p}{T}}$$

Finally, if f is strongly convex with respect to the norm $\|\cdot\|_1$, then SMD can be accelerated similarly to how SGD can be accelerated for strongly convex functions. See for example [HK14].

Theorem 20. *Let f be λ_1-strongly convex on $\mathcal{X}$ with respect to norm $\|\cdot\|_1$. Assume we have access to a stochastic gradient oracle, which upon input of $\mathbf{x} \in \mathcal{X}$, returns a random vector $\hat{\mathbf{g}}(\mathbf{x})$ such that $\mathbb{E}\hat{\mathbf{g}}(\mathbf{x}) = \nabla f(\mathbf{x})$ and $\mathbb{E}\|\hat{\mathbf{g}}(\mathbf{x})\|_\infty^2 \leq G_\infty^2$. Then a SMD-type algorithm outputs a vector $\bar{\mathbf{x}}$ for which*

$$\mathbb{E}\, f(\bar{\mathbf{x}}) - f(\mathbf{x}^*) \leq \frac{16G_\infty^2}{\lambda_1 T}.$$

Bibliography

[AA11] Scott Aaronson and Alex Arkhipov. "The Computational Complexity of Linear Optics". In: *Proceedings of the Forty-Third Annual ACM Symposium on Theory of Computing*. STOC '11. San Jose, California, USA, 2011, pp. 333–342. ISBN: 9781450306911. DOI: 10.1145/1993636.1993682. arXiv: 1011.3245. URL: https://doi.org/10.1145/1993636.1993682.

[Aar05] Scott Aaronson. "Quantum computing, postselection, and probabilistic polynomial-time". In: *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences* 461.2063 (2005), pp. 3473–3482. DOI: 10.1098/rspa.2005.1546. arXiv: quant-ph/0412187.

[Aar19] Scott Aaronson. *Scott's Supreme Quantum Supremacy FAQ! — Shtetl-Optimized: The Blog of Scott Aaronson*. [Online; accessed 1-April-2021]. Sept. 2019. URL: https://www.scottaaronson.com/blog/?p=4317.

[AB20] Kartiek Agarwal and Ning Bao. "Toy model for decoherence in the black hole information problem". In: *Phys. Rev. D* 102 (8 Oct. 2020), p. 086017. DOI: 10.1103/PhysRevD.102.086017. arXiv: 1912.09491. URL: https://link.aps.org/doi/10.1103/PhysRevD.102.086017.

[AB96] Dorit Aharonov and Michael Ben-Or. "Polynomial simulations of decohered quantum computers". In: *Proceedings of 37th Conference on Foundations of Computer Science*. IEEE. 1996, pp. 46–55. DOI: 10.1109/SFCS.1996.548463. arXiv: quant-ph/9611029.

[AC17] Scott Aaronson and Lijie Chen. "Complexity-theoretic Foundations of Quantum Supremacy Experiments". In: *Proceedings of the 32Nd Computational Complexity Conference*. CCC '17. Riga, Latvia: Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, 2017, 22:1–22:67. ISBN: 978-3-95977-040-8. DOI: 10.4230/LIPIcs.CCC.2017.22. arXiv: 1612.05903. URL: https://doi.org/10.4230/LIPIcs.CCC.2017.22.

[Aga+09] Alekh Agarwal, Martin J Wainwright, Peter L Bartlett, and Pradeep K Ravikumar. "Information-theoretic lower bounds on the oracle complexity of convex optimization". In: *Advances in Neural Information Processing Systems*. 2009, pp. 1–9.

[Aga+11] Alekh Agarwal, Dean P Foster, Daniel J Hsu, Sham M Kakade, and Alexander Rakhlin. "Stochastic convex optimization with bandit feedback". In: *Advances in Neural Information Processing Systems*. 2011, pp. 1035–1043.

[Ama98] Shun-ichi Amari. "Natural Gradient Works Efficiently in Learning". In: *Neural Computation* 10.2 (1998), pp. 251–276.

[Arr+20] Andrew Arrasmith, Lukasz Cincio, Rolando D Somma, and Patrick J Coles. "Operator sampling for shot-frugal optimization in variational algorithms". In: *arXiv preprint arXiv:2004.06252* (2020).

[Aru+19] Frank Arute, Kunal Arya, Ryan Babbush, Dave Bacon, Joseph C Bardin, Rami Barends, Rupak Biswas, Sergio Boixo, Fernando GSL Brandao, David A Buell, et al. "Quantum supremacy using a programmable superconducting processor". In: *Nature* 574.7779 (2019), pp. 505–510. DOI: 10.1038/s41586-019-1666-5.

[Ayo63] Raymond Ayoub. *Introduction to the analytic theory of numbers*. American Mathematical Society, 1963.

[Bar+09] Sean D. Barrett, Stephen D. Bartlett, Andrew C. Doherty, David Jennings, and Terry Rudolph. "Transitions in the computational power of thermal states for measurement-based quantum computation". In: *Phys. Rev. A* 80 (6 Dec. 2009), p. 062328. DOI: 10.1103/PhysRevA.80.062328. arXiv: 0807.4797. URL: https://link.aps.org/doi/10.1103/PhysRevA.80.062328.

[BC17] Jacob C Bridgeman and Christopher T Chubb. "Hand-waving and interpretive dance: an introductory course on tensor networks". In: *Journal of Physics A: Mathematical and Theoretical* 50.22 (2017), p. 223001. arXiv: 1603.03039.

[BCA20] Yimu Bao, Soonwon Choi, and Ehud Altman. "Theory of the phase transition in random unitary circuits with measurements". In: *Phys. Rev. B* 101 (10 Mar. 2020), p. 104301. DOI: 10.1103/PhysRevB.101.104301. arXiv: 1908.04305. URL: https://link.aps.org/doi/10.1103/PhysRevB.101.104301.

[Bel+15] Alexandre Belloni, Tengyuan Liang, Hariharan Narayanan, and Alexander Rakhlin. "Escaping the local minima via simulated annealing: Optimization of approximately convex functions". In: *Proceedings of the Conference on Learning Theory*. PMLR. 2015, pp. 240–265.

[Ben+19] Marcello Benedetti, Edward Grant, Leonard Wossnig, and Simone Severini. "Adversarial quantum circuit learning for pure state approximation". In: *New Journal of Physics* 21.4 (Apr. 2019), p. 043023. DOI: 10.1088/1367-2630/ab14b5. URL: https://doi.org/10.1088%2F1367-2630%2Fab14b5.

