

21st Conference on the Theory of Quantum Computation, Communication and Cryptography

TQC 2026, August 31–September 4, 2026, Sherbrooke, Québec,
Canada

Edited by

Rotem Arnon

Aram W. Harrow



Editors

Rotem Arnon 

EPFL, Lausanne, Switzerland
rotem.arnon@epfl.ch

Aram W. Harrow 

MIT, Cambridge, MA, USA
aram@mit.edu

ACM Classification 2012

Theory of computation → Computational complexity and cryptography; Theory of computation → Design and analysis of algorithms; Theory of computation → Quantum complexity theory; Theory of computation → Quantum computation theory; Theory of computation → Quantum information theory; Theory of computation → Quantum query complexity

ISBN 978-3-95977-439-0

Published online and open access by

Schloss Dagstuhl – Leibniz-Zentrum für Informatik GmbH, Dagstuhl Publishing, Saarbrücken/Wadern, Germany. Online available at <https://www.dagstuhl.de/dagpub/978-3-95977-439-0>.

Publication date

August, 2026

Bibliographic information published by the Deutsche Nationalbibliothek

The Deutsche Nationalbibliothek lists all publications of this volume in the Deutsche Nationalbibliografie; detailed bibliographic data are available in the Internet at <https://portal.dnb.de>.

License

This work is licensed under a Creative Commons Attribution 4.0 International license (CC-BY 4.0):
<https://creativecommons.org/licenses/by/4.0/legalcode>.



In brief, this license authorizes each and everybody to share (to copy, distribute and transmit) the work under the following conditions, without impairing or restricting the authors' moral rights:

- Attribution: The work must be attributed to its authors.

The copyright is retained by the corresponding authors.

Digital Object Identifier: 10.4230/LIPIcs.TQC.2026.0

ISBN 978-3-95977-439-0

ISSN 1868-8969

<https://www.dagstuhl.de/lipics>

LIPICs – Leibniz International Proceedings in Informatics

LIPICs is a series of high-quality conference proceedings across all fields in informatics. LIPICs volumes are published according to the principle of Open Access, i.e., they are available online and free of charge.

Editorial Board

- Patricia Bouyer-Decitre (Université Paris-Saclay, CNRS, ENS Paris-Saclay, FR)
- Roberto Di Cosmo (Inria and Université Paris Cité, FR)
- Faith Ellen (University of Toronto, CA)
- Holger Hermanns (Universität des Saarlandes, Saarbrücken, DE and Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Wadern, DE)
- Laura Kovács (TU Wien, AT)
- Daniel Král' (Leipzig University, DE and Max Planck Institute for Mathematics in the Sciences, Leipzig, DE – Chair)
- Sławomir Lasota (University of Warsaw, PL)
- Meena Mahajan (Institute of Mathematical Sciences, Chennai, IN)
- Chih-Hao Luke Ong (Nanyang Technological University, SG)
- Eva Rotenberg (IT University of Copenhagen, DK)
- Pierre Senellart (ENS, Université PSL, Paris, France)
- Alexandra Silva (Cornell University, Ithaca, US)

ISSN 1868-8969

<https://www.dagstuhl.de/lipics>

■ Contents

Preface	
<i>Rotem Arnon and Aram W. Harrow</i>	0:vii
Conference Organization	
.....	0:ix
Best Paper Prizes	
.....	0:xi

Papers

A Sharp Computational Phase Transition for the Partition Function of the Transverse-Field Ising Model	
<i>Alistair Sinclair and Thuy-Duong Vuong</i>	1:1–1:20
Characterization of Permutation Gates in the Third Level of the Clifford Hierarchy	
<i>Zhiyang He, Luke Robitaille, and Xinyu Tan</i>	2:1–2:17
Quantum Search with Generalized Wildcards	
<i>Arjan Cornelissen, Nikhil S. Mande, Subhasree Patro, Nithish Raja, and Swagato Sanyal</i>	3:1–3:20
Provable Speedups in Convex Optimization via Quantum Dynamics	
<i>Showvanik Chakrabarti, Dylan Herman, Jacob Watkins, Enrico Fontana, Brandon Augustino, Junhyung Lyle Kim, and Marco Pistoia</i>	4:1–4:25
Quantum Merlin-Arthur with an Internally Separable Proof	
<i>Roozbeh Bassirian, Bill Fefferman, Itai Leigh, Kunal Marwaha, and Pei Wu</i>	5:1–5:13
Limitations of Decoded Quantum Interferometry for MaxCut	
<i>Ojas Parekh</i>	6:1–6:12
On the Complexity of the Circuit Width Problem	
<i>Zhengfeng Ji, Yincheng Liu, and Zhe'ou Zhou</i>	7:1–7:22
Quantum Speedups for Sampling and Non-Convex Optimization with Stochastic Oracles	
<i>Guneykan Ozgul, Xiantao Li, Mehrdad Mahdavi, and Chunhao Wang</i>	8:1–8:25



■ Preface

The 21th Conference on The Theory of Quantum Computation, Communication and Cryptography (TQC) was hosted by Sherbrooke University, Québec, Canada, from August 31 to September 4, 2026.

TQC is a leading annual international conference for students and researchers working in the theoretical aspects of quantum information science. The scientific objective of TQC is to bring together the theoretical quantum information science community to present and discuss the latest advances in the field. Areas of interest for TQC include, but are not restricted to: quantum algorithms and models of quantum computation, quantum information theory, quantum entanglement and foundations, quantum cryptography, quantum communication, quantum complexity, quantum error correction and fault-tolerant quantum computing, tomography, learning, and many-body theory, and other topics in quantum computing (including intersections with the theory of condensed-matter physics and machine learning).

A list of the previous editions of TQC follows:

- TQC 2025, Indian Institute of Science, Bengaluru, India
- TQC 2024, Okinawa Institute for Science and Technology, Japan
- TQC 2023, University of Aveiro, Portugal
- TQC 2022, University of Illinois at Urbana-Champaign, USA
- TQC 2021, University of Latvia, Latvia (virtual conference)
- TQC 2020, University of Latvia, Latvia (virtual conference)
- TQC 2019, University of Maryland, USA
- TQC 2018, University of Technology Sydney, Australia
- TQC 2017, Université Pierre et Marie Curie, France
- TQC 2016, Freie Universität Berlin, Germany
- TQC 2015, Université libre de Bruxelles, Brussels, Belgium
- TQC 2014, National University of Singapore, Singapore
- TQC 2013, University of Guelph, Canada
- TQC 2012, University of Tokyo, Japan
- TQC 2011, Universidad Complutense de Madrid, Spain
- TQC 2010, University of Leeds, UK
- TQC 2009, Institute for Quantum Computing, University of Waterloo, Canada
- TQC 2008, University of Tokyo, Japan
- TQC 2007, Nara Institute of Science and Technology, Nara, Japan
- TQC 2006, NTT R&D Center, Atsugi, Kanagawa, Japan

The conference consisted of invited talks, contributed talks and a poster session. The invited talks were given by Greg Meyer, Omri Shmueli, Sevag Gharibian and Uma Girish.

Submissions were solicited for three tracks: With Proceedings (talk and proceedings), Without Proceedings (talk only) and Poster only submissions (no talk). There were 443 submissions for the three tracks together. The program committee selected 87 submissions for talks, including 8 to be published in the With Proceedings track.

We wish to thank the members of the Program Committee and all subreviewers for their incredible and hard work, without which we could not form an excellent program. We also thank the Local Organizing Committee for all their efforts in organizing the conference and



accommodating the the requests of the Program Committee and the Steering Committee for their guidance, as well as for maintaining the conference's high standards. Last but not least, we thank the authors of all the submissions made to TQC 2026.

Rotem Arnon and Aram W. Harrow

■ Conference Organization

Organizing Committee

- Honghao Fu
- Stefanos Kourtis
- Dave Touchette
- Cunlu Zhou

Program Committee

- Rotem Arnon (Program Committee Chair)
- Aram Harrow (Program Committee Co-Chair)
- Abhinav Deshpande
- Aleksander Kubica
- Alexander Schmidhuber
- Anand Natarajan
- Andreas Winter
- Angela Capel
- Anurag Anshu
- Barak Nehoran
- Chinmay Nirkhe
- Connor Paddock
- David Perez-Garcia
- David Sutter
- Dhara Thakkar
- Dominik Hangleiter
- Eleni Diamanti
- Eric Chitambar
- Eric Culf
- Ernest Tan
- Felix Leditzky
- Francesco Buscemi
- Freek Witteveen
- Giulio Chiribella
- Gorjan Alagic
- Hao-Chung Cheng
- Ian George
- Isaac Kim
- Jean-Daniel Bancal
- Jef Pauwels
- Jiaqing Jiang
- Joe Renes
- Jonas Helsen
- Jonathan Conrad
- Jordi Tura
- Joseph Carolan
- Joseph M. Renes

21st Conference on the Theory of Quantum Computation, Communication and Cryptography (TQC 2026).
Editors: Rotem Arnon and Aram W. Harrow



Leibniz International Proceedings in Informatics
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

- Lewis Woolderton
- Liang Jiang
- Louis Golowich
- Ludovico Lami
- Luowen Qian
- Makrand Sinha
- Marcel Hinsche
- Matthias Christandl
- Michael Wolf
- Mirjam Weilenmann
- Mischa Woods
- Nicholas Spooner
- Nicolas Delfosse
- Peter Brown
- Philippe Faist
- Qisheng Wang
- Ramona Wolf
- Robert T. König
- Sam McArdle
- Sean Hallgren
- Sergey Bravyi
- Shalev Ben David
- Shlok Nahar
- Simon Apers
- Srinivasan Arunachalam
- Tamara Kohler
- Thiago Bergamaschi
- Thomas Hahn
- Tomoyuki Morimae
- Tongyang Li
- Tony Metger
- Tuvia Gefen
- Ulysse Chabaud
- Yihui Quek
- Yuxiang Yang
- Zhiyang He

Steering Committee

- Frédéric Magniez (Steering Committee Chair)
- Bill Fefferman (Steering Committee Co-Chair)
- C.M. Chandrashekar
- Kai-Min Chung
- Shelby Kimmel
- François Le Gall
- Kae Nemoto
- Lúdia del Río

■ Best Paper Prizes

The Program Committee selected as the TQC 2026 Best Paper Prize:

- *The code distance of Floquet codes* by Keller Blackwell and Jeongwan Haah.

The Program Committee selected as the TQC 2026 Best Student Paper Prize:

- *Optimal Qubit Purification and Unitary Schur Sampling via Random SWAP Tests* by Shrigyan Brahmachari, Austin Hulse, Henry Pfister and Iman Marvian.



A Sharp Computational Phase Transition for the Partition Function of the Transverse-Field Ising Model

Alistair Sinclair 

Computer Science Division, UC Berkeley, CA, USA

Thuy-Duong Vuong 

Department of Computer Science and Engineering, UC San Diego, La Jolla, CA, USA

Abstract

We study the problem of approximating the partition function of the transverse-field Ising model (TFIM), a widely studied quantum many-body model with important applications in quantum simulation and quantum annealing. Despite its fundamental importance, the algorithmic landscape for computing the TFIM partition function has remained poorly understood beyond restricted parameter regimes.

We provide a precise characterization of the temperature regimes in which efficient approximation is possible, establishing a sharp computational phase transition. Let J denote the symmetric interaction matrix and $\Delta(J) = \lambda_{\max}(J) - \lambda_{\min}(J)$ denote its spectral width. We show that, for all inverse temperatures $\beta \in [0, 1/\Delta(J)]$, there exists an efficient classical randomized algorithm that approximates the partition function $\text{tr}(e^{-\beta H})$ to within an arbitrarily small multiplicative factor. To obtain this result, we apply the standard Trotter decomposition to map the quantum model to a classical spin system, and then leverage new techniques in Markov chain analysis to derive an efficient algorithm that samples from and computes the partition function of the resulting distribution. This temperature threshold is tight: for $\beta > 1/\Delta(J)$, we show that there exists interaction matrix J for which approximating the partition function, even within an exponential factor, is NP-hard and thus is unlikely to admit an efficient classical or quantum algorithm.

2012 ACM Subject Classification Theory of computation $\rightarrow$ Random walks and Markov chains; Theory of computation $\rightarrow$ Design and analysis of algorithms

Keywords and phrases Computational phase transition, partition function, transverse-field Ising, sampling, counting

Digital Object Identifier 10.4230/LIPIcs.TQC.2026.1

Funding Alistair Sinclair: Supported in part by NSF grant CCF-223109.

1 Introduction

A central problem in quantum statistical mechanics and quantum information theory is to determine when the partition function of a quantum many-body system can be efficiently approximated. The partition function encodes the system's thermodynamic behavior and plays a fundamental role in quantum simulation, quantum phase transitions, and complexity theory [6].

In this work, we consider the *transverse-field Ising model* (TFIM), one of the most extensively studied quantum spin systems. The TFIM serves as a canonical model for quantum phase transitions [26, 12], and plays an important role in quantum annealing [21, 24, 31]. Despite its central role, prior work has made only limited progress in understanding the computational complexity of approximating its partition function, with existing results confined to restricted parameter regimes such as the ferromagnetic case [5] or very high temperature [11].



© Alistair Sinclair and Thuy-Duong Vuong;
licensed under Creative Commons License CC-BY 4.0

21st Conference on the Theory of Quantum Computation, Communication and Cryptography (TQC 2026).

Editors: Rotem Arnon and Aram W. Harrow; Article No. 1; pp. 1:1–1:20

Leibniz International Proceedings in Informatics



Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

We consider the general TFIM with Hamiltonian

$$H = \frac{1}{2} \sum_{i,j \in [n]} J_{ij} Z_i Z_j + \sum_{i=1}^n h_i^z Z_i + \sum_{i=1}^n h_i^x X_i, \quad (1)$$

where $Z_i = I^{\otimes(i-1)} \otimes Z \otimes I^{\otimes(n-i)}$ and $X_i = I^{\otimes(i-1)} \otimes X \otimes I^{\otimes(n-i)}$ denote the Pauli operators acting on site i , with

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

Here $J = (J_{ij})_{i,j \in [n]} \in \mathbb{R}^{n \times n}$ is the *interaction matrix*, $\mathbf{h}^x = (h_i^x)_{i \in [n]} \in \mathbb{R}^n$ is the *transverse field*, and $\mathbf{h}^z = (h_i^z)_{i \in [n]} \in \mathbb{R}^n$ is the *longitudinal field*.

Without loss of generality, we may assume that J is symmetric, as symmetrizing by replacing J with $\frac{J+J^\top}{2}$ leaves the Hamiltonian invariant. Since J is symmetric, its eigenvalues are real. We let $\lambda_{\max}(J)$ and $\lambda_{\min}(J)$ denote the maximum and minimum eigenvalues of J , and let $\Delta(J) = \lambda_{\max}(J) - \lambda_{\min}(J)$ be its spectral width.

We provide a precise characterization of the temperature regimes in which the partition function of the TFIM can be efficiently approximated. Our main result establishes a sharp computational phase transition: for all inverse temperatures β in the interval $[0, \frac{1}{\Delta(J)}]$, there exists an efficient *classical* algorithm for approximating the partition function $\text{tr}(\exp(-\beta H))$ (see Theorem 20 and Theorem 1). Our approximation algorithm applies in a much broader range of parameters than previous works (see Section 1.2 for details).

For $\beta > 1/\Delta(J)$, we show that the problem is NP-hard, and thus unlikely to admit any efficient *classical or quantum* algorithm (see Theorem 2). To the best of our knowledge, this is the first example of a quantum many-body problem where approximating the partition function exhibits a provable computational phase transition.

► **Theorem 1** (Informal; see Theorem 20 for the formal statement). *For any $\beta \in [0, \frac{1}{\Delta(J)}]$, and $\varepsilon, \delta \in (0, 1)$, there exists a classical randomized algorithm running in*

$$\log \delta^{-1} \cdot \text{poly}(n, \beta \|\mathbf{h}^x\|_1, \beta \|\mathbf{h}^z\|_1, \varepsilon^{-1})$$

time that outputs $\hat{Z}$ such that

$$\Pr \left(e^{-\varepsilon} \hat{Z} \leq \text{tr}(\exp(-\beta H)) \leq e^{\varepsilon} \hat{Z} \right) \geq 1 - \delta.$$

► **Theorem 2.** *For any $\varepsilon > 0$, there exists an interaction matrix J such that at $\beta = \frac{1+\varepsilon}{\Delta(J)}$, approximating the partition function $\text{tr}(\exp(-\beta H))$ is NP-hard for any transverse field $\mathbf{h}^x$ satisfying $\|\beta \mathbf{h}^x\|_1 \leq c(\varepsilon)n$, where $c(\varepsilon) > 0$ is a constant dependent only on ε . Moreover, assuming $\text{RP} \neq \text{NP}$ (respectively, $\text{BQP} \not\subseteq \text{NP}$), no polynomial-time classical (respectively, quantum) algorithm can approximate the partition function within a multiplicative factor $\exp(c(\varepsilon)n)$.*

Theorem 2 is a straightforward consequence of [16], which establishes NP-hardness in the classical limit when $\mathbf{h}^x = 0$. Specifically, [16] shows that, when $\mathbf{h}^x = 0$, for any $\varepsilon > 0$ and $\beta > \frac{1+\varepsilon}{\Delta(J)}$, approximating the partition function within an exponential multiplicative factor $\exp(c(\varepsilon)n)$ is NP-hard. In the regime considered in Theorem 2, where the transverse field is dominated by the interaction matrix, the partition function $Z = \text{tr}(\exp(-\beta H))$ is very close to $\text{tr}(\exp(-\beta H_z))$, which is the partition function of a classical Ising model and is therefore NP-hard to approximate in the low-temperature regime $\beta > \frac{1}{\Delta(J)}$ due to [16]; hence, in the same temperature regime, approximating the partition function Z is also NP-hard.