[BFK09] Anne Broadbent, Joseph Fitzsimons, and Elham Kashefi. "Universal blind quantum computation". In: *2009 50th Annual IEEE Symposium on Foundations of Computer Science*. IEEE. 2009, pp. 517–526. DOI: 10.1109/FOCS.2009.36. arXiv: 0807.4154.

[BGM21] Sergey Bravyi, David Gosset, and Ramis Movassagh. "Classical algorithms for quantum mean values". In: *Nature Physics* (2021), pp. 1–5. DOI: 10.1038/s41567-020-01109-8. arXiv: 1909.11485.

[BJ19] Aniruddha Bapat and Stephen Jordan. "Bang-bang control as a design principle for classical and quantum optimization algorithms". In: *Quantum Information & Computation* 19 (2019), pp. 424–446.

[BJS10] Michael J Bremner, Richard Jozsa, and Dan J Shepherd. "Classical simulation of commuting quantum computations implies collapse of the polynomial hierarchy". In: *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences* 467.2126 (2010), pp. 459–472. DOI: 10.1098/rspa.2010.0301. arXiv: 1005.1407.

[BK19] Fernando GSL Brandão and Michael J Kastoryano. "Finite correlation length implies efficient preparation of quantum thermal states". In: *Communications in Mathematical Physics* 365.1 (2019), pp. 1–16. DOI: 10.1007/s00220-018-3150-8. arXiv: 1609.07877.

[BKP19] Bruno Bertini, Pavel Kos, and Tomaž Prosen. "Entanglement Spreading in a Minimal Model of Maximal Many-Body Quantum Chaos". In: *Phys. Rev. X* 9 (2 May 2019), p. 021033. DOI: 10.1103/PhysRevX.9.021033. arXiv: 1812.05090. URL: https://link.aps.org/doi/10.1103/PhysRevX.9.021033.

[BMS16] Michael J. Bremner, Ashley Montanaro, and Dan J. Shepherd. "Average-Case Complexity Versus Approximate Simulation of Commuting Quantum Computations". In: *Phys. Rev. Lett.* 117 (8 Aug. 2016), p. 080501. DOI: 10.1103/PhysRevLett.117.080501. arXiv: 1504.07999.

[BMS17] Michael J. Bremner, Ashley Montanaro, and Dan J. Shepherd. "Achieving quantum supremacy with sparse and noisy commuting quantum computations". In: *Quantum* 1 (Apr. 2017), p. 8. ISSN: 2521-327X. DOI: 10.22331/q-2017-04-25-8. arXiv: 1610.01808. URL: https://doi.org/10.22331/q-2017-04-25-8.

[Boi+18] Sergio Boixo, Sergei V Isakov, Vadim N Smelyanskiy, Ryan Babbush, Nan Ding, Zhang Jiang, Michael J Bremner, John M Martinis, and Hartmut Neven. "Characterizing quantum supremacy in near-term devices". In: *Nature Physics* 14.6 (2018), p. 595. DOI: 10.1038/s41567-018-0124-x. arXiv: 1608.00263.

[Bou+19] Adam Bouland, Bill Fefferman, Chinmay Nirkhe, and Umesh Vazirani. "On the complexity and verification of quantum random circuit sampling". In: *Nature Physics* 15.2 (2019), p. 159. DOI: 10.1038/s41567-018-0318-2. arXiv: 1803.04402.

[Bou+21] Adam Bouland, Bill Fefferman, Zeph Landau, and Yunchao Liu. "Noise and the frontier of quantum supremacy". In: (2021). arXiv: 2102.01738.

[Bra+19] Carlos Bravo-Prieto, Ryan LaRose, Marco Cerezo, Yigit Subasi, Lukasz Cincio, and Patrick J Coles. "Variational quantum linear solver: A hybrid algorithm for linear systems". In: *arXiv preprint arXiv:1909.05820* (2019).

[Bro+08] Daniel E Browne, Matthew B Elliott, Steven T Flammia, Seth T Merkel, Akimasa Miyake, and Anthony J Short. "Phase transition of computational power in the resource states for one-way quantum computation". In: *New Journal of Physics* 10.2 (2008), p. 023010. DOI: 10.1088/1367-2630/10/2/023010. arXiv: 0709.1729.

[Bub15] Sébastien Bubeck. "Convex optimization: Algorithms and complexity". In: *Foundations and Trends in Machine Learning* 8.3-4 (2015), pp. 231–357.

[Buh+06] Harry Buhrman, Richard Cleve, Monique Laurent, Noah Linden, Alexander Schrijver, and Falk Unger. "New limits on fault-tolerant quantum computation". In: *2006 47th Annual IEEE Symposium on Foundations of Computer Science (FOCS'06)*. IEEE. 2006, pp. 411–419. DOI: 10.1109/FOCS.2006.50. arXiv: quant-ph/0604141.

[Cam19] Earl Campbell. "Random Compiler for Fast Hamiltonian Simulation". In: *Phys. Rev. Lett.* 123 (7 Aug. 2019), p. 070503. DOI: 10.1103/PhysRevLett.123.070503. URL: https://link.aps.org/doi/10.1103/PhysRevLett.123.070503.

[Cer+20a] M. Cerezo, Andrew Arrasmith, Ryan Babbush, Simon C. Benjamin, Suguru Endo, Keisuke Fujii, Jarrod R. McClean, Kosuke Mitarai, Xiao Yuan, Lukasz Cincio, and Patrick J. Coles. "Variational Quantum Algorithms". In: *arXiv e-prints* (Dec. 2020). arXiv: 2012.09265 [quant-ph].

[Cer+20b] M Cerezo, Kunal Sharma, Andrew Arrasmith, and Patrick J Coles. "Variational quantum state eigensolver". In: *arXiv preprint arXiv:2004.01372* (2020).

[Cha+19] Amos Chan, Rahul M. Nandkishore, Michael Pretko, and Graeme Smith. "Unitary-projective entanglement dynamics". In: *Phys. Rev. B* 99 (22 June 2019), p. 224307. DOI: 10.1103/PhysRevB.99.224307. arXiv: 1808.05949. URL: https://link.aps.org/doi/10.1103/PhysRevB.99.224307.