► **Remark 3.** Theorem 2 implies that approximating the partition function is NP-hard when $\beta > \frac{1}{\Delta(J)}$ and the transverse field is dominated by the interaction matrix, i.e., $\|\mathbf{h}^x\|_1 \leq n \cdot O_\varepsilon(1)\Delta(J)$. In the special case of a uniform transverse field, i.e., $h_i^x = h \forall i$, this condition simplifies to $|h| \leq O_\varepsilon(1)\Delta(J)$. On the other hand, when the transverse field is very strong and dominates the interaction matrix, the problem should become easy even at low temperatures, since the system effectively behaves like non-interacting spins. Understanding exactly how the strength of the transverse field affects the temperature at which this transition to an easy regime occurs remains an intriguing open problem.

1.1 Techniques

As in previous works [5, 11], we map the partition function of the transverse-field Ising model (TFIM)

$$Z := \text{tr}(\exp(-\beta H))$$

to the partition function of a classical Ising model using the Trotter decomposition.

For intuition, we sketch this mapping here. Write $H_z := \frac{1}{2} \sum_{i,j \in [n]} J_{ij} Z_i Z_j$ and $H_x := \sum_{i=1}^n h_i^x X_i$; then [5] shows that for $M = \text{poly}(\|\beta H_z\|, \|\beta H_x\|, \frac{1}{\varepsilon})$ a sufficiently large natural number, we have

$$\begin{aligned} Z &\approx \text{tr}((e^{-\frac{\beta H_z}{M}} e^{-\frac{\beta H_x}{M}})^M) \\ &= \sum_{\mathbf{s}^{(1)}, \dots, \mathbf{s}^{(M)}} \prod_{k=1}^M \langle \mathbf{s}^{(k)} | e^{-\frac{\beta H_z}{M}} | \mathbf{s}^{(k)} \rangle \langle \mathbf{s}^{(k)} | e^{-\frac{\beta H_x}{M}} | \mathbf{s}^{(k+1)} \rangle \\ &= A \cdot \sum_{(\mathbf{s}^{(i)})_{i \in [M]}} \exp \left(\frac{-\beta}{2M} \sum_{k=1}^M \sum_{i,j=1}^n J_{ij} s_{k,i} s_{k,j} - \frac{\beta}{M} \sum_{k=1}^M \sum_{i=1}^n h_i^x s_{k,i} + \sum_{i=1}^n \sum_{k=1}^M K_i s_{k,i} s_{k+1,i} \right) \end{aligned} \quad (2)$$

with

$$A = \prod_{i: h_i^x \neq 0} \prod_{i: h_i^x \neq 0} \left(\frac{1}{2} \sqrt{|e^{2h_i^x/M} - e^{-2h_i^x/M}|} \right)^M$$

and $K_i = g(\frac{-\beta h_i^x}{M})$ where $g(a) = \frac{1}{2} \ln \left| \frac{e^a + e^{-a}}{e^a - e^{-a}} \right|$. Here we are using the conventions $\mathbf{s}^{(i)} = (s_{k,i})_{k \in [M]}$ and $\mathbf{s}^{(M+1)} = \mathbf{s}^{(1)}$.

Equation (2) says that, up to the constant factor A , the partition function of the transverse field Ising model is approximately equivalent to the partition function of a classical Ising model defined on nM vertices. This classical model describes a distribution over configurations $(s_{k,i})_{k \in [M], i \in [n]} \in \{\pm 1\}^{nM}$, in which the external field acting on $s_{k,i}$ is $-\beta h_i^x/M$ and the interactions are as follows:

- For any $k \in [M]$ and $i, j \in [n]$, the interaction between $s_{k,i}$ and $s_{k,j}$ is given by $-\beta J_{ij}/M$.
- For any $i \in [n]$ and $k \in [M]$, the interaction between $s_{k,i}$ and $s_{k+1,i}$ is $K_i = g\left(-\frac{\beta h_i^x}{M}\right)$ as defined above.

The following informal statement quantifies the approximation in Equation (2).

► **Lemma 4** (Informal version of Lemma 9). *Consider the TFIM model in Equation (1) and an inverse temperature $\beta \geq 0$. Let $Z = \text{tr}(\exp(-\beta H))$. For any $\varepsilon > 0$, there exists $M = \text{poly}(\|\beta H_z\|, \|\beta H_x\|, \frac{1}{\varepsilon})$ and $H_{\text{classical}} : \{\pm 1\}^{nM} \rightarrow \mathbb{R} \cup \{\pm\infty\}$ corresponding to a classical Ising model so that for $Z' = \sum_{\sigma} \exp(H_{\text{classical}}(\sigma))$, we have*

$$\exp(-\varepsilon)Z' \leq Z \leq \exp(\varepsilon)Z'.$$

Thus, we have reduced the problem of computing the partition function of the transverse field Ising model to that of a classical Ising model on nM vertices, where M is polynomial in the model and accuracy parameters. We can now estimate the partition function of the resulting classical Ising model using classical Markov chain Monte Carlo methods together with counting-to-sampling techniques. Using simulated annealing [13, 30], we reduce the problem of approximating the partition function to sampling from the corresponding Gibbs distribution. Since the classical Ising model arising from the mapping contains interaction terms that depend on the transverse field and may be unbounded, we cannot directly apply existing rapid mixing results based on the boundedness of the interaction matrix [18, 32, 23, 14, 1]. Instead, we view the corresponding classical Ising model as a distribution on $(\mathbb{R}^M)^n$, and invoke a recent result of Anari *et al.* [3] to show that the associated Glauber dynamics is rapidly mixing whenever $\beta < 1/\Delta(J)$. Each update of the Glauber dynamics on the $(\mathbb{R}^M)^n$ model reduces to sampling from a one-dimensional Ising model on M vertices, which can be efficiently implemented using dynamic programming. Combining these ingredients gives an algorithm to approximate the partition function when $\beta < 1/\Delta(J)$; we can push the boundary to $\beta = 1/\Delta(J)$ using a standard perturbative argument. First, we observe that for $\beta < 1/\Delta(J)$, the approximation algorithm's runtime depends polynomially on the gap $\frac{1}{1-\beta\Delta(J)}$. To estimate $\text{tr}(\exp(-\beta H))$ at $\beta = 1/\Delta(J)$, we efficiently estimate the partition function $\text{tr}(\exp(-\beta' H))$ where $\beta' = \beta(1 - \varepsilon/n)$ by noting that $\frac{1}{1-\beta'\Delta(J)} = O(n/\varepsilon)$, then observe that $\text{tr}(\exp(-\beta' H))$ approximates $\text{tr}(\exp(-\beta H))$ within an $\exp(\varepsilon)$ multiplicative factor. For details, see the proof of Theorem 20, which is the formal version of Theorem 1.

1.2 Related work

The problem of approximating partition functions of classical and quantum spin systems has a long history. For many classical spin systems, the computational complexity landscape is now well understood; see, e.g., [28, 29, 10, 8, 9, 16, 14, 1, 2]. In contrast, progress on the quantum side has been more limited, and existing algorithmic results typically require strong structural or high-temperature assumptions.

Several recent works study general quantum Hamiltonians of the form $H = \sum_a \lambda_a G_a$, where $|\lambda_a| \leq 1$, each G_a is a tensor product of single-site Pauli operators acting on a subset a , and the maximum degree of the interaction graph is denoted by $\Delta = \max_i |\{a : i \in a\}|$. In this setting, Haah *et al.* [17] and Bakshi *et al.* [4] give a $(n/\varepsilon)^{O(\log \Delta)}$ -time algorithm that approximates $\text{tr}(e^{-\beta H})$ within a multiplicative factor $1 + \varepsilon$, provided that $\beta < \frac{1}{4e^2(\Delta+1)^2}$. Helmuth and Mann [19] consider classical Hamiltonians perturbed by local quantum terms and obtain efficient approximation algorithms when the perturbation strength is sufficiently small and the inverse temperature lies in an appropriate regime. When specialized to the transverse-field Ising model (TFIM), however, these results impose restrictive bounds on the transverse and longitudinal fields, making them substantially weaker than ours.

Next, we discuss prior works that focus on the TFIM. Brayvi [5] gives an FPRAS for $\text{tr}(e^{-\beta H})$ in the ferromagnetic (stoquastic) regime, where

$$-\beta J_{ij} \geq 0, \quad -\beta h_i^x \geq 0, \quad \text{and} \quad -\beta h_i^z \geq 0.$$

In this case, the Trotter mapping gives a ferromagnetic classical Ising model, allowing the use of classical Markov chain techniques for ferromagnetic systems [20]. We emphasize that our result does not require any sign constraints on the interaction terms J_{ij} or the longitudinal and transverse fields h_i^z, h_i^x .

More recently, Crosson and Slezak [11] establish an FPRAS for the TFIM under a Dobrushin-type condition

$$\beta < \frac{1}{2(\Delta + 2)B}, \quad \text{where} \quad B = \max_{i \neq j} |J_{ij}| \quad \text{and} \quad \Delta = \max_i |\{j \neq i : J_{ij} \neq 0\}|.$$

This condition is strictly stronger than ours, which implies an efficient approximation algorithm whenever $\beta \leq \frac{1}{2\Delta B}$. To see this, assuming without loss of generality that $J_{ii} = 0$ for all $i \in [n]$ ¹, the Gershgorin circle theorem implies

$$\Delta(J) \leq 2\|J\|_\infty \leq 2\Delta B, \quad \text{where} \quad \|J\|_\infty = \max_i \sum_{j \neq i} |J_{ij}|.$$

We remark that the Dobrushin-type conditions $\beta = O(1/\|J\|_\infty)$ or $\beta < O(\frac{1}{\Delta B})$ can be substantially more restrictive than the spectral constraint $\beta < \frac{1}{\Delta(J)}$. In particular, when the interaction terms have mixed signs, significant cancellations may occur, leading to $\Delta(J) \ll \|J\|_\infty$. A concrete and important example arises when the couplings are i.i.d. Gaussian with normalized variance, $J_{ij} \sim \mathcal{N}(0, 1/n)$, as in the Sherrington–Kirkpatrick model and related spin-glass systems [27, 25]. In this case, random matrix theory implies that $\Delta(J) = \Theta(1)$, while $\|J\|_\infty = \Theta(\sqrt{n})$; hence, our Theorem 20 implies an efficient approximation algorithm when $\beta = O(1)$ while [11] requires a much higher temperature $\beta = O(\frac{1}{\sqrt{n}})$. We can also consider diluted spin-glass models [14, 1], where the matrix J is supported on a random d -regular graph G with Rademacher disorder, i.e., $J_{ij} \sim \text{Uniform}(\{\pm 1\})$ if i, j are neighbors in G , and $J_{ij} = 0$ otherwise. For this model, Friedman’s theorem implies that $\Delta(J) \leq 4\sqrt{d-1}$ (see [14, 1] for details), yet $\|J\|_\infty = d$; hence, Theorem 20 implies an efficient approximation algorithm when $\beta = O(\frac{1}{\sqrt{d}})$ while the Dobrushin-type conditions require $\beta = O(\frac{1}{d})$.

1.3 Organization

Following some background and notational preliminaries in Section 2, in Section 3 we spell out the details of how to map the partition function of the transverse-field Ising model to that of a classical Ising model. In Section 4, we show how to sample from the resulting classical Ising model using Markov chains. In Section 5, we combine these ingredients to prove our main results Theorem 20, Theorem 1 and Theorem 2.

2 Preliminaries

2.1 Notation

For a vector $x \in \mathbb{R}^N$, we let $\|x\|$ denote its Euclidean norm. For matrix $A \in \mathbb{R}^{N \times N}$, we let $\|A\|$ denote its operator norm, i.e., $\|A\| = \|A\|_{\text{op}} = \max_{x \neq 0} \frac{\|Ax\|}{\|x\|}$. If A is Hermitian, its eigenvalues are real, and we let $\lambda_{\max}(A)$ and $\lambda_{\min}(A)$ denote its maximum and minimum eigenvalues respectively, and let $\Delta(J) = \lambda_{\max}(A) - \lambda_{\min}(J)$.

¹ Changing diagonal entries rescales the partition function by a constant factor and does not affect approximation complexity.

2.2 Linear algebra

► **Proposition 5.** *Let $H, H' \in \mathbb{R}^{N \times N}$ be Hermitian matrices where $\|H - H'\| \leq \epsilon$. Then*

$$e^{-\epsilon} \operatorname{tr}(\exp(H)) \leq \operatorname{tr}(\exp(H')) \leq e^{\epsilon} \operatorname{tr}(\exp(H)).$$

Proof. This is a simple consequence of Weyl's inequality. Since H, H' are Hermitian, their eigenvalues are real. Let $\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_N$ and $\lambda'_1 \geq \lambda'_2 \geq \dots \geq \lambda'_N$ be the eigenvalues of H and H' , respectively; then by Weyl's inequality

$$|\lambda_i - \lambda'_i| \leq \|H - H'\| \leq \epsilon,$$

and thus

$$\begin{aligned} e^{-\epsilon} \operatorname{tr}(\exp(H)) &= e^{-\epsilon} \sum_i \exp(\lambda_i) \leq \operatorname{tr}(\exp(H')) = \sum_i \exp(\lambda'_i) \\ &\leq e^{\epsilon} \sum_i \exp(\lambda_i) = e^{\epsilon} \operatorname{tr}(\exp(H)). \end{aligned} \quad \blacktriangleleft$$

2.3 Markov chains and mixing time

Our analysis of the mixing time of the Glauber dynamics for the classical distribution arising from the quantum-to-classical mapping is based on establishing a fundamental information-theoretic inequality known as the *Approximate Tensorization of Entropy* (ATE). Let $\Omega = \Omega_1 \times \dots \times \Omega_n$ be a product space. For $i \in [n]$, define the Markov kernel $D_{\sim i}$ as one that maps $x = (x_1, \dots, x_n) \in \Omega$ deterministically to $x_{\sim i}$ by erasing its i th component, i.e.,

$$x_{\sim i} := (x_1, \dots, x_{i-1}, \emptyset, x_{i+1}, \dots, x_n).$$

Let $D_{n \rightarrow n-1}$ be the Markov kernel that picks a coordinate $i \in [n]$ uniformly at random and erases it, i.e., returns the average of $D_{\sim 1}, \dots, D_{\sim n}$. We call this the “down operator.”

The Glauber dynamics associated with a probability distribution μ on Ω is the Markov chain that, at each step, selects a coordinate $i \in [n]$ uniformly at random, erases the current value of x_i , and resamples it according to the conditional distribution $\mu(\cdot \mid x_{\sim i})$. Equivalently, each transition consists of first applying the down operator $D_{n \rightarrow n-1}$ and then restoring the erased coordinate by sampling from the appropriate conditional marginal of μ .

► **Definition 6** (Approximate Tensorization of Entropy, [22, 7]). *Let μ be a probability measure on $\mathbb{R}^n$. We say μ satisfies Approximate Tensorization of Entropy (ATE) with constant C if for every other measure ν absolutely continuous w.r.t. μ ,*

$$\mathcal{D}_{KL}(\nu D_{n \rightarrow n-1} \parallel \mu D_{n \rightarrow n-1}) \leq \left(1 - \frac{1}{Cn}\right) \mathcal{D}_{KL}(\nu \parallel \mu).$$

ATE is a very strong property, which yields an array of useful consequences “for free” once established. In particular, it yields functional inequalities and implies fast mixing of the Glauber dynamics.

► **Proposition 7** (Fact 3.5 of [10]). *If ATE holds with constant C for measure μ , then:*

1. *The MLSI (Modified Log-Sobolev Inequality) for the Glauber dynamics holds with constant $1/Cn$.*

2. As a consequence of the MLSI, on a discrete space, defining

$$\mu_{\min} = \min \{ \mu(x) \mid x \in \text{supp}(\mu) \},$$

the Glauber dynamics on μ has ϵ -mixing time $O(Cn(\log \log(1/\mu_{\min}) + \log(1/\epsilon)))$.

Equivalently, if $\hat{\mu}$ denotes the distribution obtained after

$$t = O(Cn(\log \log(1/\mu_{\min}) + \log(1/\epsilon)))$$

steps of the Glauber dynamics starting from an arbitrary initial state, then $d_{\text{TV}}(\hat{\mu}, \mu) \leq \epsilon$.

3 Mapping the transverse-field Ising model to a classical Ising model

In this section, we reproduce the standard mapping from the partition function of the transverse-field Ising model to that of a classical Ising model using the Trotter decomposition, as was sketched in Section 1.1. While this mapping is standard (see, e.g., [11, 5]), we include it here to keep our discussion self-contained.

We will use the following fact from [5].

► **Proposition 8** ([5, Lemma 3]). Let $Z = \text{tr}(\exp(H_z + H_x))$, $Z' = \text{tr}((e^{\frac{H_z}{M}} e^{\frac{H_x}{M}})^M)$ where $M \in \mathbb{N}$, $M \geq 3$, and

$$H_z = \frac{1}{2} \sum_{i,j \in [n]} J_{ij} Z_i Z_j + \sum_{i \in [n]} h_i^z Z_i; \quad H_x = \sum_{i \in [n]} h_i^x X_i. \quad (3)$$

If $M \geq \max \left\{ 2(\|H_z\| + \|H_x\|), \sqrt{\frac{12(\|H_z\| + \|H_x\|)^3}{\epsilon}} \right\}$ then

$$e^{-\epsilon} Z \leq Z' \leq e^{\epsilon} Z.$$

Next, we record the standard argument that Z' corresponds to the partition function of a classical Ising model. Since the proof is standard, we defer it to the appendix.

► **Lemma 9** ([11, 5]). Let $Z' = \text{tr}((e^{\frac{H_z}{M}} e^{\frac{H_x}{M}})^M)$ where H_z and H_x are defined as in (3).

Let $D_M = \left\{ \frac{1}{\sqrt{M}} \mathbf{x} \mid \mathbf{x} \in \{\pm 1\}^M \right\} \subseteq S^{M-1} = \{ \mathbf{x} \in \mathbb{R}^M \mid \|\mathbf{x}\| = 1 \}$. For $i \in [n]$,

- If $h_i^x = 0$: let $\mathbf{v} = \frac{1}{\sqrt{M}} [1 \ 1 \ \cdots \ 1]^\top$ and $\Omega_i = \{ \mathbf{v}, -\mathbf{v} \} \subseteq D_M$ and let $w^{(i)} : \Omega_i \rightarrow \mathbb{R}$ be defined by $w^{(i)}(\mathbf{v}) = h_i^z$ and $w^{(i)}(-\mathbf{v}) = -h_i^z$.
- If $h_i^x \neq 0$: Let $\Omega_i = D_M$ and $w^{(i)} : \Omega_i \rightarrow \mathbb{R}$ be defined by, for $\mathbf{x} = [x_1 \ x_2 \ \cdots \ x_M]^\top$,

$$w^{(i)}(\mathbf{x}) = \frac{h_i^z}{\sqrt{M}} \sum_{k=1}^M x_k + K_i M (x_1 x_2 + x_2 x_3 + \cdots + x_M x_1),$$

where

$$K_i = \ln \sqrt{\frac{e^{h_i^x/M} + e^{-h_i^x/M}}{|e^{h_i^x/M} - e^{-h_i^x/M}|}} = \frac{1}{2} \ln \left(\frac{e^{h_i^x/M} + e^{-h_i^x/M}}{|e^{h_i^x/M} - e^{-h_i^x/M}|} \right).$$

Let $H_{\text{classical}} : \Omega_1 \times \Omega_2 \times \cdots \times \Omega_n \rightarrow \mathbb{R}_{\geq 0}$ be defined by

$$H_{\text{classical}}((\mathbf{x}_i)_{i \in [n]}) = \frac{1}{2} \sum_{i,j \in [n]} J_{ij} \langle \mathbf{x}_i, \mathbf{x}_j \rangle + \sum_{i=1}^n w^{(i)}(\mathbf{x}_i).$$

Then

$$Z' = A \cdot \sum_{\mathbf{x}_1 \in \Omega_1, \dots, \mathbf{x}_n \in \Omega_n} \exp(H_{\text{classical}}((\mathbf{x}_i)_{i \in [n]})),$$

where

$$A = \prod_{i: h_i^x \neq 0} \left(\frac{1}{2} \sqrt{|e^{2h_i^x/M} - e^{-2h_i^x/M}|} \right)^M.$$

4 Sampling from the classical Ising model

In this section, we will record the ingredients for establishing an efficient algorithm to sample from the classical Ising model that arises from the quantum-to-classical mapping described in Lemma 9 in the regime $\beta \Delta J < 1$. In Section 5, we combine this with the standard counting-to-sampling argument implies that the partition function can be efficiently approximated in the same regime. After appropriate transformations, the classical Ising model can be reformulated as the $O(M)$ -spin model in (4). Lemma 10 establishes *Approximate Tensorization of Entropy* (ATE), which implies fast mixing of the Glauber dynamics by Proposition 7, and each step of the Glauber dynamics can be efficiently implemented by Proposition 12.

► **Lemma 10.** Consider the distribution $\nu_\beta : (S^{M-1})^n \rightarrow \mathbb{R}_{\geq 0}$ defined by

$$\nu_\beta((\mathbf{x}_i)_{i \in [n]}) \propto \exp\left(\frac{\beta}{2} \sum_{i,j \in [n]} J_{ij} \langle \mathbf{x}_i, \mathbf{x}_j \rangle\right) \prod_{i=1}^n \mu^{(i)}(\mathbf{x}_i), \quad (4)$$

where for each $i \in [n]$, $\mu^{(i)} : S^{M-1} \rightarrow \mathbb{R}_{\geq 0}$ is an arbitrary density function on the sphere $S^{M-1} = \{\mathbf{x} \in \mathbb{R}^M : \|\mathbf{x}\| = 1\}$. Suppose $J \succeq 0$ and $\beta \in [0, \frac{1}{\|J\|_{\text{op}}})$. Then the distribution ν_β satisfies ATE with constant $C = \frac{1}{1 - \beta \cdot \|J\|_{\text{op}}}$.

Following [15], we say that a measure $\mu : \mathbb{R}^k \rightarrow \mathbb{R}_{\geq 0}$ is β -semi-log-concave if $\text{Cov}(\mathcal{T}_{\mathbf{w}}\mu) \preceq \beta I$ for every $\mathbf{w} \in \mathbb{R}^k$, where the measure $\mathcal{T}_{\mathbf{w}}\mu : \mathbb{R}^k \rightarrow \mathbb{R}_{\geq 0}$ is defined by $\mathcal{T}_{\mathbf{w}}\mu(\mathbf{x}) \propto \exp(\langle \mathbf{w}, \mathbf{x} \rangle) \mu(\mathbf{x})$. We used the following proposition in the proof of Lemma 10.

► **Proposition 11.** Consider a distribution $\mu : \mathbb{R}^k \rightarrow \mathbb{R}_{\geq 0}$ with

$$\text{supp}(\mu) \subseteq \{\mathbf{x} \in \mathbb{R}^k : \|\mathbf{x}\| \leq 1\}.$$

Then μ is 1-semi-log-concave.

Proof. Let

$$\Sigma := \text{Cov}(\mu) = \mathbb{E}_{X \sim \mu}[(X - \mathbb{E}X)(X - \mathbb{E}X)^\top].$$

We first show that the operator norm of Σ is at most 1.

For any unit vector $\mathbf{v} \in \mathbb{R}^k$, we have

$$\begin{aligned} \mathbf{v}^\top \Sigma \mathbf{v} &= \mathbb{E}_{X \sim \mu}[(\mathbf{v}^\top X - \mathbb{E}[\mathbf{v}^\top X])^2] \\ &= \text{Var}_{X \sim \mu}(\mathbf{v}^\top X) \\ &\leq \mathbb{E}_{X \sim \mu}[(\mathbf{v}^\top X)^2] \\ &\leq \mathbb{E}_{X \sim \mu}[(\|\mathbf{v}\| \|X\|)^2] \\ &= \mathbb{E}_{X \sim \mu}[\|X\|^2] \\ &\leq 1, \end{aligned}$$

where the last inequality follows from the assumption that $\text{supp}(\mu) \subseteq \{\mathbf{x} : \|\mathbf{x}\| \leq 1\}$. Since this holds for all unit vectors $\mathbf{v}$, we conclude that $\|\Sigma\|_{\text{op}} \leq 1$.

Now let $\mathbf{w} \in \mathbb{R}^k$ and consider the tilted measure $\mathcal{T}_{\mathbf{w}}\mu$. Tilting does not change the support of the distribution, so

$$\text{supp}(\mathcal{T}_{\mathbf{w}}\mu) \subseteq \{\mathbf{x} \in \mathbb{R}^k : \|\mathbf{x}\| \leq 1\}.$$

Applying the same argument as above shows that

$$\|\text{Cov}(\mathcal{T}_{\mathbf{w}}\mu)\|_{\text{op}} \leq 1.$$

Therefore, μ is 1-semi-log-concave. ◀

Proof of Lemma 10. This is essentially due to [3, Theorem 9], which assumes that the diagonal of J is constant, i.e., $J_{ii} = \alpha \ \forall i \in [n]$, which is not necessarily true in our case. However, by following that proof, we see that this assumption can be relaxed to $J_{ii} \geq \alpha \ \forall i \in [n]$. To apply their result, we only need to verify that, for each $i \in [n]$ and $s \geq 0$, the distribution $\mathcal{G}_{-s}\mu^{(i)}$ defined by $\mathcal{G}_{-s}\mu^{(i)} \propto \exp(s\|\mathbf{x}\|^2)\mu^{(i)}(\mathbf{x})$ is 1-semi-log-concave. Since $\mu^{(i)}$ is supported on S^{M-1} , the distribution $\mathcal{G}_{-s}\mu^{(i)}$ is simply $\mu^{(i)}$ itself, and is 1-semi-log-concave by Proposition 11. We may now apply [3, Theorem 9] with $\rho(s) = 1 \ \forall s \geq 0$ and $\alpha = 0, \eta = 0$, and deduce that the distribution ν_β satisfies Approximate Tensorization of Entropy with constant

$$C = \exp\left(\int_0^{\|\beta J\|_{\text{OP}}} q(z) dz\right),$$

where $q \equiv q_{\eta, \rho}$ solves the equation

$$q(z) = 1 + \int_0^z q(s)^2 ds.$$

It is easy to see that the function $q(z) = \frac{1}{1-z}$ solves this equation, and hence ν_β satisfies ATE with constant $C = \frac{1}{1-\|\beta J\|_{\text{OP}}}$. The lemma follows. ◀

► **Proposition 12.** Let $H_{\text{classical}} : \Omega_1 \times \Omega_2 \times \cdots \times \Omega_n \rightarrow \mathbb{R}$ be as in Lemma 9. For $\beta \in \mathbb{R}$, let $\nu_\beta : \Omega_1 \times \Omega_2 \times \cdots \times \Omega_n \rightarrow \mathbb{R}_{\geq 0}$ be the distribution defined by $\nu_\beta(\sigma) \propto \exp(\beta H_{\text{classical}}(\sigma))$. Each step of the Glauber dynamics on ν_β samples from a 1-dimensional Ising model, and thus can be done in $O(M^2 + Mn)$ time.

Proof. Each step of Glauber samples from the conditional distribution $\nu(\mathbf{x}_i | (\mathbf{x}_j)_{j \neq i}) : \Omega_i \rightarrow \mathbb{R}_{\geq 0}$. Let $\mathbf{x}_i = \frac{1}{\sqrt{M}}(s_{1,i}, \dots, s_{M,i}) \in D_M$. Sampling from $\nu(\mathbf{x}_i | (\mathbf{x}_j)_{j \neq i})$ reduces to sampling from the Ising model $\pi : \{\pm 1\}^M \rightarrow \mathbb{R}_{\geq 0}$ where

$$\pi((s_{1,i}, \dots, s_{M,i})) \propto \left(\sum_{k=1}^M K_i s_{k,i} s_{k+1,i} + \sum_{i=1}^M \hat{h}_k s_{k,i}\right)$$

and

$$\hat{h}_k = \frac{1}{M} \sum_{j \neq i} J_{ij} s_{k,j} + \frac{1}{M} h_i^z s_{k,i}.$$

Computing $(\hat{h}_k)_{k \in [M]}$ takes $O(Mn)$ time in total, and sampling takes $O(M^2)$ time, as we prove in Proposition 13 below. ◀

► **Proposition 13.** Consider the 1D Ising model $\nu : \{\pm 1\}^M \rightarrow \mathbb{R}_{\geq 0}$ with either the open boundary condition

$$\nu(x) \propto \exp\left(\sum_{i=1}^{M-1} J_{i,i+1} x_i x_{i+1} + \sum_{i=1}^M h_i x_i\right), \quad (5)$$

or the closed boundary condition

$$\nu(x) \propto \exp\left(\sum_{i=1}^M J_{i,i+1} x_i x_{i+1} + \sum_{i=1}^M h_i x_i\right), \quad x_{M+1} = x_1. \quad (6)$$

In both cases, one can compute all relevant conditional marginals exactly and therefore sample from ν in $O(M^2)$ time.

Proof. We show how to compute the conditional marginals

$$\nu(X_i = \pm 1 \mid X_{[i-1]} = \sigma_{[i-1]})$$

exactly by dynamic programming, for any $i \in [M]$ and any partial assignment $\sigma_{[i-1]}$ with $\nu(x_{[i-1]} = \sigma_{[i-1]}) > 0$. Once these marginals are available, we may sample sequentially using the standard autoregressive procedure: sample $\sigma_1 \sim \nu(X_1)$, then sample $\sigma_2 \sim \nu(X_2 \mid X_1 = \sigma_1)$, and so on, until we have sampled $\sigma_n \sim \nu(X_n \mid X_{[n-1]} = \sigma_{[n-1]})$, at which point we output $\sigma = \sigma_{[n]}$. The output distribution is exactly ν .

Open boundary case (5). It suffices to compute $\nu(x_1)$, since for $i > 1$ the conditional distribution $\nu(x_i \mid x_{[i-1]} = \sigma_{[i-1]})$ is itself the open-boundary Ising model on a smaller interval.

Define

$$p_N(\sigma_M) = \exp(h_M \sigma_M), \quad \sigma_M \in \{\pm 1\}.$$

For $k = N - 1, \dots, 1$, compute recursively

$$p_k(\sigma_k) = \sum_{\sigma_{k+1} \in \{\pm 1\}} \exp(J_{k,k+1} \sigma_k \sigma_{k+1} + h_k \sigma_k) p_{k+1}(\sigma_{k+1}).$$

Then

$$\nu(x_1 = \sigma_1) = \frac{p_1(\sigma_1)}{p_1(1) + p_1(-1)}, \quad \sigma_1 \in \{\pm 1\}.$$

Closed boundary case (6). When $i > 1$, conditioning on $x_{[i-1]} = \sigma_{[i-1]}$ breaks the cycle and reduces the problem to the open-boundary case.

For the case $i = 1$, define

$$p_M^{(\sigma_1)}(\sigma_M) = \exp(J_{N,1} \sigma_1 \sigma_M + h_M \sigma_M), \quad (\sigma_1, \sigma_M) \in \{\pm 1\}^2.$$

For $k = M - 1, \dots, 1$, compute

$$p_k^{(\sigma_1)}(\sigma_k) = \sum_{\sigma_{k+1} \in \{\pm 1\}} \exp(J_{k,k+1} \sigma_k \sigma_{k+1} + h_k \sigma_k) p_{k+1}^{(\sigma_1)}(\sigma_{k+1}).$$

Finally, set

$$p_1(\sigma_1) = p_1^{(\sigma_1)}(\sigma_1)$$

and compute the marginal

$$\nu(x_1 = \sigma_1) = \frac{p_1(\sigma_1)}{p_1(1) + p_1(-1)}.$$

Runtime. Each marginal computation requires $O(M)$ operations. Sampling by autoregression requires computing M marginals, so the runtime is $O(M^2)$. ◀

5 On computing the partition function

In this section, we prove our main results, Theorem 1 and Theorem 2 from the Introduction. We begin with Theorem 2, which is a direct consequence of the following result from [16].

► **Theorem 14** ([16, Theorem 1]). *There exists a symmetric matrix $J \in \mathbb{R}^{n \times n}$ with $\Delta(J) = 1 + \varepsilon > 1$ such that approximating the partition function $\text{tr}(\exp(\frac{1}{2} \sum_{i,j} J_{ij} Z_i Z_j)) = \sum_{\sigma \in \{\pm 1\}^n} \exp(\frac{1}{2} \sum_{i,j} J_{ij} \sigma_i \sigma_j)$ within an $\exp(c(\varepsilon)n)$ -multiplicative factor is NP-hard, where the constant $c(\varepsilon) > 0$ depends on ε .*

Proof of Theorem 2. Let $Z = \text{tr}(\exp(-\beta H))$. Let $\tilde{J} = -\beta J$, and $\tilde{Z} = \text{tr}(\exp(-\beta H_z)) = \text{tr}(\exp(\frac{1}{2} \sum_{i,j} \tilde{J}_{ij} Z_i Z_j))$. When $\|\beta \mathbf{h}^x\|_1 \leq \frac{c(\varepsilon)n}{2}$, we have

$$\|-\beta H - (-\beta H_z)\| = \left\| \beta \sum_{i \in [n]} h_i^x X_i \right\| \leq \|\beta \mathbf{h}^x\|_1 \leq \frac{c(\varepsilon)n}{2}$$

and thus by Proposition 5,

$$Z \exp(-\frac{c(\varepsilon)n}{2}) \leq \tilde{Z} \leq Z \exp(\frac{c(\varepsilon)n}{2}).$$

Thus, if we can output a $\hat{Z}$ that approximates Z within a multiplicative factor $\exp(\frac{c(\varepsilon)n}{2})$, then $\hat{Z}$ will approximate $\tilde{Z}$ within a factor of $\exp(c(\varepsilon)n)$. The claim then follows from Theorem 14. ◀

The remainder of this section is devoted to the proof of Theorem 20, which we will restate more formally later as Theorem 1. We first recall some classical results on counting-to-sampling via simulated annealing.