[Cho+20] Soonwon Choi, Yimu Bao, Xiao-Liang Qi, and Ehud Altman. "Quantum Error Correction in Scrambling Dynamics and Measurement-Induced Phase Transition". In: *Phys. Rev. Lett.* 125 (3 July 2020), p. 030505. DOI: 10.1103/PhysRevLett.125.030505. arXiv: 1903.05124. URL: https://link.aps.org/doi/10.1103/PhysRevLett.125.030505.

[Col03] Benoît Collins. "Moments and cumulants of polynomial random variables on unitary groups, the Itzykson-Zuber integral, and free probability". In: *International Mathematics Research Notices* 2003.17 (2003), pp. 953–982. DOI: 10.1155/S107379280320917X. arXiv: math-ph/0205010.

[Cro+19] Andrew W. Cross, Lev S. Bishop, Sarah Sheldon, Paul D. Nation, and Jay M. Gambetta. "Validating quantum computers using randomized model circuits". In: *Phys. Rev. A* 100 (3 Sept. 2019), p. 032328. DOI: 10.1103/PhysRevA.100.032328. arXiv: 1811.12926. URL: https://link.aps.org/doi/10.1103/PhysRevA.100.032328.

[CŚ06] Benoît Collins and Piotr Śniady. "Integration with respect to the Haar measure on unitary, orthogonal and symplectic group". In: *Communications in Mathematical Physics* 264.3 (2006), pp. 773–795. DOI: 10.1007/s00220-006-1554-3. arXiv: math-ph/0402073.

[CT91] Thomas M. Cover and Joy A. Thomas. *Elements of Information Theory*. New York, NY, USA: Wiley-Interscience, 1991. ISBN: 0-471-06259-6.

[Des+18] Abhinav Deshpande, Bill Fefferman, Minh C. Tran, Michael Foss-Feig, and Alexey V. Gorshkov. "Dynamical Phase Transitions in Sampling Complexity". In: *Phys. Rev. Lett.* 121 (3 July 2018), p. 030501. DOI: 10.1103/PhysRevLett.121.030501. arXiv: 1703.05332. URL: https://link.aps.org/doi/10.1103/PhysRevLett.121.030501.

[DHB20] Alexander M Dalzell, Nicholas Hunter-Jones, and Fernando GSL Brandão. "Random quantum circuits anti-concentrate in log depth". In: (2020). arXiv: 2011.12277.

[DOP07] Oscar CO Dahlsten, Roberto Oliveira, and Martin B Plenio. "The emergence of typical entanglement in two-party random processes". In: *Journal of Physics A: Mathematical and Theoretical* 40.28 (2007), p. 8081. DOI: 10.1088/1751-8113/40/28/s16. arXiv: quant-ph/0701125.

[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)*. 2018, pp. 23–34. DOI: 10.1109/FOCS.2018.00012. arXiv: 1711.09457.

[FA20] Yohei Fuji and Yuto Ashida. "Measurement-induced quantum criticality under continuous monitoring". In: *Phys. Rev. B* 102 (5 Aug. 2020), p. 054302. DOI: 10.1103/PhysRevB.102.054302. arXiv: 2004.11957. URL: https://link.aps.org/doi/10.1103/PhysRevB.102.054302.

[Fan+20] Ruihua Fan, Sagar Vijay, Ashvin Vishwanath, and Yi-Zhuang You. *Self-Organized Error Correction in Random Unitary Circuits with Measurement*. 2020. arXiv: 2002.12385 [cond-mat.stat-mech].

[Fey82] Richard P. Feynman. "Simulating physics with computers". In: *International Journal of Theoretical Physics* 21.6 (1982), pp. 467–488. DOI: 10.1007/BF02650179. URL: https://doi.org/10.1007/BF02650179.

[FGG14] Edward Farhi, Jeffrey Goldstone, and Sam Gutmann. "A quantum approximate optimization algorithm". In: *arXiv:1411.4028* (2014).

[FHH21] Lukasz Fidkowski, Jeongwan Haah, and Matthew B. Hastings. "How Dynamical Quantum Memories Forget". In: *Quantum* 5 (Jan. 2021), p. 382. ISSN: 2521-327X. DOI: 10.22331/q-2021-01-17-382. arXiv: 2008.10611. URL: https://doi.org/10.22331/q-2021-01-17-382.

[FKM05] Abraham D Flaxman, Adam Tauman Kalai, and H Brendan McMahan. "Online convex optimization in the bandit setting: gradient descent without a gradient". In: *Proceedings of the sixteenth annual ACM-SIAM symposium on Discrete algorithms*. Society for Industrial and Applied Mathematics. 2005, pp. 385–394.

[FN18] Edward Farhi and Hartmut Neven. "Classification with quantum neural networks on near term processors". In: *arXiv:1802.06002* (2018).

[GAW19] András Gilyén, Srinivasan Arunachalam, and Nathan Wiebe. "Optimizing quantum optimization algorithms via faster quantum gradient computation". In: *Proceedings of the Thirtieth Annual ACM-SIAM Symposium on Discrete Algorithms* (2019), pp. 1425–1444.

[GD18] Xun Gao and Luming Duan. *Efficient classical simulation of noisy quantum computation*. 2018. arXiv: 1810.03176.

[GH20a] Michael J. Gullans and David A. Huse. "Dynamical Purification Phase Transition Induced by Quantum Measurements". In: *Phys. Rev. X* 10 (4 Oct. 2020), p. 041020. DOI: 10.1103/PhysRevX.10.041020. arXiv: 1905.05195. URL: https://link.aps.org/doi/10.1103/PhysRevX.10.041020.

[GH20b] Michael J. Gullans and David A. Huse. "Scalable Probes of Measurement-Induced Criticality". In: *Phys. Rev. Lett.* 125 (7 Aug. 2020), p. 070606. DOI: 10.1103/PhysRevLett.125.070606. arXiv: 1910.00020. URL: https://link.aps.org/doi/10.1103/PhysRevLett.125.070606.

[Got98] Daniel Gottesman. "The Heisenberg representation of quantum computers". In: *Proc. XXII International Colloquium on Group Theoretical Methods in Physics, 1998*. 1998, pp. 32–43. arXiv: quant-ph/9807006.

[GS17] Gian Giacomo Guerreschi and Mikhail Smelyanskiy. *Practical optimization for hybrid quantum-classical algorithms*. 2017. eprint: arXiv:1701.01450.

[Gu13] Yinzheng Gu. "Moments of random matrices and weingarten functions". PhD thesis. 2013.

[Gun11] Adityanand Guntuboyina. "Lower bounds for the minimax risk using f-divergences, and applications". In: *IEEE Transactions on Information Theory* 57.4 (2011), pp. 2386–2399.

[GWD17] Xun Gao, Sheng-Tao Wang, and L-M Duan. "Quantum supremacy for simulating a translation-invariant Ising spin model". In: *Physical Review Letters* 118.4 (2017), p. 040502. arXiv: 1607.04947.

[Has17] Matthew B Hastings. "The asymptotics of quantum max-flow min-cut". In: *Communications in Mathematical Physics* 351.1 (2017), pp. 387–418. DOI: 10.1007/s00220-016-2791-8. arXiv: 1603.03717.

[Hav+19] Vojtěch Havlíček, Antonio D Córcoles, Kristan Temme, Aram W Harrow, Abhinav Kandala, Jerry M Chow, and Jay M Gambetta. "Supervised learning with quantum-enhanced feature spaces". In: *Nature* 567.7747 (2019), pp. 209–212.

[Hay+16] Patrick Hayden, Sepehr Nezami, Xiao-Liang Qi, Nathaniel Thomas, Michael Walter, and Zhao Yang. "Holographic duality from random tensor networks". In: *Journal of High Energy Physics* 2016.11 (2016), p. 9. arXiv: 1601.01694.

[HK14] Elad Hazan and Satyen Kale. "Beyond the regret minimization barrier: optimal algorithms for stochastic strongly-convex optimization". In: *The Journal of Machine Learning Research* 15.1 (2014), pp. 2489–2512.

[HLW06] Patrick Hayden, Debbie W Leung, and Andreas Winter. "Aspects of generic entanglement". In: *Communications in Mathematical Physics* 265.1 (2006), pp. 95–117. DOI: 10.1007/s00220-006-1535-6. arXiv: quant-ph/0407049.

[HM17] Aram W Harrow and Ashley Montanaro. "Quantum computational supremacy". In: *Nature* 549.7671 (2017), p. 203. DOI: 10.1038/nature23458. arXiv: 1809.07442.

[HM18] Aram W. Harrow and Saeed Mehraban. *Approximate unitary t-designs by short random quantum circuits using nearest-neighbor and long-range gates*. 2018. arXiv: 1809.06957.

[HN03] Aram W. Harrow and Michael A. Nielsen. "Robustness of quantum gates in the presence of noise". In: *Phys. Rev. A* 68 (1 July 2003), p. 012308. DOI: 10.1103/PhysRevA.68.012308. arXiv: quant-ph/0301108. URL: https://link.aps.org/doi/10.1103/PhysRevA.68.012308.

[HN21] Aram W. Harrow and John C. Napp. "Low-Depth Gradient Measurements Can Improve Convergence in Variational Hybrid Quantum-Classical Algorithms". In: *Phys. Rev. Lett.* 126 (14 Apr. 2021), p. 140502. DOI: 10.1103/PhysRevLett.126.140502. arXiv: 1901.05374 [quant-ph]. URL: https://link.aps.org/doi/10.1103/PhysRevLett.126.140502.

[Hou50] Raymond Marie Ferdinand Houtappel. "Order-disorder in hexagonal lattices". In: *Physica* 16.5 (1950), pp. 425–455. DOI: 10.1016/0031-8914(50)90130-3.

[HOW07] Michał Horodecki, Jonathan Oppenheim, and Andreas Winter. "Quantum State Merging and Negative Information". In: *Communications in Mathematical Physics* 269.1 (Jan. 2007), pp. 107–136. ISSN: 1432-0916. DOI: 10.1007/s00220-006-0118-x. arXiv: quant-ph/0512247.

[Hua19] Yichen Huang. *Dynamics of Renyi entanglement entropy in local quantum circuits with charge conservation*. 2019. arXiv: 1902.00977.

[Hun19] Nicholas Hunter-Jones. *Unitary designs from statistical mechanics in random quantum circuits*. 2019. arXiv: 1905.12053.

[IK21] Matteo Ippoliti and Vedika Khemani. "Postselection-Free Entanglement Dynamics via Spacetime Duality". In: *Phys. Rev. Lett.* 126 (6 Feb. 2021), p. 060501. DOI: 10.1103/PhysRevLett.126.060501. arXiv: 2010.15840. URL: https://link.aps.org/doi/10.1103/PhysRevLett.126.060501.

[Ipp+21] Matteo Ippoliti, Michael J. Gullans, Sarang Gopalakrishnan, David A. Huse, and Vedika Khemani. "Entanglement Phase Transitions in Measurement-Only Dynamics". In: *Phys. Rev. X* 11 (1 Feb. 2021), p. 011030. DOI: 10.1103/PhysRevX.11.011030. arXiv: 2004.09560. URL: https://link.aps.org/doi/10.1103/PhysRevX.11.011030.

[Jia+20] Chao-Ming Jian, Yi-Zhuang You, Romain Vasseur, and Andreas W. W. Ludwig. "Measurement-induced criticality in random quantum circuits". In: *Phys. Rev. B* 101 (10 Mar. 2020), p. 104302. DOI: 10.1103/PhysRevB.101.104302. arXiv: 1908.08051. URL: https://link.aps.org/doi/10.1103/PhysRevB.101.104302.

[JN11] Anatoli Juditsky and Arkadi Nemirovski. "First order methods for non-smooth convex large-scale optimization". In: *Optimization for Machine Learning* (2011), pp. 121–148.

[JNR12] Kevin G Jamieson, Robert Nowak, and Ben Recht. "Query complexity of derivative-free optimization". In: *Advances in Neural Information Processing Systems*. 2012, pp. 2672–2680.

[Jor05] Stephen P Jordan. "Fast quantum algorithm for numerical gradient estimation". In: *Physical Review Letters* 95.5 (2005), p. 050501.

[Kal11] Gil Kalai. *How quantum computers fail: quantum codes, correlations in physical systems, and noise accumulation*. 2011. arXiv: 1106.0485 [quant-ph].

[Kan+17] Abhinav Kandala, Antonio Mezzacapo, Kristan Temme, Maika Takita, Markus Brink, Jerry M Chow, and Jay M Gambetta. "Hardware-efficient variational quantum eigensolver for small molecules and quantum magnets". In: *Nature* 549.7671 (2017), p. 242.

[KB19] Bálint Koczor and Simon C. Benjamin. *Quantum natural gradient generalised to non-unitary circuits*. 2019. eprint: arXiv:1912.08660.