► **Lemma 15.** *Consider a discrete set Ω and a Hamiltonian $H : \Omega \rightarrow [-L, L]$ for some $L \geq 0$. For $\beta \in \mathbb{R}$, let μ_β be the distribution on Ω defined by*

$$\mu_\beta(\sigma) = \frac{\exp(-\beta H(\sigma))}{Z(\beta)}, \quad \text{where } Z(\beta) = \sum_{\sigma \in \Omega} \exp(-\beta H(\sigma)).$$

Suppose we know $Z(\beta_{\min})$, and we have oracle access to the Hamiltonian H and to samples from μ_β for $\beta \in [\beta_{\min}, \beta_{\max}]$. For any $\epsilon > 0$, there exists a (classical) algorithm that outputs an estimator $\hat{Z}$ s.t.

$$\Pr \left((1 - \epsilon)Z(\beta_{\max}) \leq \hat{Z} \leq (1 + \epsilon)Z(\beta_{\max}) \right) \geq 3/4.$$

The algorithm uses $k = O\left(\left(\frac{(\beta_{\max} - \beta_{\min})L}{\epsilon}\right)^2\right)$ samples from μ_β and $O(k)$ Hamiltonian evaluations.

Using a standard coupling argument, we can also relax the precondition of Lemma 15 to allow for samples from a distribution that is close to μ_β .

► **Lemma 16.** *Consider a discrete set Ω and a Hamiltonian $H : \Omega \rightarrow [-L, L]$ for some $L \geq 0$. For $\beta \in \mathbb{R}$, let μ_β be the distribution on Ω defined by*

$$\mu_\beta(\sigma) = \frac{\exp(-\beta H(\sigma))}{Z(\beta)} \quad \text{where } Z(\beta) = \sum_{\sigma \in \Omega} \exp(-\beta H(\sigma)).$$

Suppose we know $Z(\beta_{\min})$, and we have oracle access to the Hamiltonian H and to samples from $\hat{\mu}_\beta$ where $d_{TV}(\hat{\mu}_\beta, \mu_\beta) \leq \epsilon_{TV}$ for $\beta \in [\beta_{\min}, \beta_{\max}]$. For any $\epsilon > 0$, there exists a (classical) algorithm that outputs an estimator $\hat{Z}$ s.t.

$$\Pr\left((1 - \epsilon)Z(\beta_{\max}) \leq \hat{Z} \leq (1 + \epsilon)Z(\beta_{\max})\right) \geq 3/4 - \epsilon_{TV}k.$$

The algorithm uses $k = O\left(\left(\frac{(\beta_{\max} - \beta_{\min})L}{\epsilon}\right)^2\right)$ samples from $\hat{\mu}_\beta$ and $O(k)$ Hamiltonian evaluations.

Lemmas 15 and 16 are standard, and we leave their proofs to the appendix.

► **Remark 17.** A standard application of Chernoff bounds allows us to improve the success probability in Lemmas 15 and 16 to $1 - \delta$ by taking the median of $\Theta(\log(1/\delta))$ independent trials.

In order to apply Lemmas 15 and 16, we need the following proposition which bounds the classical Hamiltonian arising from the quantum-to-classical mapping in Lemma 9, whose proof we leave to the appendix.

► **Proposition 18.** $\forall a \in [-1, 1] \setminus \{0\}$, we have $0 \leq \ln\left(\frac{e^a + e^{-a}}{|e^a - e^{-a}|}\right) \leq \ln\frac{2}{|a|}$.

► **Proposition 19.** Let $H_{\text{classical}} : \Omega_1 \times \Omega_2 \times \cdots \times \Omega_n \rightarrow \mathbb{R}_{\geq 0}$ be as in Lemma 9. Suppose $M \geq |h_i^x| \forall i \in [n]$ and $\min\{|h_i^x| \mid h_i^x \neq 0\} \geq r$. Let

$$L = n \|J\| + \|\mathbf{h}^z\|_1 + nM \ln(2M/r).$$

Then

$$|H((\mathbf{x}_i)_{i \in [n]})| \leq L \quad \forall (\mathbf{x}_i)_{i \in [n]} \in \Omega_1 \times \Omega_2 \times \cdots \times \Omega_n.$$

Proof. Let $H \equiv H_{\text{classical}}$. By the triangle inequality,

$$|H((\mathbf{x}_i)_{i \in [n]})| \leq \left| \sum_{i,j \in [n]} J_{ij} \langle \mathbf{x}_i, \mathbf{x}_j \rangle \right| + \sum_{i \in [n]} |w^{(i)}(\mathbf{x}_i)|. \quad (7)$$

Recall that $\mathbf{x}_i = \frac{1}{\sqrt{M}}(s_{1,i}, \dots, s_{M,i}) \in D_M$; therefore

$$\left| \sum_{i,j \in [n]} J_{ij} \langle \mathbf{x}_i, \mathbf{x}_j \rangle \right| = \left| \frac{1}{M} \sum_{k=1}^M \sum_{i,j \in [n]} J_{ij} s_{k,i} s_{k,j} \right| \leq \frac{1}{M} \sum_{k=1}^M \left| \sum_{i,j \in [n]} J_{ij} s_{k,i} s_{k,j} \right| \leq n \|J\|.$$

For $i \in [n]$ and $\mathbf{x} \in \Omega_i$, we will show

$$|w^{(i)}(\mathbf{x})| \leq |h_i^z| + M \ln(2M/r).$$

If $h_i^x = 0$ then $|w^{(i)}(\mathbf{x})| = |h_i^z|$ and we are done. So we assume $h_i^x \neq 0$, and then for any $\mathbf{x} = [x_1 \ x_2 \ \cdots \ x_M]^\top$, by the triangle inequality

$$\begin{aligned} |w^{(i)}(\mathbf{x})| &\leq \frac{|h_i^z|}{\sqrt{M}} \sum_{k=1}^M |x_k| + |K_i| M (|x_1 x_2| + |x_2 x_3| + \cdots + |x_M x_1|) \\ &= |h_i^z| + |K_i| M, \end{aligned}$$

where we recall

$$K_i = \ln \sqrt{\frac{e^{h_i^x/M} + e^{-h_i^x/M}}{|e^{h_i^x/M} - e^{-h_i^x/M}|}} = \frac{1}{2} \ln \left(\frac{e^{h_i^x/M} + e^{-h_i^x/M}}{|e^{h_i^x/M} - e^{-h_i^x/M}|} \right)$$

satisfies $0 \leq K_i \leq \ln \frac{2M}{|h_i^x|} \leq \ln(2M/r)$ by Proposition 18. Plugging this back into Equation (7) finishes the proof. ◀

Finally, we are ready to prove Theorem 1. In fact, we will prove a more formal version of the theorem, which we now state.

► **Theorem 20** (Formal version of Theorem 1). *Consider the TFIM model in Equation (1). We assume without loss of generality that $J = (J_{ij})_{i,j \in [n]} \in \mathbb{R}^{n \times n}$ is a symmetric matrix. Let $\lambda_{\max}(J)$ and $\lambda_{\min}(J)$ denote its maximum and minimum eigenvalues, and let $\Delta(J) = \lambda_{\max}(J) - \lambda_{\min}(J)$. Let $Z := \text{tr}(\exp(-\beta H))$.*

Suppose $\beta \in [0, \frac{1}{\Delta(J)}]$. Let $\|\mathbf{h}^z\|_1 = \sum_{i=1}^n |h_i^z|$ and $\|\mathbf{h}^x\|_1 = \sum_{i=1}^n |h_i^x|$. For any $\epsilon, \delta \in (0, 1)$, there exists a classical algorithm that runs in

$$\frac{1}{1 - \beta \Delta(J)} \cdot \log \delta^{-1} \cdot \text{poly}(n, \beta \|\mathbf{h}^x\|_1, \beta \|\mathbf{h}^z\|_1, \epsilon^{-1})$$

time and outputs $\hat{Z}$ such that,

$$\Pr \left(e^{-\epsilon \hat{Z}} \leq Z \leq e^{\epsilon \hat{Z}} \right) \geq 1 - \delta.$$

Moreover, for the critical value $\beta = \frac{1}{\Delta(J)}$, the same approximation guarantee holds with runtime $\log \delta^{-1} \cdot \text{poly}(n, \beta \|\mathbf{h}^x\|_1, \beta \|\mathbf{h}^z\|_1, \epsilon^{-1})$.

Proof. We begin with a proof of the first statement; the second statement will follow relatively straightforwardly from the first.

Letting $\tilde{J} = -\beta J + \beta \lambda_{\max}(J) \cdot I$, we have $0 \preceq \tilde{J} \preceq (\beta \Delta(J)) \cdot I \preceq I$. Now let $\tilde{\mathbf{h}}^z = -\beta \mathbf{h}^z$, $\tilde{\mathbf{h}}^x = -\beta \mathbf{h}^x$, and

$$H_z = \frac{1}{2} \sum_{i,j \in [n]} \tilde{J}_{ij} Z_i Z_j + \sum_{i \in [n]} \tilde{h}_i^z Z_i, \quad H_x = \sum_{i \in [n]} \tilde{h}_i^x X_i.$$

This implies that $\text{tr}(\exp(-\beta H)) / \text{tr}(\exp(H_z + H_x)) = c$ for some constant $c > 0$ that we can efficiently compute, so we only need to approximate $Z := \text{tr}(\exp(H_z + H_x))$.

Note that, since H_z is a diagonal matrix with diagonal entries given by $\frac{1}{2} \sum_{i,j \in [n]} \tilde{J}_{ij} z_i z_j + \sum_{i \in [n]} \tilde{h}_i^z z_i$ for $z \in \{\pm 1\}^n$, we obtain the following bounds:

$$\begin{aligned} \|H_z\| &\leq \max_{z \in \{\pm 1\}^n} \left| \frac{1}{2} \sum_{i,j \in [n]} \tilde{J}_{ij} z_i z_j + \sum_{i \in [n]} \tilde{h}_i^z z_i \right| \leq n \|\tilde{J}\| + \|\tilde{\mathbf{h}}^z\|_1 \leq (\beta \Delta(J))n + \|\tilde{\mathbf{h}}^z\|_1; \\ \|H_x\| &\leq \sum_{i \in [n]} |\tilde{h}_i^x| \cdot \|X_i\| = \sum_{i \in [n]} |\tilde{h}_i^x| = \|\tilde{\mathbf{h}}^x\|_1. \end{aligned}$$

Letting

$$C := (\beta \Delta(J))n + \|\tilde{\mathbf{h}}^x\|_1 + \|\tilde{\mathbf{h}}^z\|_1$$

we have $C \geq \|H_z\| + \|H_x\|$. Let $r > 0$ and $M \in \mathbb{N}_{\geq 3}$ be parameters to be chosen later. Letting

$$I_{\text{bad}} = \{ i \in [n] : \tilde{h}_i^x \neq 0 \text{ and } |\tilde{h}_i^x| \leq r \},$$

and

$$\tilde{H} = H_z + \tilde{H}_x, \quad \tilde{H}_x = \sum_{i \in [n] \setminus I_{\text{bad}}} \tilde{h}_i^x X_i,$$

1:14 Computational Phase Transition for TFIM

we get

$$\|H - \tilde{H}\| \leq \sum_{i \in I_{\text{bad}}} |\tilde{h}_i^x| \cdot \|X_i\| \leq rn.$$

Hence by Proposition 5, setting $Z = \text{tr}(\exp(H_z + H_x))$ and $\tilde{Z} = \text{tr}(\exp(H_z + \tilde{H}_x))$ and choosing $r = \frac{\epsilon}{3n}$, we have

$$e^{-\epsilon/3} \tilde{Z} \leq Z \leq e^{\epsilon/3} \tilde{Z}. \quad (8)$$

We now approximate $\tilde{Z}$. Set

$$M = \left(\frac{3C^3}{\epsilon} \right)^{1/2}. \quad (9)$$

By Proposition 8 and Lemma 9, let the classical Hamiltonian $H_{\text{classical}}$ be defined as in Lemma 9 wrt $\tilde{H}$ and M , so that, for

$$\tilde{A} = \prod_{i: \tilde{h}_i^x \neq 0} \left(\frac{1}{2} \sqrt{|e^{2\tilde{h}_i^x/M} - e^{-2\tilde{h}_i^x/M}|} \right)^M,$$

we have

$$Z' = \tilde{A} \sum_{\mathbf{x}_1 \in \Omega_1, \dots, \mathbf{x}_n \in \Omega_n} \exp(H_{\text{classical}}((\mathbf{x}_i)_{i \in [n]}))$$

and

$$\exp(-\epsilon/3) \tilde{Z} \leq Z' \leq \tilde{Z} \exp(\epsilon/3).$$

In particular, combining this with Equation (8) we obtain

$$\exp(-2\epsilon/3) Z' \leq Z \leq Z' \exp(2\epsilon/3). \quad (10)$$

Next, we approximate Z' . Setting

$$L = C + nM \log(M/r) = \tilde{O}(nM),$$

yields

$$|H_{\text{classical}}((\mathbf{x}_i)_{i \in [n]})| \leq L \forall (\mathbf{x}_i)_{i \in [n]} \in \Omega_1 \times \dots \times \Omega_n.$$

For $\beta' \in [0, 1]$, consider the distribution defined by $\nu_{\beta'}((\mathbf{x}_i)_{i \in [n]}) \propto \exp(\beta' H_{\text{classical}}((\mathbf{x}_i)_{i \in [n]}))$. Applying Lemma 10 and Proposition 7 to this distribution, and noting that

$$(\nu_{\beta'})_{\min} := \min \{ \nu_{\beta'}(x) \mid x \in \text{supp}(\nu_{\beta'}) \} \geq \exp(-2L) 2^{-nM},$$

we obtain that the Glauber dynamics² outputs a sample from $\hat{\nu}_{\beta'}$ s.t. $d_{TV}(\hat{\nu}_{\beta'}, \nu_{\beta'}) \leq \epsilon_{TV}$ in

$$T = O\left(\frac{1}{1 - \beta \Delta(J)} \cdot n \log\left(\frac{n}{\epsilon_{TV}} \right) \text{poly log}(C, \epsilon^{-1}) \right)$$

steps, where each step takes $O(M^2)$ time by Proposition 12.

² We can initialize the Glauber dynamics at any $(\mathbf{x}_i)_{i \in [n]} \in \Omega_1 \times \dots \times \Omega_n$, for example, at $(\mathbf{x}_i)_{i \in [n]}$ where $\mathbf{x}_i = \frac{1}{\sqrt{M}} [1 \quad 1 \quad \dots \quad 1]^\top \forall i \in [n]$.

Applying Lemma 16 and Proposition 19 for $\epsilon_{TV} = \frac{1}{8k}$, $\beta_{\min} = 0, \beta_{\max} = 1$ (note that $Z(\beta_{\min}) = \prod_{i=1}^n |\Omega_i|$ can be easily computed and $M \geq C \geq \max |\tilde{h}_i^x|$), we obtain a classical algorithm that outputs $\hat{Z}$ so that with probability $\geq 5/8$,

$$\exp(-\epsilon/3)Z' \leq \hat{Z} \leq Z' \exp(\epsilon/3),$$

which, by Equation (10), implies

$$\hat{Z} \exp(-\epsilon) \leq Z \leq \hat{Z} \exp(\epsilon)$$

using $k = O\left(\frac{L^2}{\epsilon^2}\right)$ samples from $\hat{\nu}_{\beta'}$ and evaluations of $H_{\text{classical}}$.

The overall running time of the algorithm is

$$O(kTM^2 + kn^2M) = \tilde{O}\left(\frac{1}{1 - \beta\Delta(J)} \cdot \epsilon^{-4}n^3((\beta\Delta(J))n + \beta\|\mathbf{h}^x\|_1 + \beta\|\mathbf{h}\|_1^z)^6\right),$$

where $\tilde{O}$ hides polylogarithmic factors.

We can boost the success rate from $5/8$ to $1 - \delta$ for any $\delta > 0$ at the cost of an extra $O(\log(1/\delta))$ factor in the runtime (see Remark 17). This finishes the proof of the first statement.

For the proof of the second statement, where $\beta = \frac{1}{\Delta(J)}$, let $\hat{J} = (1 - \frac{\epsilon}{2n\beta\Delta(J)})\tilde{J}$ and

$$\tilde{Z} = \text{tr}(\exp(\hat{H}_z + H_x)) \quad \text{where} \quad \hat{H}_z = \frac{1}{2} \sum_{i,j \in [n]} \hat{J}_{ij} Z_i Z_j + \sum_{i \in [n]} \tilde{h}_i^z Z_i.$$

Note that

$$\|\hat{H}_z - H_z\| \leq \frac{\epsilon}{2n\beta\Delta(J)} \cdot \max_{z \in \{\pm 1\}^n} \left| \frac{1}{2} \sum_{i,j \in [n]} \tilde{J}_{ij} z_i z_j \right| \leq \frac{\epsilon}{2n\beta\Delta(J)} \cdot n \cdot \|\tilde{J}\| \leq \epsilon/2.$$

By Proposition 5,

$$\tilde{Z} \exp(-\epsilon/2) \leq Z \leq \tilde{Z} \exp(\epsilon/2).$$

Noting that $\|\hat{J}\| \leq 1 - \frac{\epsilon}{2n}$, we can now use the first statement to output $\hat{Z}$ that approximates $\tilde{Z}$ within a factor of $\exp(\epsilon/2)$, and thus approximates Z within a factor of $\exp(\epsilon)$. The runtime guarantee follows from the first statement. $\blacktriangleleft$

References

- 1 Nima Anari, Vishesh Jain, Frederic Koehler, Huy Tuan Pham, and Thuy-Duong Vuong. Entropic independence i: Modified log-Sobolev inequalities for fractionally log-concave distributions and high-temperature Ising models. *arXiv preprint*, 2021. [arXiv:2106.04105](https://arxiv.org/abs/2106.04105).
- 2 Nima Anari, Vishesh Jain, Frederic Koehler, Huy Tuan Pham, and Thuy-Duong Vuong. Entropic independence: optimal mixing of down-up random walks. In *Proceedings of the 54th Annual ACM SIGACT Symposium on Theory of Computing*, STOC 2022, pages 1418–1430, New York, NY, USA, 2022. Association for Computing Machinery. doi:10.1145/3519935.3520048.
- 3 Nima Anari, Frederic Koehler, and Thuy Duong Vuong. Trickle-down in localization schemes and applications. *Proceedings of the 56th Annual ACM Symposium on Theory of Computing*, 2024. URL: <https://api.semanticscholar.org/CorpusID:270392522>.
- 4 Ainesh Bakshi, Allen Liu, Ankur Moitra, and Ewin Tang. High-temperature gibbs states are unentangled and efficiently preparable. *2024 IEEE 65th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 1027–1036, 2024. URL: <https://api.semanticscholar.org/CorpusID:268680477>.

- 5 Sergey Bravyi. Monte carlo simulation of stoquastic hamiltonians. *Quantum Inf. Comput.*, 15:1122–1140, 2014. doi:10.26421/QIC15.13-14-3.
- 6 Sergey Bravyi, Anirban Narayan Chowdhury, David Gosset, and Pawel Wocjan. Quantum hamiltonian complexity in thermal equilibrium. *Nature Physics*, 18:1367–1370, 2021. URL: <https://api.semanticscholar.org/CorpusID:240288401>.
- 7 Pietro Caputo, Georg Menz, and Prasad Tetali. Approximate tensorization of entropy at high temperature, 2014. doi:10.48550/arXiv.1405.0608.
- 8 Xiaoyu Chen, Weiming Feng, Yitong Yin, and Xinyuan Zhang. Rapid mixing of glauher dynamics via spectral independence for all degrees. *2021 IEEE 62nd Annual Symposium on Foundations of Computer Science (FOCS)*, pages 137–148, 2021. URL: <https://api.semanticscholar.org/CorpusID:235254724>.
- 9 Yuansi Chen and Ronen Eldan. Localization schemes: A framework for proving mixing bounds for markov chains (extended abstract). *2022 IEEE 63rd Annual Symposium on Foundations of Computer Science (FOCS)*, pages 110–122, 2022. doi:10.1109/FOCS54457.2022.00018.
- 10 Zongchen Chen, Kuikui Liu, and Eric Vigoda. Optimal mixing of glauher dynamics: Entropy factorization via high-dimensional expansion. In *Proceedings of the 53rd Annual ACM SIGACT Symposium on Theory of Computing*, pages 1537–1550, 2021. doi:10.1145/3406325.3451035.
- 11 Elizabeth Crosson and Samuel Slezak. Classical simulation of high temperature quantum ising models. *Quantum*, 9:1237, 2025. doi:10.22331/q-2025-02-18-1237.
- 12 Amit Dutta, Gabriel Aeppli, Bikas K. Chakrabarti, Uma Divakaran, Thomas F. Rosenbaum, and Diptiman Sen. *Quantum Phase Transitions in Transverse Field Spin Models: From Statistical Physics to Quantum Information*. Cambridge University Press, Cambridge, 2015. doi:10.1017/CB09781107706057.
- 13 Martin Dyer, Leeds K, and Alan Frieze. Computing the volume of convex bodies: A case where randomness provably helps. *Proc Symp Appl Math*, 44, February 1998. doi:10.1090/psapm/044/1141926.
- 14 Ronen Eldan, Frederic Koehler, and Ofer Zeitouni. A spectral condition for spectral gap: fast mixing in high-temperature ising models. *Probability Theory and Related Fields*, 182:1035–1051, 2020. URL: <https://api.semanticscholar.org/CorpusID:220546472>.
- 15 Ronen Eldan and Omer Shamir. Log concavity and concentration of lipschitz functions on the boolean hypercube. *Journal of functional analysis*, 282(8):109392, 2022.
- 16 Andreas Galanis, Alkis Kalavasis, and Anthimos Vardis Kandiros. On Sampling from Ising Models with Spectral Constraints. In Amit Kumar and Noga Ron-Zewi, editors, *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques (APPROX/RANDOM 2024)*, volume 317 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 70:1–70:14, Dagstuhl, Germany, 2024. Schloss Dagstuhl – Leibniz-Zentrum für Informatik. doi:10.4230/LIPIcs.APPROX/RANDOM.2024.70.
- 17 Jeongwan Haah, Robin Kothari, and Ewin Tang. Learning quantum hamiltonians from high-temperature gibbs states and real-time evolutions. *Nature Physics*, 20:1027–1031, 2021. URL: <https://api.semanticscholar.org/CorpusID:282058447>.
- 18 Thomas P Hayes. A simple condition implying rapid mixing of single-site dynamics on spin systems. In *2006 47th Annual IEEE Symposium on Foundations of Computer Science (FOCS'06)*, pages 39–46. IEEE, 2006. doi:10.1109/FOCS.2006.6.
- 19 Tyler Helmuth and Ryan L. Mann. Efficient algorithms for approximating quantum partition functions at low temperature. *Quantum*, 7:1155, October 2023. doi:10.22331/q-2023-10-25-1155.
- 20 Mark Jerrum and Alistair Sinclair. Polynomial-time approximation algorithms for the ising model. *SIAM Journal on Computing*, 22(5):1087–1116, 1993. doi:10.1137/0222066.
- 21 Tadashi Kadowaki and Hidetoshi Nishimori. Quantum annealing in the transverse ising model. *Physical Review E*, 58(5):5355–5363, 1998. doi:10.1103/PhysRevE.58.5355.
- 22 Katalin Marton. An inequality for relative entropy and logarithmic sobolev inequalities in euclidean spaces. *Journal of Functional Analysis*, 264(1):34–61, 2013.

- 23 Katalin Marton. Logarithmic sobolev inequalities in discrete product spaces. *Combinatorics, Probability and Computing*, 28(6):919–935, 2019. doi:10.1017/S0963548319000099.
- 24 R. Martoňák, G. E. Santoro, and E. Tosatti. Quantum annealing by the path-integral monte carlo method: The two-dimensional random ising model. *Physical Review B*, 66(9):094203, 2002. doi:10.1103/PhysRevB.66.094203.
- 25 Marc Mézard, Giorgio Parisi, and Miguel Angel Virasoro. *Spin glass theory and beyond: An Introduction to the Replica Method and Its Applications*, volume 9. World Scientific Publishing Company, 1987.
- 26 Subir Sachdev. *Quantum Phase Transitions*. Cambridge University Press, Cambridge, 2 edition, 2011.
- 27 David Sherrington and Scott Kirkpatrick. Solvable model of a spin-glass. *Physical review letters*, 35(26):1792, 1975.
- 28 Allan Sly. Computational transition at the uniqueness threshold. In *2010 IEEE 51st Annual Symposium on Foundations of Computer Science*, pages 287–296. IEEE, 2010. doi:10.1109/FOCS.2010.34.
- 29 Allan Sly and Nike Sun. The computational hardness of counting in two-spin models on d-regular graphs. *2012 IEEE 53rd Annual Symposium on Foundations of Computer Science*, pages 361–369, 2012. URL: <https://api.semanticscholar.org/CorpusID:1610854>.
- 30 Daniel Stefankovic, Santosh S. Vempala, and Eric Vigoda. Adaptive simulated annealing: A near-optimal connection between sampling and counting. *48th Annual IEEE Symposium on Foundations of Computer Science (FOCS'07)*, pages 183–193, 2006. URL: <https://api.semanticscholar.org/CorpusID:2713057>.
- 31 Yuki Susa, Yusuke Yamashiro, Masayuki Yamamoto, Itay Hen, Daniel A. Lidar, and Hidetoshi Nishimori. Quantum annealing of the p -spin model under inhomogeneous transverse field driving. *Physical Review A*, 98(4):042326, 2018. doi:10.1103/PhysRevA.98.042326.
- 32 Liming Wu. Poincaré and transportation inequalities for gibbs measures under the dobrushin uniqueness condition. *The Annals of Probability*, 34(5):1960–1989, 2006.

A Deferred proofs

Proof of Proposition 8. Inserting resolutions of the identity in terms of the computational basis, i.e., $I = \sum_{\mathbf{s}^{(k)}} |\mathbf{s}^{(k)}\rangle \langle \mathbf{s}^{(k)}|$, we have

$$\begin{aligned}
 Z' &= \text{tr} \left(\left(\sum_{\mathbf{s}^{(1)}} |\mathbf{s}^{(1)}\rangle \langle \mathbf{s}^{(1)}| \right) e^{\frac{H_z}{M}} e^{\frac{H_x}{M}} \left(\sum_{\mathbf{s}^{(2)}} |\mathbf{s}^{(2)}\rangle \langle \mathbf{s}^{(2)}| \right) e^{\frac{H_z}{M}} e^{\frac{H_x}{M}} \cdots \left(\sum_{\mathbf{s}^{(M)}} |\mathbf{s}^{(M)}\rangle \langle \mathbf{s}^{(M)}| \right) e^{\frac{H_z}{M}} e^{\frac{H_x}{M}} \right) \\
 &= \sum_{\mathbf{s}^{(1)}, \dots, \mathbf{s}^{(M)}} \prod_{k=1}^M \langle \mathbf{s}^{(k)} | e^{\frac{H_z}{M}} e^{\frac{H_x}{M}} | \mathbf{s}^{(k+1)} \rangle
 \end{aligned} \tag{11}$$

with periodic boundary condition $\mathbf{s}^{(M+1)} = \mathbf{s}^{(1)}$. Since H_z is diagonal in the computational basis, the above can be rewritten as

$$Z' = \sum_{\mathbf{s}^{(1)}, \dots, \mathbf{s}^{(M)}} \prod_{k=1}^M \langle \mathbf{s}^{(k)} | e^{\frac{H_z}{M}} | \mathbf{s}^{(k)} \rangle \langle \mathbf{s}^{(k)} | e^{\frac{H_x}{M}} | \mathbf{s}^{(k+1)} \rangle. \tag{12}$$

We identify the computational basis vector $\mathbf{s}^{(k)}$ with $(s_{k,1}, \dots, s_{k,N}) \in \{\pm 1\}^n$. Then

$$\begin{aligned}
 \langle \mathbf{s}^{(k)} | e^{\frac{H_z}{M}} | \mathbf{s}^{(k)} \rangle &= \exp \left(\frac{1}{M} \left(\frac{1}{2} \sum_{i,j} J_{ij} s_{k,i} s_{k,j} + \sum_{i=1}^n h_i^z s_{k,i} \right) \right) \\
 &= \exp \left(\frac{1}{2} \sum_{i,j \in [n]} J_{ij} \langle \mathbf{x}_i, \mathbf{x}_j \rangle + \sum_{i \in [n]} \frac{h_i^z}{\sqrt{M}} \sum_{k=1}^M (\mathbf{x}_i)_k \right),
 \end{aligned} \tag{13}$$

where we let $\mathbf{x}_i = \frac{1}{\sqrt{M}}(s_{1,i}, s_{2,i}, \dots, s_{M,i}) \in D_M$.

1:18 Computational Phase Transition for TFIM

Because $H_x = \sum_i h_i^x X_i$, we have

$$e^{H_x} = \prod_i e^{h_i^x X_i}. \quad (14)$$

If $h_i^x = 0$ then $e^{h_i^x X_i} = I$ is diagonal in the computational basis, so

$$\prod_{k=1}^M \langle \mathbf{s}^{(k)} | e^{\frac{H_x}{M}} | \mathbf{s}^{(k+1)} \rangle \neq 0 \Rightarrow s_{k,i} = s_{k+1,i} \quad \forall k \in [M] \Rightarrow \mathbf{x}_i \in \{\pm \mathbf{v}\} = \Omega_i. \quad (15)$$

Next, consider i where $h_i^x \neq 0$. We consider the power e^{aX} of the Pauli matrix $X = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$, where $a \in \mathbb{R}_{\neq 0}$. Since X has eigenvalues $1, -1$ with corresponding eigenvectors $v = \begin{bmatrix} \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \end{bmatrix}$ and $w = \begin{bmatrix} \frac{1}{\sqrt{2}} \\ -\frac{1}{\sqrt{2}} \end{bmatrix}$, e^{aX} has eigenvalues e^a, e^{-a} with the same eigenvectors, and thus

$$e^{aX} = e^a v v^\top + e^{-a} w w^\top = \frac{1}{2} \begin{bmatrix} e^a + e^{-a} & e^a - e^{-a} \\ e^a - e^{-a} & e^a + e^{-a} \end{bmatrix} = C_a \cdot \begin{bmatrix} e^{K_a} & \text{sign}(a) e^{-K_a} \\ \text{sign}(a) e^{-K_a} & e^{K_a} \end{bmatrix}, \quad (16)$$

where $C_a = \frac{1}{2} \sqrt{(e^a + e^{-a}) \cdot |e^a - e^{-a}|} = \frac{1}{2} \sqrt{|e^{2a} - e^{-2a}|}$ and

$$K_a = g(a) = \ln \sqrt{\frac{e^a + e^{-a}}{|e^a - e^{-a}|}} = \frac{1}{2} \ln \frac{e^a + e^{-a}}{|e^a - e^{-a}|}.$$

Let $R = \{i \in [n] \mid h_i^x \neq 0\}$, $R_+ = \{i \in [n] \mid h_i^x > 0\}$ and $R_- = \{i \in [n] \mid h_i^x < 0\}$. For $i \in R$, let

$$K_i = g(h_i^x/M) \quad \text{and} \quad C_i = \frac{1}{2} \sqrt{|e^{2h_i^x/M} - e^{-2h_i^x/M}|}. \quad (17)$$

We have

$$\begin{aligned} & \prod_{i=1}^M \langle \mathbf{s}^{(k)} | e^{\frac{1}{M} H_x} | \mathbf{s}^{(k+1)} \rangle \\ &= \mathbf{1}[s_{k,i} = s_{k+1,i} \forall k \in [M], i \notin R] \left(\prod_{i \in R} \prod_{k=1}^M C_i \exp(K_i s_{k,i} s_{k+1,i}) \right) \prod_{i \in R_-} \prod_{k=1}^M (s_{k,i} s_{k+1,i}) \\ &= \mathbf{1}[s_{k,i} = s_{k+1,i} \forall k \in [M], i \notin R] \prod_{i \in R} \prod_{k=1}^M C_i \exp(K_i s_{k,i} s_{k+1,i}), \end{aligned} \quad (18)$$

where the second equality follows because, for any $i \in [n]$ and assignment $(s_{k,i})_{k \in [M]} \in \{\pm 1\}^M$, $\prod_{k=1}^M (s_{k,i} s_{k+1,i}) = \prod_{k=1}^M s_{k,i}^2 = 1$.

Hence, substituting Equations (13) and (18) into Equation (12), we obtain

$$\begin{aligned} Z' &= \prod_{i \in R} C_i^M \times Z'' \quad \text{with} \\ Z'' &= \sum_{\substack{(\mathbf{s}^{(i)})_{i \in \{\pm 1\}}^n \\ s_{1,i} = \dots = s_{M,i} \forall i \notin R}} \exp \left(\frac{1}{2M} \sum_{k=1}^M \left(\sum_{i,j=1}^n J_{ij} s_{k,i} s_{k,j} + \frac{1}{M} \sum_{i=1}^n h_i^z s_{k,i} + \sum_{i \in R} K_i s_{k,i} s_{k+1,i} \right) \right) \\ &= \sum_{\mathbf{x}_i \in \Omega_i \forall i \in [n]} \exp \left(\frac{1}{2} \sum_{i,j \in [n]} J_{ij} \langle \mathbf{x}_i, \mathbf{x}_j \rangle + \sum_{i \in [n]} w^{(i)}(\mathbf{x}_i) \right) \end{aligned}$$

thus

$$Z' = \prod_{i: h_i^x \neq 0} \left(\frac{1}{2} \sqrt{|e^{2h_i^x/M} - e^{-2h_i^x/M}|} \right)^M \sum_{\mathbf{x}_1 \in \Omega_1, \dots, \mathbf{x}_n \in \Omega_n} \exp(H_{\text{classical}}((\mathbf{x}_i)_{i \in [n]})),$$

which concludes the proof of the lemma. $\blacktriangleleft$

Proof of Lemma 15. For the proof, we will need the following result due to [13] (restated in [30, Theorem 2.2]).