[Kem+08] Julia Kempe, Oded Regev, Falk Unger, and Ronald De Wolf. "Upper bounds on the noise threshold for fault-tolerant quantum computing". In: *International Colloquium on Automata, Languages, and Programming*. Springer. 2008, pp. 845–856. DOI: 10.1007/978-3-540-70575-8_69. arXiv: 0802.1464.

[Key+18] C. W. von Keyserlingk, Tibor Rakovszky, Frank Pollmann, and S. L. Sondhi. "Operator Hydrodynamics, OTOCs, and Entanglement Growth in Systems without Conservation Laws". In: *Phys. Rev. X* 8 (2 Apr. 2018), p. 021013. DOI: 10.1103/PhysRevX.8.021013. arXiv: 1705.08910. URL: https://link.aps.org/doi/10.1103/PhysRevX.8.021013.

[Kha+19] Sumeet Khatri, Ryan LaRose, Alexander Poremba, Lukasz Cincio, Andrew T. Sornborger, and Patrick J. Coles. "Quantum-assisted quantum compiling". In: *Quantum* 3 (May 2019), p. 140. ISSN: 2521-327X. DOI: 10.22331/q-2019-05-13-140. URL: https://doi.org/10.22331/q-2019-05-13-140.

[Kim17a] Isaac H. Kim. *Holographic quantum simulation*. 2017. arXiv: 1702.02093 [quant-ph].

[Kim17b] Isaac H. Kim. *Noise-resilient preparation of quantum many-body ground states*. 2017. arXiv: 1703.00032.

[KK14] Gil Kalai and Guy Kindler. *Gaussian Noise Sensitivity and BosonSampling*. 2014. arXiv: 1409.3093.

[KMM21] Yasuhiro Kondo, Ryuhei Mori, and Ramis Movassagh. "Fine-grained analysis and improved robustness of quantum supremacy for random circuit sampling". In: (2021). arXiv: 2102.01960.

[Küb+20] Jonas M. Kübler, Andrew Arrasmith, Lukasz Cincio, and Patrick J. Coles. "An Adaptive Optimizer for Measurement-Frugal Variational Algorithms". In: *Quantum* 4 (May 2020), p. 263. ISSN: 2521-327X. DOI: 10.22331/q-2020-05-11-263. URL: https://doi.org/10.22331/q-2020-05-11-263.

[LAB21] Ali Lavasani, Yahya Alavirad, and Maissam Barkeshli. "Measurement-induced topological entanglement transitions in symmetric random quantum circuits". In: *Nature Physics* (2021), pp. 1–6. DOI: 10.1038/s41567-020-01112-z. arXiv: 2004.07243.

[LB17] Ying Li and Simon C Benjamin. "Efficient variational quantum simulator incorporating active error minimization". In: *Physical Review X* 7.2 (2017), p. 021050.

[LCF18] Yaodong Li, Xiao Chen, and Matthew P. A. Fisher. "Quantum Zeno effect and the many-body entanglement transition". In: *Phys. Rev. B* 98 (20 Nov. 2018), p. 205136. DOI: 10.1103/PhysRevB.98.205136. arXiv: 1808.06134. URL: https://link.aps.org/doi/10.1103/PhysRevB.98.205136.

[LCF19] Yaodong Li, Xiao Chen, and Matthew P. A. Fisher. "Measurement-driven entanglement transition in hybrid quantum circuits". In: *Phys. Rev. B* 100 (13 Oct. 2019), p. 134306. DOI: 10.1103/PhysRevB.100.134306. arXiv: 1901.08092. URL: https://link.aps.org/doi/10.1103/PhysRevB.100.134306.

[LDD19] Sirui Lu, Lu-Ming Duan, and Dong-Ling Deng. "Quantum Adversarial Machine Learning". In: *arXiv preprint arXiv:2001.00030* (2019).

[Lev03] L. A. Levin. "The Tale of One-Way Functions". In: *Problems of Information Transmission* 39.1 (2003), pp. 92–103. DOI: 10.1023/A:1023634616182. URL: https://doi.org/10.1023/A:1023634616182.

[LF20] Yaodong Li and Matthew P. A. Fisher. *Statistical Mechanics of Quantum Error-Correcting Codes*. 2020. arXiv: 2007.03822 [quant-ph].

[Li+17] Jun Li, Xiaodong Yang, Xinhua Peng, and Chang-Pu Sun. "Hybrid Quantum-Classical Approach to Quantum Optimal Control". In: *Phys. Rev. Lett.* 118 (15 Apr. 2017), p. 150503. DOI: 10.1103/PhysRevLett.118.150503. URL: https://link.aps.org/doi/10.1103/PhysRevLett.118.150503.

[Li+20] Yaodong Li, Xiao Chen, Andreas W. W. Ludwig, and Matthew P. A. Fisher. *Conformal invariance and quantum non-locality in hybrid quantum circuits*. 2020. arXiv: 2003.12721 [quant-ph].

[Liu+19] Jin-Guo Liu, Yi-Hong Zhang, Yuan Wan, and Lei Wang. "Variational quantum eigensolver with fewer qubits". In: *Phys. Rev. Research* 1 (2 Sept. 2019), p. 023025. DOI: 10.1103/PhysRevResearch.1.023025. URL: https://link.aps.org/doi/10.1103/PhysRevResearch.1.023025.

[LP20] Oliver Lunt and Arijeet Pal. "Measurement-induced entanglement transitions in many-body localized systems". In: *Phys. Rev. Research* 2 (4 Oct. 2020), p. 043072. DOI: 10.1103/PhysRevResearch.2.043072. arXiv: 2005.13603. URL: https://link.aps.org/doi/10.1103/PhysRevResearch.2.043072.

[LS14] Hong Liu and S. Josephine Suh. "Entanglement Tsunami: Universal Scaling in Holographic Thermalization". In: *Phys. Rev. Lett.* 112 (1 Jan. 2014), p. 011601. DOI: 10.1103/PhysRevLett.112.011601. eprint: 1305.7244.

[LW18] Jin-Guo Liu and Lei Wang. "Differentiable learning of quantum circuit Born machines". In: *Phys. Rev. A* 98 (6 Dec. 2018), p. 062324. DOI: 10.1103/PhysRevA.98.062324. URL: https://link.aps.org/doi/10.1103/PhysRevA.98.062324.

[McA+19] Sam McArdle, Tyson Jones, Suguru Endo, Ying Li, Simon C. Benjamin, and Xiao Yuan. "Variational ansatz-based quantum simulation of imaginary time evolution". In: *npj Quantum Information* 5.1 (2019), p. 75. DOI: 10.1038/s41534-019-0187-2. URL: https://doi.org/10.1038/s41534-019-0187-2.