► **Proposition 21** ([30, Theorem 2.2]). *Let $W_1, \dots, W_\ell$ be independent random variables with $\frac{\mathbb{E}[W_i^2]}{\mathbb{E}[W_i]^2} \leq B$ for $i \in [\ell]$. Let $\widehat{W} = W_1 \cdots W_\ell$. Let S_i be the average of $16B\ell/\varepsilon^2$ independent random samples from W_i for $i \in [\ell]$. Let $\widehat{S} = S_1 S_2 \cdots S_\ell$. Then*

$$\Pr \left((1 - \varepsilon) \mathbb{E}[\widehat{W}] \leq \widehat{S} \leq (1 + \varepsilon) \mathbb{E}[\widehat{W}] \right) \geq \frac{3}{4}.$$

Let $\ell = L(\beta_{\max} - \beta_{\min})$, and define the sequence $\beta_{\min} = \beta_0 < \beta_1 < \dots < \beta_\ell = \beta_{\max}$ where $\beta_i = \beta_{\min} + \frac{i}{L} \forall i \in \{0, \dots, \ell\}$. For $i \in \{1, \dots, \ell\}$, let W_i be the random variable obtained by sampling σ from $\mu_{\beta_{i-1}}$ and then outputting the value $\exp(-(\beta_i - \beta_{i-1})H(\sigma))$. Note that

$$\begin{aligned} \mathbb{E}[W_i] &= \sum_{\sigma \in \Omega} \mu_{\beta_{i-1}}(\sigma) \exp(-(\beta_i - \beta_{i-1})H(\sigma)) \\ &= \frac{1}{Z(\beta_{i-1})} \sum_{\sigma \in \Omega} \exp(-(\beta_i - \beta_{i-1})H(\sigma)) \exp(-\beta_{i-1}H(\sigma)) \\ &= \frac{1}{Z(\beta_{i-1})} \sum_{\sigma \in \Omega} \exp(-\beta_i H(\sigma)) \\ &= \frac{Z(\beta_i)}{Z(\beta_{i-1})}, \end{aligned}$$

and thus for $\widehat{W} = W_1 W_2 \cdots W_\ell$,

$$\mathbb{E}[\widehat{W}] = \prod_{i=1}^{\ell} \mathbb{E}[W_i] = \frac{Z(\beta_{\max})}{Z(\beta_{\min})}.$$

Since $(\beta_i - \beta_{i-1}) = \frac{1}{L}$ and $|H(\sigma)| \leq L$, $\frac{\mathbb{E}[W_i^2]}{\mathbb{E}[W_i]^2} \leq B$ for $B = e^2$.

Let S_i be the average of $16B\ell/\varepsilon^2$ independent random samples from W_i for $i \in [\ell]$, and let $\widehat{S} = S_1 S_2 \cdots S_\ell$. The algorithm will output $\widehat{Z} = Z(\beta_{\min})\widehat{S}$. By Proposition 21,

$$\Pr \left((1 - \varepsilon) Z(\beta_{\max}) \leq \widehat{Z} \leq (1 + \varepsilon) Z(\beta_{\max}) \right) = \Pr \left((1 - \varepsilon) \mathbb{E}[\widehat{W}] \leq \widehat{S} \leq (1 + \varepsilon) \mathbb{E}[\widehat{W}] \right) \geq \frac{3}{4}. \blacktriangleleft$$

1:20 Computational Phase Transition for TFIM

Proof of Lemma 16. We use the same algorithm as in Lemma 15. We couple the distributions μ_β and $\hat{\mu}_\beta$ via a coupling Π_β such that

$$\mathbb{P}_{(\sigma, \sigma') \sim \Pi_\beta}[\sigma \neq \sigma'] \leq \epsilon_{TV}.$$

By a union bound, with probability at least $1 - k\epsilon_{TV}$, all k samples drawn from $\hat{\mu}_\beta$ are identical to the corresponding samples drawn from μ_β . Consequently, with probability at least $1 - k\epsilon_{TV}$, the execution of the algorithm using samples from $\hat{\mu}_\beta$ is identical to that of the algorithm using samples from μ_β . Hence the guarantee on the output follows from Lemma 15. $\blacktriangleleft$

Proof of Proposition 18. Clearly, $\frac{e^a + e^{-a}}{|e^a - e^{-a}|} \geq 1$, so $0 \leq \ln \left(\frac{e^a + e^{-a}}{|e^a - e^{-a}|} \right)$.

Wlog, we can assume $a \in (0, 1]$. The inequality $\ln \left(\frac{e^a + e^{-a}}{|e^a - e^{-a}|} \right) \leq \ln \frac{2}{|a|}$ becomes equivalent to

$$f(a) := e^a - e^{-a} - \frac{a}{2} \cdot (e^a + e^{-a}) \geq 0.$$

We calculate

$$f'(a) = e^a + e^{-a} - \frac{1}{2}(e^a + e^{-a}) - \frac{a}{2} \cdot (e^a - e^{-a}) = \frac{1}{2}((e^a + e^{-a}) - a \cdot (e^a - e^{-a}))$$

and

$$2f''(a) = e^a - e^{-a} - (e^a - e^{-a}) - a(e^a + e^{-a}) = -a(e^a + e^{-a}) \leq 0 \quad \forall a \in [0, +\infty).$$

Thus,

$$\forall a \in [0, 1] : f'(a) \geq f'(1) = \frac{1}{e} > 0,$$

and hence

$$\forall a \in [0, 1] : f(a) \geq f(0) = 0. \quad \blacktriangleleft$$

Characterization of Permutation Gates in the Third Level of the Clifford Hierarchy

Zhiyang He 

Department of Mathematics, Massachusetts Institute of Technology, Cambridge, MA, USA

Luke Robitaille 

Department of Mathematics, Massachusetts Institute of Technology, Cambridge, MA, USA

Xinyu Tan 

Department of Mathematics, Massachusetts Institute of Technology, Cambridge, MA, USA

Abstract

The Clifford hierarchy is a fundamental structure in quantum computation whose mathematical properties are not fully understood. In this work, we characterize permutation gates – unitaries which permute the 2^n basis states – in the third level of the hierarchy. We prove that any permutation gate in the third level must be a product of Toffoli gates in what we define as *staircase form*, up to left and right multiplications by Clifford permutations. We then present necessary and sufficient conditions for a staircase form permutation gate to be in the third level of the Clifford hierarchy. As a corollary, we construct a family of non-semi-Clifford permutation gates $\{U_k\}_{k \geq 3}$ in staircase form such that each U_k is in the third level but its inverse is *not* in the k -th level.

2012 ACM Subject Classification Theory of computation → Quantum information theory

Keywords and phrases Quantum fault-tolerance, Clifford hierarchy, permutation gates

Digital Object Identifier 10.4230/LIPIcs.TQC.2026.2

Related Version *Full Version*: <https://arxiv.org/pdf/2510.04993> [24]

Funding *Zhiyang He*: National Science Foundation Graduate Research Fellowship under Grant No. 2141064.

Xinyu Tan: U.S. Department of Energy, Office of Science, National Quantum Information Science Research Centers, Co-design Center for Quantum Advantage (C2QA) under contract number DE-SC0012704.

Acknowledgements The work was conducted as a part of the 2024 Summer Program for Undergraduate Research (SPUR) and 2024-2025 Undergraduate Research Opportunities Program (UROP) at MIT. We thank Jonathan Bloom, Isaac Chuang, David Jerison, and Peter Shor for their mentorship. X. Tan would like to thank Robert Calderbank for introducing the problem of characterizing the third-level Clifford hierarchy to her in 2022 and many insightful discussions afterwards. We thank Jonas Anderson, Jeongwan Haah, Aram Harrow, Greg Kahanamoku-Meyer, Andrey Khesin and Anirudh Krishna for helpful discussions.

1 Introduction

The Clifford hierarchy is a ubiquitous structure in quantum computation which classifies unitary operations based on their conjugate actions on the Pauli group. Precisely, the first level $\mathcal{C}_1$ of the hierarchy $\mathcal{CH}$ is the Pauli group $\mathcal{P}$, and its subsequent levels are defined recursively: $\mathcal{C}_k$ is the set of all unitaries U such that $UPU^\dagger \in \mathcal{C}_{k-1}$ for all Pauli operators P . Within $\mathcal{CH}$, gates in the third level are of unique importance to the study of fault-tolerant quantum computation (FTQC) [33, 34, 19], as established by several foundational works. The Gottesman-Knill theorem [20] states that any circuits made of Pauli and Clifford gates, which are the first two levels of $\mathcal{CH}$, can be efficiently simulated by a classical computer. In contrast,



© Zhiyang He, Luke Robitaille, and Xinyu Tan;

licensed under Creative Commons License CC-BY 4.0

21st Conference on the Theory of Quantum Computation, Communication and Cryptography (TQC 2026).

Editors: Rotem Arnon and Aram W. Harrow; Article No. 2; pp. 2:1–2:17

Leibniz International Proceedings in Informatics



LIPICs Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

adding any non-Clifford gate to the Clifford group forms a universal gate set. Among the non-Clifford unitaries, gates in $\mathcal{CH}$ can be implemented fault-tolerantly by gate teleportation [21], where higher-level gates are performed using resource states and lower-level gates. From this perspective, the non-Clifford gates that are in $\mathcal{C}_3$ are the “easiest-to-implement” non-Clifford gates. Consequently, $\mathcal{C}_3$ gates have been at the center of study for FTQC, with decades of extensive research studying their fault-tolerant implementations [34, 8], synthesis into unitaries [15], algorithmic resource costs [14] and more.

Despite its importance, the mathematical structure of the Clifford hierarchy is not fully understood. Notably, $\mathcal{C}_k$ no longer forms a group for any $k \geq 3$, as it is neither closed under multiplication nor closed under inverse. Ample work has been done to elucidate structures within the hierarchy, which we briefly discuss in Section 1.3.

In this paper, we study the permutation gates, which are n -qubit unitaries that permute the 2^n computational basis states, in $\mathcal{C}_3$. We denote these gates by $\mathcal{C}_3^{\text{sym}}$, and remark that they are both interesting and important for a few reasons. First of all, permutation gates correspond to all reversible classical computations on n bits. Gates in $\mathcal{C}_3$, on the other hand, can be implemented fault-tolerantly using resource states and Clifford gates. $\mathcal{C}_3^{\text{sym}}$ therefore captures classical gates and computational subroutines which are relatively low-cost to implement for FTQC. The canonical example is the Toffoli gate TOF, which is classically universal and ubiquitous in quantum circuits.

Unfortunately, products of Toffoli gates (which capture all permutations) are notoriously unruly. For instance, while a single Toffoli gate is in $\mathcal{C}_3$, a product of as few as two Toffoli gates can leave the hierarchy.¹ Prior work by Anderson [1] conjectured that $\mathcal{C}_3^{\text{sym}}$ are precisely products of pairwise commuting Toffoli gates (up to multiplying on both sides by Clifford permutations), and that $\mathcal{C}_k^{\text{sym}}$ is closed under inverse for all k . We disprove both of these conjectures in this work.

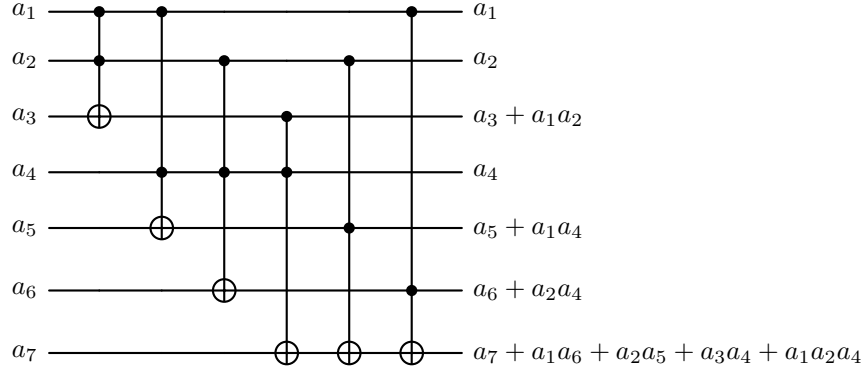
Furthermore, permutation gates are important components for gates in $\mathcal{CH}$. Beigi and Shor [5] proved that all gates in $\mathcal{C}_3$ are *generalized semi-Clifford*, which means they can be written as $\phi_1 \pi d \phi_2$ for Clifford gates ϕ_1, ϕ_2 , a diagonal gate d , and a permutation gate π . The conjecture that all gates in $\mathcal{CH}$ are generalized semi-Clifford remains open [36]. In [13], Cui, Gottesman, and Krishna fully characterized all the diagonal gates in $\mathcal{CH}$. Therefore, characterizing the permutation gates is a crucial step towards characterizing all gates in $\mathcal{C}_3$ and potentially $\mathcal{CH}$.

In this work, we present a full characterization of $\mathcal{C}_3^{\text{sym}}$ (Result 1, Result 2), elucidating an essential substructure of $\mathcal{C}_3$. We then present a family of gates in $\mathcal{C}_3^{\text{sym}}$, which disproves Anderson’s conjectures and challenges prior understanding of $\mathcal{C}_3$ (Result 3). Finally, we derive lower bounds on the number of qubits a gate in $\mathcal{C}_3^{\text{sym}}$ must be supported on given the “complexity” of the function it implements, showing that our construction is optimal (Result 4).

1.1 Main results and techniques

To characterize the gates in $\mathcal{C}_3^{\text{sym}}$, we study structured products of Toffoli gates. We define a product of distinct Toffoli gates to be in *staircase form* if each gate $\text{TOF}_{i,j,k}$ in the product has $i, j < k$ and the target qubit indices are nondecreasing in the order that the gates are applied. For example, the gate in Figure 1 is in staircase form.

¹ Specifically, $\text{TOF}_{1,2,3} \text{TOF}_{1,3,2}$ is not in $\mathcal{CH}$; see equation (E.2) of [1].



■ **Figure 1** A permutation gate in staircase form, consisting of 6 Toffoli gates.

Our first main result, which was presented in an earlier version of this paper [23], states that every permutation in $\mathcal{C}_3^{\text{sym}}$ can be written in staircase form.

► **Result 1** (Theorem 25). *If $\pi \in \mathcal{C}_3$ is a permutation gate, then there exist Clifford permutations ϕ_1, ϕ_2 and a product μ of Toffoli gates in staircase form such that $\mu \in \mathcal{C}_3$ and $\pi = \phi_1\mu\phi_2$.*

We remark that there are permutations in staircase form which are not in $\mathcal{C}_3$. To derive a full characterization, we consider bilinear products defined over vectors of $\mathbb{F}_2^n$, which we denote by juxtaposition. For a commutative and associative bilinear product, we call it a *descending multiplication* if for standard basis vectors $\{e_i\}_{i \in [n]}$, it holds that $e_i e_i = 0$, and $e_i e_j$ (for $i < j$) is in the span of $\{e_k : k > j\}$ (see Definition 33). Our second main result identifies a one-to-one correspondence between gates in $\mathcal{C}_3^{\text{sym}}$ and descending multiplications.

► **Result 2** (Theorem 34). *Every staircase form permutation gate in $\mathcal{C}_3$ induces a descending multiplication over $\mathbb{F}_2^n$. In correspondence, every descending multiplication over $\mathbb{F}_2^n$ induces a staircase form permutation gate in $\mathcal{C}_3$.*

Using this characterization, we construct a novel family of permutation gates in $\mathcal{C}_3$. For each integer $k \geq 3$, we take $n = 2^k - 1$ and label each standard basis vector of $\mathbb{F}_2^n$ by e_S for a nonempty subset $S \subseteq [k]$. We define a bilinear product operation by setting $e_S e_T = e_{S \cup T}$ if $S \cap T = \emptyset$, and $e_S e_T = 0$ if $S \cap T \neq \emptyset$, and extending linearly. This gives a descending multiplication, which induces a staircase form $\mathcal{C}_3$ permutation gate U_k .

► **Result 3** (Theorem 42). *For all $k \geq 3$, we have $U_k \in \mathcal{C}_3$ but $U_k^\dagger \notin \mathcal{C}_k$.*

We note that the permutation in Figure 1 is precisely U_3 . This construction brings new understanding to the study of $\mathcal{C}_3$ gates in a few ways. First of all, $\{U_k\}_{k \geq 3}$ are the first examples of permutation gates in $\mathcal{C}_3$ whose inverses leave $\mathcal{C}_3$ (in fact, leave any fixed level of $\mathcal{CH}$). As a result, they disprove both conjectures of Anderson [1]. Previously, Gottesman and Mochon presented a gate in $\mathcal{C}_3$ (which is not a permutation; see Lemma 10 of main text) whose inverse is not in $\mathcal{C}_3$. Our U_3 is actually equivalent to their example up to conjugation by a Clifford gate.

The permutation implemented by U_k is also interesting. For a permutation π on n bits, we represent it by n Boolean polynomials in the n input variables, one polynomial for each output bit. For example, $\text{CNOT}_{1,2}$ maps $(a_1, a_2) \mapsto (a_1, a_2 + a_1)$ and $\text{TOF}_{1,2,3}$ maps $(a_1, a_2, a_3) \mapsto (a_1, a_2, a_3 + a_1a_2)$. For U_k , if we consider qubits $1, 2, 4, 8, \dots, 2^{k-1}$

as “controls”, and set the other $2^k - 1 - k$ “ancilla” input variables to be zero, then the $2^k - 1$ output polynomials representing U_k are precisely all the non-constant monomials of $a_1, a_2, a_4, a_8, \dots, a_{2^{k-1}}$. In other words, arbitrarily powerful classical computation can be implemented using one $\mathcal{C}_3$ gate followed by CNOT gates, at the expense of a large number of ancilla qubits. This extra ancilla cost is, in fact, optimal: using the characterization with descending multiplications, we show the following lower bound.

► **Result 4 (Theorem 47).** *Any permutation gate in $\mathcal{C}_3$ with an output bit represented by a polynomial of degree at least k must be supported on at least $2^k - 1$ qubits.*

1.2 Future directions

Combining these four results, our characterization discovers and delineates a novel design space within $\mathcal{C}_3^{\text{sym}}$. By constructing descending multiplications, which are easier to reason with than Toffoli circuits, we can derive permutation gates in $\mathcal{C}_3$ which encode different classical computations. These computations can then be implemented in FTQC via gate teleportation, where the majority of the cost is offloaded to resource state preparation. It would be interesting to explore whether the cost of important FTQC primitives, such as arithmetic, can be reduced with this approach.

In addition to gate design and teleportation, it would be interesting to explore whether our results can be generalized to higher levels of $\mathcal{CH}$. Evidently, characterizing more or all of the permutation gates in $\mathcal{CH}$ is consequential to our understanding of $\mathcal{CH}$. Our results mark a concrete step in this direction.

Another important direction of research is to understand the product of permutation and diagonal gates. Significant progress was made in [1], where Anderson showed several results which, in the case of $\mathcal{C}_3$, imply that if $\pi d \in \mathcal{C}_3$, then $\pi \in \mathcal{C}_3$ and $d \in \mathcal{CH}$. Since every gate in $\mathcal{C}_3$ is generalized semi-Clifford, given our characterization of permutation gates in $\mathcal{C}_3$ and the characterization of diagonal gates in $\mathcal{CH}$ by [13], can we precisely characterize all of $\mathcal{C}_3$? Such a result would have significant implications for our study of FTQC, given the unique importance of $\mathcal{C}_3$ gates.

Finally, our Result 4 raises an interesting complexity theory question. Specifically, while $\{U_k\}_{k \geq 3}$ demonstrate that polynomials of arbitrarily high degree can be encoded into a single $\mathcal{C}_3$ gate, such encodings necessarily incur an ancilla cost exponential in k . In contrast, implementing a degree k monomial in $\mathcal{C}_{k+1}$ takes only one k -controlled NOT gate, supported on $k+1$ qubits. How does this tradeoff between polynomial degree and ancilla cost transform through the different levels of $\mathcal{CH}$? What implications does this tradeoff have on the fundamental cost of gate teleportation, which is ubiquitous in FTQC?

1.3 Prior works

Since its introduction by Gottesman and Chuang in 1999 [21], the Clifford hierarchy has been studied by many prior works. Early works defined and studied the (generalized) semi-Clifford gates [37, 18, 22, 36, 5]. Subsequent works have elucidated various structures within the qubit Clifford hierarchy [6, 30, 32, 1, 4, 2] and the qudit Clifford hierarchy [13, 16, 17]. As the original motivation comes from FTQC, prior works have studied characterizations of and improved gate teleportation protocols for semi-Clifford gates [37, 36, 16, 12]; efficient resource state preparation methods, notably magic state distillation [8, 7, 10, 28, 35], for various non-Clifford gates; and constructions and limitations of quantum codes which admit transversal implementation of gates in different levels of the hierarchy [9, 3, 27, 26, 25].

1.4 Organization

The content of the paper is divided into sections as follows. Section 2 gives background results and lemmas for later use. In particular, Section 2.2 discusses representation of permutation gates as polynomials over $\mathbb{F}_2$. In Section 3, we present the definition for staircase form, in which all permutations in $\mathcal{C}_3$ can be written (up to Clifford permutations). In Section 4, we define descending multiplications and prove their one-to-one correspondence with staircase form permutations in $\mathcal{C}_3$. In Section 5, we construct our family of non-semi-Clifford gates $\{U_k\}_{k \geq 3}$ where each permutation U_k is in staircase form. We prove in Theorem 42 that each $U_k \in \mathcal{C}_3$ but $U_k^\dagger \notin \mathcal{C}_k$. In particular, the smallest example in this family U_3 is a 7-qubit permutation, and it is conjugate to the Gottesman–Mochon example by a Clifford operator. All omitted proofs can be found in the full version of this paper [24].

2 Preliminaries

The *Pauli group* on n qubits, denoted as $\mathcal{P}_n$, is the collection of all gates of the form $cP_1 \otimes P_2 \otimes \cdots \otimes P_n$ for $c \in \{\pm 1, \pm i\}$ and one-qubit gates $P_1, \dots, P_n \in \{I_2, X, Y, Z\}$. In particular, we denote the set of all the n -qubit Pauli X operators as $\mathcal{X} = \{I_2, X\}^{\otimes n}$. The *Clifford group* on n qubits is the normalizer of the Pauli group in the unitary group. It can be generated by the Pauli group, the Hadamard and phase gate on each qubit, the CNOT gate on each ordered pair of distinct qubits, and $\{cI : |c| = 1\}$. Henceforth we refer to elements of $\{cI : |c| = 1\}$ as *phases* (not to be confused with the phase gate).

The action of CNOT can be viewed as $|a_1\rangle \otimes |a_2\rangle \mapsto |a_1\rangle \otimes |a_1 + a_2\rangle$ where the first qubit is the *control* and the second qubits is the *target*. Similarly, we can view the Toffoli gate as $|a_1\rangle \otimes |a_2\rangle \otimes |a_3\rangle \mapsto |a_1\rangle \otimes |a_2\rangle \otimes |a_3 + a_1a_2\rangle$ where the first two qubits are controls and the third is the target. We use subscripts to denote the qubits that a gate acts upon. For example, Y_4 is a Pauli Y gate acting on the fourth qubit, and $\text{CNOT}_{3,1}$ is a CNOT gate with the third qubit as control and the first qubit as target.

2.1 Gates in the Clifford hierarchy

The Clifford hierarchy is defined recursively, with the first level being the Pauli gates.

► **Definition 5** (The Clifford hierarchy). *Let n be the number of qubits. Let $\mathcal{C}_1 = \mathcal{P}_n$. For $k \geq 2$, inductively define $\mathcal{C}_k$ to be the set of all unitaries U such that $UPU^\dagger \in \mathcal{C}_{k-1}$ for all $P \in \mathcal{P}_n$. Note that $\mathcal{C}_2$ is the Clifford group. The set $\mathcal{CH} := \mathcal{C}_1 \cup \mathcal{C}_2 \cup \mathcal{C}_3 \cup \dots$ is called the Clifford hierarchy; we refer to $\mathcal{C}_k$ as the k -th layer of $\mathcal{CH}$.*

We list a few standard facts of the Clifford hierarchy.

► **Fact 6.**

1. For any k , $\mathcal{C}_k$ is finite up to phase and $\mathcal{C}_k \subseteq \mathcal{C}_{k+1}$.
2. For $k \geq 2$, $\mathcal{C}_k$ is closed under left and right multiplication of Clifford gates.
3. For $k \geq 3$, $\mathcal{C}_k$ is not a group.
4. For any k , $\mathcal{C}_k$ is closed under complex conjugation.

We say that a gate is a *permutation gate* if it corresponds to a $2^n \times 2^n$ permutation matrix. Note that this is different from only permuting the qubits. Note that $\mathcal{X}$ is exactly the set of all permutation gates in $\mathcal{P}_n$. A gate is called *diagonal* if its associated matrix is diagonal.

► **Definition 7** (Semi-Clifford and generalized semi-Clifford gates). *A gate is semi-Clifford if it can be written as $\phi_1 d \phi_2$ for some Clifford gates ϕ_1, ϕ_2 and a diagonal gate d . A gate is generalized semi-Clifford if it can be written as $\phi_1 \pi d \phi_2$ for some Clifford gates ϕ_1, ϕ_2 , a permutation gate π , and a diagonal gate d .*

Observe that the inverse of a semi-Clifford gate is semi-Clifford. The inverse of a generalized semi-Clifford gate is generalized semi-Clifford, as we can write $(\phi_1 \pi d \phi_2)^{-1} = \phi_2^{-1} \pi^{-1} (\pi d^{-1} \pi^{-1}) \phi_1^{-1}$, and $\pi d^{-1} \pi^{-1}$ is diagonal. If we multiply a semi-Clifford (resp. generalized semi-Clifford) element on the left or right by a Clifford gate, the resulting operator is still semi-Clifford (resp. generalized semi-Clifford).

For a maximal abelian subgroup A of $\mathcal{P}_n$, denote by $\text{span}(A)$ its linear span with complex coefficients. The lemma below shows equivalent definitions of (generalized) semi-Clifford gates.

► **Lemma 8.** *An operator U is semi-Clifford if and only if there exist maximal abelian subgroups A_1 and A_2 of $\mathcal{P}_n$ such that $U A_1 U^\dagger = A_2$. An operator U is generalized semi-Clifford if and only if there exist maximal abelian subgroups A_1 and A_2 of $\mathcal{P}_n$ such that $U \text{span}(A_1) U^\dagger = \text{span}(A_2)$.*

We note that in literature, semi-Clifford and generalized semi-Clifford are usually defined as in Lemma 8 whereas Definition 7 is proved as a proposition [18, 36, 5, 1]. For proofs of this equivalence, we refer readers to Appendix A of [1].

► **Proposition 9** (Semi-Clifford gates are closed under taking inverses). *For any k , the inverse of any semi-Clifford element of $\mathcal{C}_k$ is in $\mathcal{C}_k$.*

Proof. For any $U \in \mathcal{C}_k$ that is semi-Clifford, by Definition 7, we can write $U = \phi_1 d \phi_2$ for some Clifford gates ϕ_1, ϕ_2 and a diagonal gate d . Using Fact 6 repeatedly, we know that $d = \phi_1^{-1} U \phi_2^{-1} \in \mathcal{C}_k$. Hence, $d^{-1} = d^\dagger = \bar{d} \in \mathcal{C}_k$ and thus $U^{-1} = \phi_2^{-1} d^{-1} \phi_1^{-1} \in \mathcal{C}_k$. ◀

It was conjectured in [36] that all gates in $\mathcal{C}_3$ are semi-Clifford. This was disproved by Gottesman and Mochon [5] with the following 7-qubit unitary.

► **Lemma 10.** *For $n = 7$, $\mathcal{C}_3$ contains a non-semi-Clifford element.*

Proof. Let G be the 7-qubit gate given by

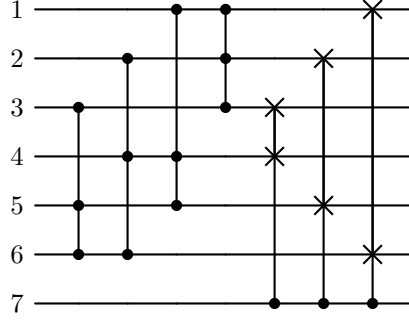
$$G = \text{CSWAP}_{7,1,6} \text{CSWAP}_{7,2,5} \text{CSWAP}_{7,4,3} \cdot \text{CCZ}_{1,2,3} \text{CCZ}_{1,4,5} \text{CCZ}_{2,4,6} \text{CCZ}_{3,5,6},$$

where CSWAP denotes the controlled SWAP gate and CCZ denotes the controlled controlled Z gate. See Figure 2 for a circuit diagram.

It can be verified with a computer program that $G \in \mathcal{C}_3$. If G were semi-Clifford, then we would have $G^{-1} \in \mathcal{C}_3$ by Proposition 9. However, a computer calculation shows that $G^{-1} \notin \mathcal{C}_3$ (in particular, $G^{-1} X_7 G \notin \mathcal{C}_2$). Thus, G is not semi-Clifford. ◀

This 7-qubit operator is the smallest known example of a non-semi-Clifford operator in $\mathcal{C}_3$. [36] showed that for $n \leq 3$, all elements of $\mathcal{C}_3$ are semi-Clifford. Recently, [2] showed that for $n = 4$, all elements of $\mathcal{C}_3$ are semi-Clifford. For $n = 5$ or 6 , it is an open problem whether there is a $\mathcal{C}_3$ operator that is non-semi-Clifford. We make partial progress on this problem in Appendix A of the full version of this paper [24] by showing that all permutation gates in $\mathcal{C}_3$ on at most six qubits are semi-Clifford.

While the semi-Clifford characterization of $\mathcal{C}_3$ was disproved, Beigi and Shor [5] proved that every gate in $\mathcal{C}_3$ is generalized semi-Clifford.



■ **Figure 2** Circuit diagram for the Gottesman–Mochon seven-qubit gate G (with time flowing from left to right).

► **Theorem 11.** *Every element of $\mathcal{C}_3$ is generalized semi-Clifford.*

The conjecture that every gate in $\mathcal{CH}$ is generalized semi-Clifford [36] remains open. Partial progress was made in [1].

► **Conjecture 12.** *Every element of the Clifford hierarchy is generalized semi-Clifford.*

► **Remark 13.** A generalized semi-Clifford gate takes the form $\phi_1 \pi d \phi_2$. A similar form of $\pi d \phi$ is considered in the context of approximate unitary designs or pseudorandom unitaries in [29, 11], where ϕ and π are sampled uniformly at random from their respective groups, and d is a diagonal gate with random ± 1 entries.

2.2 Polynomial representations of permutations

To study the permutation gates in the Clifford hierarchy, we represent them as collections of Boolean polynomials.

► **Fact 14.** *Any function from $\mathbb{F}_2^n$ to $\mathbb{F}_2$ can be uniquely written as an n -variable polynomial that has degree at most 1 in each variable.*

► **Lemma 15.** *For any function $f : \mathbb{F}_2^n \rightarrow \mathbb{F}_2$, the diagonal gate $\sum_{a \in \mathbb{F}_2^n} (-1)^{f(a)} |a\rangle\langle a|$ is in $\mathcal{C}_k$ if and only if f , considered as a polynomial, has degree at most k .*

Proof. This is a special case of the main theorem in [13, See Eq. (1)]. ◀

► **Definition 16** (Polynomial representation). *Given a permutation gate $\pi : \mathbb{F}_2^n \rightarrow \mathbb{F}_2^n$, let $\pi_i : \mathbb{F}_2^n \rightarrow \mathbb{F}_2$ denote the function π restricted to the i -th output bit, i.e.*

$$\pi = \sum_{a \in \mathbb{F}_2^n} |\pi_1(a), \dots, \pi_n(a)\rangle\langle a|.$$

From Fact 14 we know that each π_i can be written as a polynomial in the input bits. We refer to $(\pi_1, \dots, \pi_n)$ as the polynomial representation of π and π_i as the i -th coordinate of π .

As an example, $\text{TOF}_{1,2,3}$ can be represented as $(a_1, a_2, a_3) \mapsto (a_1, a_2, a_3 + a_1 a_2)$. We prove a few useful lemmas regarding the polynomial representations of permutation gates.

► **Lemma 17.** *For any integer $k \geq 1$ and permutation gate $\pi \in \mathcal{C}_{k+1}$, each coordinate of π^{-1} has degree at most k .*

Proof. For each $i \in [n]$, we have

$$\mathcal{C}_k \ni \pi Z_i \pi^{-1} = \sum_{a \in \mathbb{F}_2^n} (-1)^{a_i} |\pi(a)\rangle \langle \pi(a)| = \sum_{a \in \mathbb{F}_2^n} (-1)^{(\pi^{-1}(a))_i} |a\rangle \langle a| = \sum_{a \in \mathbb{F}_2^n} (-1)^{\pi_i^{-1}(a)} |a\rangle \langle a|. \quad (1)$$

By Lemma 15, one has that π_i^{-1} , the i -th coordinate of π^{-1} , must have degree at most k . $\blacktriangleleft$

► **Remark 18.** For $\pi \in \mathcal{C}_3$, Lemma 17 tells us that every coordinate of π^{-1} has degree at most 2; however, as we will see in Section 5, the coordinates of π themselves do not necessarily have degree at most 2.

We denote by $e_1, \dots, e_n$ the standard basis of $\mathbb{F}_2^n$.

► **Proposition 19** (Clifford permutations). *For any $n \times n$ invertible matrix M over $\mathbb{F}_2$ and any vector w , the permutation gate sending $|v\rangle \mapsto |Mv + w\rangle$ is Clifford. Conversely, any Clifford permutation is of this form for some M and w .*

Recall that $\mathcal{X}$ is the set of all the n -qubit Pauli X operators.

► **Proposition 20** (Pauli permutations). *Suppose $X'_1, X'_2, \dots, X'_m \in \mathcal{X}$ are independent (that is, no nontrivial product of them yields the identity). Then there exists some Clifford permutation ν such that $\nu|0^n\rangle = |0^n\rangle$ and $\nu X_i \nu^{-1} = X'_i$ for all $i \in [m]$.*

The proofs of Propositions 19 and 20 can be found at the full version of this paper [24].

2.3 Anderson's conjectures

Besides polynomials, we consider the more operational representation of permutation gates as products of multi-controlled NOT gates, which we denote by $C^k X$ for $k \geq 0$ (note that Toffoli is $C^2 X$). In [1], Anderson considered *mismatch-free* circuits, which are products of pairwise commuting multi-controlled NOT gates.

► **Theorem 21** (Theorem D.4 of [1]). *A mismatch-free permutation circuit is in $\mathcal{CH}$ at the level of the highest-level gate in the circuit.*

Following this classification, Anderson gave two conjectures on permutation gates in $\mathcal{CH}$.

► **Conjecture 22.** *A permutation is in $\mathcal{C}_3$ if and only if it can be written as a circuit of commuting Toffoli gates, up to left and right multiplication by Clifford gates.*

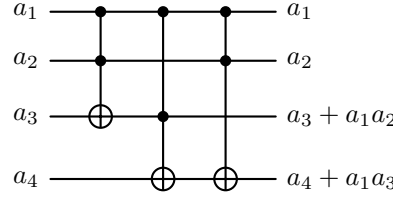
► **Conjecture 23.** *For a permutation $\pi \in \mathcal{C}_k$, we have $\pi^\dagger \in \mathcal{C}_k$.*

In Appendix B of the full version of this paper [24], we show that mismatch-free permutation circuits are precisely the semi-Clifford permutation gates. However, as we will see in the next few sections, the permutation gates in $\mathcal{C}_3$ form a much richer space, which we characterize in this work. In particular, we disprove both Conjecture 22 and Conjecture 23.

3 Staircase form representations of $\mathcal{C}_3$ permutations

We begin by presenting the most important definition of our work.