[McC+16] Jarrod R McClean, Jonathan Romero, Ryan Babbush, and Alán Aspuru-Guzik. "The theory of variational hybrid quantum-classical algorithms". In: *New Journal of Physics* 18.2 (2016), p. 023023.

[McC+18] Jarrod R McClean, Sergio Boixo, Vadim N Smelyanskiy, Ryan Babbush, and Hartmut Neven. "Barren plateaus in quantum neural network training landscapes". In: *Nature Communications* 9.1 (2018), pp. 1–6.

[Mit+18] Kosuke Mitarai, Makoto Negoro, Masahiro Kitagawa, and Keisuke Fujii. "Quantum circuit learning". In: *Physical Review A* 98.3 (2018), p. 032309.

[MMD19] Gopikrishnan Muraleedharan, Akimasa Miyake, and Ivan H Deutsch. "Quantum computational supremacy in the sampling of bosonic random walkers on a one-dimensional lattice". In: *New Journal of Physics* 21.5 (2019), p. 055003. DOI: 10.1088/1367-2630/ab0610. arXiv: 1805.01858 [quant-ph].

[Mov19] Ramis Movassagh. "Quantum supremacy and random circuits". In: (2019). arXiv: 1909.06210.

[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. arXiv: quant-ph/0511069.

[Nah+21] Adam Nahum, Sthitadhi Roy, Brian Skinner, and Jonathan Ruhman. *Measurement and entanglement phase transitions in all-to-all quantum circuits, on quantum trees, and in Landau-Ginsburg theory*. 2021. arXiv: 2009.11311 [cond-mat.stat-mech].

[Nap+19] John C Napp, Rolando L La Placa, Alexander M Dalzell, Fernando GSL Brandao, and Aram W Harrow. *Efficient classical simulation of random shallow 2D quantum circuits*. 2019. arXiv: 2001.00021 [quant-ph]. In submission.

[NC00] Michael A. Nielsen and Isaac L. Chuang. *Quantum Computation and Quantum Information*. Cambridge University Press, 2000. ISBN: 978-0521635035.

[Nei+18] C. Neill, P. Roushan, K. Kechedzhi, S. Boixo, S. V. Isakov, V. Smelyanskiy, A. Megrant, B. Chiaro, A. Dunsworth, K. Arya, R. Barends, B. Burkett, Y. Chen, Z. Chen, A. Fowler, B. Foxen, M. Giustina, R. Graff, E. Jeffrey, T. Huang, J. Kelly, P. Klimov, E. Lucero, J. Mutus, M. Neeley, C. Quintana, D. Sank, A. Vainsencher, J. Wenner, T. C. White, H. Neven, and J. M. Martinis. "A blueprint for demonstrating quantum supremacy with superconducting qubits". In: *Science* 360.6385 (2018), pp. 195–199. ISSN: 0036-8075. DOI: 10.1126/science.aao4309. arXiv: 1709.06678. URL: https://science.sciencemag.org/content/360/6385/195.

[Nem+09] Arkadi Nemirovski, Anatoli Juditsky, Guanghui Lan, and Alexander Shapiro. "Robust stochastic approximation approach to stochastic programming". In: *SIAM Journal on Optimization* 19.4 (2009), pp. 1574–1609.

[NS20] Adam Nahum and Brian Skinner. "Entanglement and dynamics of diffusion-annihilation processes with Majorana defects". In: *Phys. Rev. Research* 2 (2 June 2020), p. 023288. DOI: 10.1103/PhysRevResearch.2.023288. arXiv: 1911.11169. URL: https://link.aps.org/doi/10.1103/PhysRevResearch.2.023288.

[NVH18] Adam Nahum, Sagar Vijay, and Jeongwan Haah. "Operator Spreading in Random Unitary Circuits". In: *Phys. Rev. X* 8 (2 Apr. 2018), p. 021014. DOI: 10.1103/PhysRevX.8.021014. arXiv: 1705.08975. URL: https://link.aps.org/doi/10.1103/PhysRevX.8.021014.

[OB18] Michał Oszmaniec and Daniel J Brod. "Classical simulation of photonic linear optics with lost particles". In: *New Journal of Physics* 20.9 (Sept. 2018), p. 092002. DOI: 10.1088/1367-2630/aadfa8. arXiv: 1801.06166. URL: https://doi.org/10.1088%2F1367-2630%2Faadfa8.

[Orú14] Román Orús. "A practical introduction to tensor networks: Matrix product states and projected entangled pair states". In: *Annals of Physics* 349 (2014), pp. 117–158. DOI: 10.1016/j.aop.2014.06.013. arXiv: 1306.2164.

[Osb06] Tobias J. Osborne. "Efficient Approximation of the Dynamics of One-Dimensional Quantum Spin Systems". In: *Phys. Rev. Lett.* 97 (15 Oct. 2006), p. 157202. DOI: 10.1103/PhysRevLett.97.157202. URL: https://link.aps.org/doi/10.1103/PhysRevLett.97.157202.

[Pag93] Don N. Page. "Average entropy of a subsystem". In: *Phys. Rev. Lett.* 71 (9 Aug. 1993), pp. 1291–1294. DOI: 10.1103/PhysRevLett.71.1291. arXiv: gr-qc/9305007. URL: https://link.aps.org/doi/10.1103/PhysRevLett.71.1291.

[Pan+20] Feng Pan, Pengfei Zhou, Sujie Li, and Pan Zhang. "Contracting Arbitrary Tensor Networks: General Approximate Algorithm and Applications in Graphical Models and Quantum Circuit Simulations". In: *Phys. Rev. Lett.* 125 (6 Aug. 2020), p. 060503. DOI: 10.1103/PhysRevLett.125.060503. arXiv: 1912.03014. URL: https://link.aps.org/doi/10.1103/PhysRevLett.125.060503.

[Pat92] Ramamohan Paturi. "On the degree of polynomials that approximate symmetric Boolean functions (preliminary version)". In: *Proceedings of the twenty-fourth annual ACM symposium on Theory of computing*. ACM. 1992, pp. 468–474. DOI: 10.1145/129712.129758.

[Per+14] Alberto Peruzzo, Jarrod McClean, Peter Shadbolt, Man-Hong Yung, Xiao-Qi Zhou, Peter J Love, Alán Aspuru-Guzik, and Jeremy L O'brien. "A variational eigenvalue solver on a photonic quantum processor". In: *Nature Communications* 5 (2014), p. 4213.

[Pre18] John Preskill. "Quantum Computing in the NISQ era and beyond". In: *Quantum* 2 (2018), p. 79.

[RA19] Jonathan Romero and Alan Aspuru-Guzik. "Variational quantum generators: Generative adversarial quantum machine learning for continuous distributions". In: *arXiv preprint arXiv:1901.00848* (2019).

[Rak07] Evguenii A Rakhmanov. "Bounds for polynomials with a unit discrete norm". In: *Annals of mathematics* (2007), pp. 55–88.

[Raz04] Alexander A Razborov. "An upper bound on the threshold quantum decoherence rate". In: *Quantum Information & Computation* 4.3 (2004), pp. 222–228. DOI: 10.26421/QIC4.3. arXiv: quant-ph/0310136.

[RB01] Robert Raussendorf and Hans J. Briegel. "A One-Way Quantum Computer". In: *Phys. Rev. Lett.* 86 (22 May 2001), pp. 5188–5191. DOI: 10.1103/PhysRevLett.86.5188. URL: https://link.aps.org/doi/10.1103/PhysRevLett.86.5188.

[RBH05] Robert Raussendorf, Sergey Bravyi, and Jim Harrington. "Long-range quantum entanglement in noisy cluster states". In: *Phys. Rev. A* 71 (6 June 2005), p. 062313. DOI: 10.1103/PhysRevA.71.062313. arXiv: quant-ph/0407255. URL: https://link.aps.org/doi/10.1103/PhysRevA.71.062313.

[Rom+18] Jonathan Romero, Ryan Babbush, Jarrod McClean, Cornelius Hempel, Peter Love, and Alán Aspuru-Guzik. "Strategies for quantum computing molecular energies using the unitary coupled cluster ansatz". In: *Quantum Science and Technology* (2018).

[Ros13] David J. Rosenbaum. "Optimal Quantum Circuits for Nearest-Neighbor Architectures". In: *8th Conference on the Theory of Quantum Computation, Communication and Cryptography (TQC 2013)*. Vol. 22. 2013, pp. 294–307. DOI: 10.4230/LIPIcs.TQC.2013.294. arXiv: 1205.0036.

[RR11] Maxim Raginsky and Alexander Rakhlin. "Information-based complexity, feedback and dynamics in convex programming". In: *IEEE Transactions on Information Theory* 57.10 (2011), pp. 7036–7056.

[SB09] Dan Shepherd and Michael J. Bremner. "Temporally unstructured quantum computation". In: *Proceedings of the Royal Society A* 465 (2009), pp. 1413–1439. DOI: 10.1098/rspa.2008.0443. arXiv: 0809.0847.

[Sch+08] Norbert Schuch, Michael M. Wolf, Frank Verstraete, and J. Ignacio Cirac. "Entropy Scaling and Simulability by Matrix Product States". In: *Phys. Rev. Lett.* 100 (3 Jan. 2008), p. 030504. DOI: 10.1103/PhysRevLett.100.030504. arXiv: 0705.0292. URL: https://link.aps.org/doi/10.1103/PhysRevLett.100.030504.

[Sch+19] Maria Schuld, Ville Bergholm, Christian Gogolin, Josh Izaac, and Nathan Killoran. "Evaluating analytic gradients on quantum hardware". In: *Physical Review A* 99.3 (2019), p. 032331.

[Sch+20] Maria Schuld, Alex Bocharov, Krysta M Svore, and Nathan Wiebe. "Circuit-centric quantum classifiers". In: *Physical Review A* 101.3 (2020), p. 032308.

[SH20] Shengqi Sang and Timothy H. Hsieh. *Measurement Protected Quantum Phases*. 2020. arXiv: 2004.09509 [cond-mat.stat-mech].

[Sho96] Peter W Shor. "Fault-tolerant quantum computation". In: *Proceedings of 37th Conference on Foundations of Computer Science*. IEEE. 1996, pp. 56–65. DOI: 10.1109/SFCS.1996.548464. arXiv: quant-ph/9605011.

[Sho97] Peter W. Shor. "Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer". In: *SIAM Journal on Computing* 26.5 (1997), pp. 1484–1509. DOI: 10.1137/S0097539795293172. eprint: https://doi.org/10.1137/S0097539795293172. URL: https://doi.org/10.1137/S0097539795293172.

[SK19] Maria Schuld and Nathan Killoran. "Quantum machine learning in feature Hilbert spaces". In: *Physical Review Letters* 122.4 (2019), p. 040504.

[SK20] Barnaby van Straaten and Bálint Koczor. *Measurement cost of metric-aware variational quantum algorithms*. 2020. eprint: arXiv:2005.05172.

[SRN19] Brian Skinner, Jonathan Ruhman, and Adam Nahum. "Measurement-Induced Phase Transitions in the Dynamics of Entanglement". In: *Phys. Rev. X* 9 (3 July 2019), p. 031009. DOI: 10.1103/PhysRevX.9.031009. arXiv: 1808.05953. URL: https://link.aps.org/doi/10.1103/PhysRevX.9.031009.

[SRS19] M. Szyniszewski, A. Romito, and H. Schomerus. "Entanglement transition from variable-strength weak measurements". In: *Phys. Rev. B* 100 (6 Aug. 2019), p. 064204. DOI: 10.1103/PhysRevB.100.064204. arXiv: 1903.05452. URL: https://link.aps.org/doi/10.1103/PhysRevB.100.064204.

[SRS20] M. Szyniszewski, A. Romito, and H. Schomerus. "Universality of Entanglement Transitions from Stroboscopic to Continuous Measurements". In: *Phys. Rev. Lett.* 125 (21 Nov. 2020), p. 210602. DOI: 10.1103/PhysRevLett.125.210602. arXiv: 2005.01863. URL: https://link.aps.org/doi/10.1103/PhysRevLett.125.210602.

[Ste70] John Stephenson. "Ising-Model Spin Correlations on the Triangular Lattice. IV. Anisotropic Ferromagnetic and Antiferromagnetic Lattices". In: *Journal of Mathematical Physics* 11.2 (1970), pp. 420–431. DOI: 10.1063/1.1665155.

[Sto+20] James Stokes, Josh Izaac, Nathan Killoran, and Giuseppe Carleo. "Quantum Natural Gradient". In: *Quantum* 4 (May 2020), p. 269. ISSN: 2521-327X. DOI: 10.22331/q-2020-05-25-269. URL: https://doi.org/10.22331/q-2020-05-25-269.

[Swe+19] Ryan Sweke, Frederik Wilde, Johannes Meyer, Maria Schuld, Paul K Fährmann, Barthélémy Meynard-Piganeau, and Jens Eisert. "Stochastic gradient descent for hybrid quantum-classical optimization". In: *arXiv preprint arXiv:1910.01155* (2019).

[TD04] Barbara M Terhal and David P DiVincenzo. "Adptive quantum computation, constant depth quantum circuits and arthur-merlin games". In: *Quantum Information & Computation* 4.2 (2004), pp. 134–145. DOI: 10.26421/QIC4.2. arXiv: quant-ph/0205133.

[TFD20] Xhek Turkeshi, Rosario Fazio, and Marcello Dalmonte. "Measurement-induced criticality in $(2 + 1)$-dimensional hybrid quantum circuits". In: *Phys. Rev. B* 102 (1 July 2020), p. 014315. DOI: 10.1103/PhysRevB.102.014315. arXiv: 2007.02970. URL: https://link.aps.org/doi/10.1103/PhysRevB.102.014315.

[TZ20] Qicheng Tang and W. Zhu. "Measurement-induced phase transition: A case study in the nonintegrable model by density-matrix renormalization group calculations". In: *Phys. Rev. Research* 2 (1 Jan. 2020), p. 013022. DOI: 10.1103/PhysRevResearch.2.013022. arXiv: 1908.11253. URL: https://link.aps.org/doi/10.1103/PhysRevResearch.2.013022.

[Vas+19] Romain Vasseur, Andrew C. Potter, Yi-Zhuang You, and Andreas W. W. Ludwig. "Entanglement transitions from holographic random tensor networks". In: *Phys. Rev. B* 100 (13 Oct. 2019), p. 134203. DOI: 10.1103/PhysRevB.100.134203. arXiv: 1807.07082. URL: https://link.aps.org/doi/10.1103/PhysRevB.100.134203.

[VC06] F. Verstraete and J. I. Cirac. "Matrix product states represent ground states faithfully". In: *Phys. Rev. B* 73 (9 Mar. 2006), p. 094423. DOI: 10.1103/PhysRevB.73.094423. arXiv: cond-mat/0505140. URL: https://link.aps.org/doi/10.1103/PhysRevB.73.094423.

[VHP05] S. Virmani, Susana F. Huelga, and Martin B. Plenio. "Classical simulability, entanglement breaking, and quantum computation thresholds". In: *Phys. Rev. A* 71 (4 Apr. 2005), p. 042328. DOI: 10.1103/PhysRevA.71.042328. arXiv: quant-ph/0408076. URL: https://link.aps.org/doi/10.1103/PhysRevA.71.042328.

[Vid03] Guifré Vidal. "Efficient Classical Simulation of Slightly Entangled Quantum Computations". In: *Phys. Rev. Lett.* 91 (14 Oct. 2003), p. 147902. DOI: 10.1103/PhysRevLett.91.147902. arXiv: quant-ph/0301063. URL: https://link.aps.org/doi/10.1103/PhysRevLett.91.147902.

[Vid04] Guifré Vidal. "Efficient Simulation of One-Dimensional Quantum Many-Body Systems". In: *Phys. Rev. Lett.* 93 (4 July 2004), p. 040502. DOI: 10.1103/PhysRevLett.93.040502. arXiv: quant-ph/0310089. URL: https://link.aps.org/doi/10.1103/PhysRevLett.93.040502.

[Vij20] Sagar Vijay. *Measurement-Driven Phase Transition within a Volume-Law Entangled Phase*. 2020. arXiv: 2005.03052 [quant-ph].

[Vil+20] Benjamin Villalonga, Dmitry Lyakh, Sergio Boixo, Hartmut Neven, Travis S Humble, Rupak Biswas, Eleanor G Rieffel, Alan Ho, and Salvatore Mandrà. "Establishing the quantum supremacy frontier with a 281 Pflop/s simulation". In: *Quantum Science and Technology* 5.3 (Apr. 2020), p. 034003. DOI: 10.1088/2058-9565/ab7eeb. arXiv: 1905.00444. URL: https://doi.org/10.1088/2058-9565/ab7eeb.

[WHT15] Dave Wecker, Matthew B Hastings, and Matthias Troyer. "Progress towards practical quantum variational algorithms". In: *Physical Review A* 92.4 (2015), p. 042303.

[Xu+19] Xiaosi Xu, Jinzhao Sun, Suguru Endo, Ying Li, Simon C Benjamin, and Xiao Yuan. "Variational algorithms for linear algebra". In: *arXiv preprint arXiv:1909.03898* (2019).

[Yan+17] Zhi-Cheng Yang, Armin Rahmani, Alireza Shabani, Hartmut Neven, and Claudio Chamon. "Optimizing variational quantum algorithms using Pontryagin's minimum principle". In: *Physical Review X* 7.2 (2017), p. 021027.

[YG17] Man-Hong Yung and Xun Gao. *Can Chaotic Quantum Circuits Maintain Quantum Supremacy under Noise?* 2017. arXiv: 1706.08913.

[Zab+20] Aidan Zabalo, Michael J. Gullans, Justin H. Wilson, Sarang Gopalakrishnan, David A. Huse, and J. H. Pixley. "Critical properties of the measurement-induced transition in random quantum circuits". In: *Phys. Rev. B* 101 (6 Feb. 2020), p. 060301. DOI: 10.1103/PhysRevB.101.060301. arXiv: 1911.00008. URL: https://link.aps.org/doi/10.1103/PhysRevB.101.060301.

[ZLW19] Christa Zoufal, Aurélien Lucchi, and Stefan Woerner. "Quantum Generative Adversarial Networks for learning and loading random distributions". In: *npj Quantum Information* 5.1 (2019), p. 103. DOI: 10.1038/s41534-019-0223-2. URL: https://doi.org/10.1038/s41534-019-0223-2.

[ZN19] Tianci Zhou and Adam Nahum. "Emergent statistical mechanics of entanglement in random unitary circuits". In: *Phys. Rev. B* 99 (17 May 2019), p. 174205. DOI: 10.1103/PhysRevB.99.174205. arXiv: 1804.09737. URL: https://link.aps.org/doi/10.1103/PhysRevB.99.174205.