► **Definition 24** (Staircase form Toffoli circuits). *A product of pairwise distinct Toffoli gates is said to be in staircase form if each gate $\text{TOF}_{i,j,k}$ in the product has $i < j < k$ and the target qubits are in nondecreasing order in the order that the gates are applied. See Figure 3 for an example.*



■ **Figure 3** This circuit for $\text{TOF}_{1,2,4}\text{TOF}_{1,3,4}\text{TOF}_{1,2,3}$ is in staircase form but not mismatch-free, as qubit 3 is used as a control for $\text{TOF}_{1,3,4}$ and a target for $\text{TOF}_{1,2,3}$.

Note that a staircase form Toffoli circuit is unique up to ordering of gates with the same target. We say that a permutation is in staircase form if it can be written as a Toffoli circuit in staircase form.

Our main result of this section is the following.

► **Theorem 25.** *Suppose $\pi \in \mathcal{C}_3$ is a permutation gate. Then there exist Clifford permutations ϕ_1, ϕ_2 and a staircase form permutation $\mu \in \mathcal{C}_3$ such that $\pi = \phi_1\mu\phi_2$.*

Since $\pi \in \mathcal{C}_3$ is a permutation, each $\pi X_j \pi^{-1}$ is a Clifford permutation. Then by Proposition 19, we can find a binary matrix A_j and a vector b_j such that $\pi X_j \pi^{-1}$ implements the permutation $|v\rangle \mapsto |v + A_j v + b_j\rangle$. To prove Theorem 25, we construct a sequence of Clifford operators to reduce A_j, b_j to a specific form, which will help us build a staircase form representation of π .

We caution that Theorem 25 is not if-and-only-if, because there exist permutations in staircase form that are not in $\mathcal{C}_3$. For example, $\pi' = \text{TOF}_{3,4,5}\text{TOF}_{1,2,3}$ is in staircase form, but $\pi' X_1 \pi'^{-1} = X_1 \text{CNOT}_{2,3} \text{TOF}_{2,4,5} \notin \mathcal{C}_2$.

3.1 Reducing $\mathcal{C}_3$ permutations

The goal of this subsection is to reduce a $\mathcal{C}_3$ permutation to a much more friendly form, which serves as the first step in proving Theorem 25:

► **Lemma 26** (Reducing $\mathcal{C}_3$ permutations). *For any permutation gate $\pi \in \mathcal{C}_3$, there exist Clifford permutations ϕ_1, ϕ_2 such that the following is true. Let $\tau = \phi_1\pi\phi_2$. Then $\tau|0^n\rangle = |0^n\rangle$ and for each i , we have $\tau|e_i\rangle = |e_i\rangle$ and $\tau X_i \tau^{-1}$ is given by $|v\rangle \mapsto |v + A_i v + e_i\rangle$ for some strictly lower triangular matrix A_i over $\mathbb{F}_2$.*

To prove Lemma 26, we will use a series of helper lemmas which can be proved via standard linear algebra arguments.

The first helper lemma stated below is essentially the same as standard results on simultaneous triangularization of commuting nilpotent matrices; see, for example, [31]. We include a proof in the full version of this paper [24].

► **Lemma 27.** *Suppose $A_1, \dots, A_k$ are linear transformations of an n -dimensional vector space V over a field F such that $A_j^2 = 0$ and $A_i A_j = A_j A_i$. Then there exists a basis of F^n in which all the A_i are strictly lower triangular. Recall that a matrix is strictly lower triangular if it is lower triangular, and all diagonal elements are 0.*

For any nonzero column vector v over $\mathbb{F}_2$, let $\alpha(v)$ denote the index of its first nonzero component. Set $\alpha(0) = \infty$ as convention.

► **Lemma 28.** *Suppose A is an $n \times n$ strictly lower triangular matrix over $\mathbb{F}_2$, and b is a nonzero column vector in $\mathbb{F}_2^n$. Then $\alpha(Ab) > \alpha(b)$.*

Proof. This follows directly from the definition of strictly lower triangular. ◀

Lemma 28 will be used tacitly throughout what follows.

► **Proposition 29.** *Suppose we have a list of tuples $(A_1, b_1), \dots, (A_n, b_n)$, where each A_i is an $n \times n$ strictly lower triangular matrix over $\mathbb{F}_2$ and each b_i is a column vector in $\mathbb{F}_2^n$. Suppose we can perform the following operations:*

1. “Swap”: swap the indices of two pairs (A_i, b_i) and (A_j, b_j) , or
2. “Compose”: choose two distinct indices i and j , and update A_i to be $A_i + A_j + A_i A_j$ and update b_i to be $b_i + b_j + A_i b_j$.

Then it is possible to perform operations either to reach a state where $b_i = e_i$ for all i , or to reach a state where some b_i is 0.

Using Propositions 19, 20, and 29 and Lemma 27, we can prove Lemma 26. The omitted proofs of Proposition 29 and Lemma 26 can be found at the full version of this paper [24].

3.2 Proof of Theorem 25

Now we only need one more ingredient to prove Theorem 25: the polynomial representations of staircase form permutations, whose proof follows intuitively from the circuit.

► **Lemma 30** (Characterization of staircase form permutation via the polynomial representation of its inverse). *A permutation gate π is staircase form if and only if, in the polynomial representation of π^{-1} , for all k , the k -th coordinate is a_k plus a (possibly empty) sum of terms of the form $a_i a_j$ with $i < j < k$.*

Furthermore, given a permutation π written as a staircase form Toffoli circuit, for any $i < j < k$, we have that $\text{TOF}_{i,j,k}$ appears in the product if and only if the k -th coordinate in the polynomial form of π^{-1} contains an $a_i a_j$ term.

Proof. If π is a product of Toffoli gates in staircase form, then π^{-1} is the product of those same Toffoli gates in reverse order. In that product, whenever a gate is applied, its controls have never been targeted so far, and thus are unchanged from the input. This means that, when performing the gates of π^{-1} in order, a gate $\text{TOF}_{i,j,k}$ adds a term of $a_i a_j$ to the k -th coordinate of the output. All parts of the desired result now can be easily shown. ◀

We are now ready to prove Theorem 25, restated below for convenience.

► **Theorem 25.** *Suppose $\pi \in \mathcal{C}_3$ is a permutation gate. Then there exist Clifford permutations ϕ_1, ϕ_2 and a staircase form permutation $\mu \in \mathcal{C}_3$ such that $\pi = \phi_1 \mu \phi_2$.*

Proof. By Lemma 26, we can assume without loss of generality that $\pi|0^n\rangle = |0^n\rangle$ and for each $i \in [n]$, $\pi|e_i\rangle = |e_i\rangle$ and $\pi X_i \pi^{-1}$ is given by $|v\rangle \mapsto |v + A_i v + e_i\rangle$ for some strictly lower triangular matrix A_i over $\mathbb{F}_2$.

We now show that for any v , if $\pi|v\rangle = |w\rangle$, then $\alpha(v) = \alpha(w)$. Suppose for the sake of contradiction that this is false. We know it is true for $v = 0^n$ or e_n , so $\alpha(v) < n$ in any counterexample. Take the largest k for which there exists a counterexample with $\alpha(v) = k$. We know $\pi|e_k\rangle = |e_k\rangle$, so $v \neq e_k$. Let $u \neq 0^n$ be such that $\pi|v + e_k\rangle = |u\rangle$. Then $\alpha(v + e_k) > k$, so $\alpha(u) = \alpha(v + e_k) = m$ for some $m > k$. We have

$$|w\rangle = \pi|v\rangle = \pi X_k |v + e_k\rangle = \pi X_k \pi^{-1} |u\rangle = |u + A_k u + e_k\rangle.$$

Since $\alpha(u) = m > k$, and $\alpha(A_k u) > m > k$, we must have $\alpha(w) = \alpha(e_k + u + A_k u) = k = \alpha(v)$, which is a contradiction.

We now build a polynomial representation (see Definition 16) for π^{-1} . By Lemma 17, every coordinate of π^{-1} has degree at most 2. Since $\pi^{-1}|0^n\rangle = |0^n\rangle$ and $\pi^{-1}|e_i\rangle = |e_i\rangle$ for all i , we have that the constant term of every coordinate is 0 and the linear term of the i -th coordinate is a_i for all i . Thus we can write π^{-1} as

$$|a_1, \dots, a_n\rangle \mapsto |a_1 + q_1, \dots, a_n + q_n\rangle,$$

where each q_k is a sum of some (possibly zero) monomials of the form $a_i a_j$ with $i < j$.

For any $i < j$, we have $\pi^{-1}|e_i + e_j\rangle = |e_i + e_j + w_{ij}\rangle$, where w_{ij} has ones exactly at the positions k for which q_k contains the monomial $a_i a_j$. Let v be such that $\pi|e_j + w_{ij}\rangle = |v\rangle$, we have

$$\pi X_i \pi^{-1}|v\rangle = \pi X_i|e_j + w_{ij}\rangle = \pi|e_i + e_j + w_{ij}\rangle = |e_i + e_j\rangle = |v + A_i v + e_i\rangle,$$

which means $v + A_i v = e_j$, and $j = \alpha(v + A_i v) = \alpha(v)$. Since $\pi|e_j + w_{ij}\rangle = |v\rangle$, we must also have $\alpha(e_j + w_{ij}) = \alpha(v)$. Thus $\alpha(e_j + w_{ij}) = j$, which means $\alpha(w_{ij}) > j$. In other words, any appearance of an $a_i a_j$ term can only be in a q_k with $k > j$. Then Lemma 30 implies that π is in staircase form.

Thus, unraveling our without-loss-of-generality assumptions, we can write $\pi = \phi_1 \mu \phi_2$ for Clifford permutations ϕ_1 and ϕ_2 and a staircase form permutation μ . Finally, $\mu = \phi_1^{-1} \pi \phi_2^{-1}$ is in $\mathcal{C}_3$ by Fact 6, since $\phi_1^{-1}, \phi_2^{-1} \in \mathcal{C}_2$ and $\pi \in \mathcal{C}_3$. ◀

3.3 A consequence of Theorem 25

Below, in Corollary 32, we strengthen the result of Beigi and Shor [5], who showed that any $\mathcal{C}_3$ element is generalized semi-Clifford (restated in Theorem 11). Our corollary sharpens this by showing that the underlying permutation can be taken to be in staircase form.

► **Lemma 31.** *For any permutation gate π and diagonal gate d , if $\pi d \in \mathcal{C}_3$, then $\pi \in \mathcal{C}_3$.*

Proof. This is a special case of Corollary A.11.2 from [1]. ◀

► **Corollary 32.** *For any $\psi \in \mathcal{C}_3$, there exist $\phi_1, \phi_2 \in \mathcal{C}_2$, a diagonal gate d , and a staircase form permutation $\pi \in \mathcal{C}_3$ with $\psi = \phi_1 \pi d \phi_2$.*

Proof. By Theorem 11, we can write $\psi = \phi_3 \pi_0 d_0 \phi_4$ for $\phi_3, \phi_4 \in \mathcal{C}_3$, a permutation gate π_0 , and a diagonal gate d_0 . Now $\pi_0 d_0 = \phi_3^{-1} \psi \phi_4^{-1} \in \mathcal{C}_3$, so $\pi_0 \in \mathcal{C}_3$ by Lemma 31. Then, by Theorem 25, we can write $\pi_0 = \phi_5 \mu \phi_6$ for Clifford permutations ϕ_5, ϕ_6 and a staircase form permutation $\mu \in \mathcal{C}_3$. Now $\phi_6 d_0 \phi_6^{-1}$ is diagonal as ϕ_6 is a permutation and d_0 is diagonal. Let $\phi_1 = \phi_3 \phi_5$, $\pi = \mu$, $d = \phi_6 d_0 \phi_6^{-1}$, $\phi_2 = \phi_6 \phi_4$. Then $\phi_1, \phi_2 \in \mathcal{C}_2$, π is a staircase form permutation in $\mathcal{C}_3$, and d is diagonal, and

$$\phi_1 \pi d \phi_2 = \phi_3 \phi_5 \mu \phi_6 d_0 \phi_6^{-1} \phi_6 \phi_4 = \phi_3 \phi_5 \mu \phi_6 d_0 \phi_4 = \phi_3 \pi_0 d_0 \phi_4 = \psi. \quad \blacktriangleleft$$

4 Descending multiplications

As we proved that every permutation in $\mathcal{C}_3$ can be written in staircase form, in this section we show sufficient conditions for a staircase form permutation to be in $\mathcal{C}_3$. Previously we have been working with $\mathbb{F}_2^n$ as a vector space without any multiplicative structure. Our next definition captures bilinear products defined over $\mathbb{F}_2^n$ which encode staircase form Toffoli circuits in $\mathcal{C}_3$.

► **Definition 33** (Descending multiplications). A map $\mathbb{F}_2^n \times \mathbb{F}_2^n \rightarrow \mathbb{F}_2^n$, denoted by juxtaposition, is called a descending multiplication if

- it is linear in each coordinate (distributive), associative, and commutative,
- for all $i \in [n]$, we have $e_i e_i = e_i^2 = 0$, and
- for all $i < j \in [n]$, we have $e_i e_j$ is in the span of $\{e_k : k > j\}$.

Observe that, for any descending multiplication, we have $v^2 = 0$ for all v . Henceforth, for any permutation gate π , we will also interpret π as a permutation of $\mathbb{F}_2^n$, so that whenever $\pi|v\rangle = |w\rangle$, we can write $\pi(v) = w$. We will also write 0 and 0^n interchangeably.

We now state the main theorem of this section.

► **Theorem 34** (A bijection between descending multiplications and $\mathcal{C}_3$ permutations in staircase form). There is a one-to-one correspondence of descending multiplications to $\mathcal{C}_3$ permutations in staircase form, which we describe as follows. For each descending multiplication, the corresponding $\mathcal{C}_3$ permutation π is given by

$$\forall S \subseteq [n], \quad \pi \left| \sum_{i \in S} e_i \right\rangle = \left| \sum_{T \subseteq S, T \neq \emptyset} \prod_{i \in T} e_i \right\rangle. \quad (2)$$

Each permutation $\pi \in \mathcal{C}_3$ in staircase form induces a multiplication operation where each $e_i e_j$ is given by

$$\pi |e_i + e_j\rangle = |e_i + e_j + e_i e_j\rangle. \quad (3)$$

The multiplication is then extended linearly.

We break the proof of this theorem into several propositions. We refer readers to the full version [24] of this paper for the proofs of Propositions 35–38.

► **Proposition 35.** For any descending multiplication, the resulting π from Equation (2) is indeed a staircase form permutation in $\mathcal{C}_3$.

► **Proposition 36.** For any staircase form permutation π in $\mathcal{C}_3$, the resulting operation from Equation (3) is indeed a descending multiplication.

► **Proposition 37.** Given a descending multiplication, applying the procedure in Equation (2) to get a permutation π , then applying the procedure in Equation (3) to that permutation, yields the original multiplication.

► **Proposition 38.** Given a staircase form permutation π in $\mathcal{C}_3$, applying the procedure in Equation (3) to get a descending multiplication, then applying the procedure in Equation (2) to that multiplication, yields the original π .

Proof of Theorem 34. Combining Propositions 35–38 yields the theorem. ◀

5 A family of non-semi-Clifford Permutations

Utilizing our characterization of $\mathcal{C}_3$ permutations as descending multiplications, we construct an infinite family of permutation gates $\{U_k\}_{k \geq 3}$ in Definition 39 and prove in Theorem 42 that every U_k is non-semi-Clifford in $\mathcal{C}_3$. This family of gates disproves both of Anderson’s conjectures (restated in Conjecture 22 and Conjecture 23 for convenience). We study the smallest case of $k = 3$ in Section 5.3, and show that this 7-qubit permutation U_3 is in fact conjugate to the Gottesman–Mochon example by a Clifford operator.

► **Definition 39** (A family of permutation gates in $\mathcal{C}_3$ from descending multiplications). For each integer $k \geq 3$, let $n = 2^k - 1$. We label a basis of $\mathbb{F}_2^n$ as e_S for nonempty subsets $S \subseteq [n]$. Define a multiplication by setting $e_{S \cap T} = e_{S \sqcup T}$ if $S \cap T = \emptyset$, and $e_{S \cap T} = 0$ if $S \cap T \neq \emptyset$, and extending linearly; it is easy to check that this yields a descending multiplication. Let U_k be the staircase form $\mathcal{C}_3$ permutation corresponding to this descending multiplication, as in Theorem 34.

► **Proposition 40.** We can express U_k as a product of Toffoli gates in staircase form as follows: for each pair S, T of nonempty disjoint subsets of $[n]$ with $S < T$, apply the gate $\text{TOF}_{S,T,S \sqcup T}$; specifically, apply these gates in nondecreasing order of target.

► **Remark 41.** From the perspective of labeling gates with integers instead of sets, Proposition 40 states that we can express U_k as a product of Toffoli gates in staircase form as follows: for each pair of indices $i < j$ that do not have any 1s in the same place as each other in binary, apply $\text{TOF}_{i,j,i+j}$; specifically, apply these gates in nondecreasing order of target.

The main feature of these permutations is captured by the following theorem.

► **Theorem 42** (Inverses of $\mathcal{C}_3$ permutations may lie outside $\mathcal{C}_k$). For any integer $k \geq 3$, we have $U_k \in \mathcal{C}_3$ but $U_k^{-1} \notin \mathcal{C}_k$. Thus U_k is not semi-Clifford.

5.1 Proof of Proposition 40

► **Proposition 43.** Given a staircase form permutation π in $\mathcal{C}_3$ and $i < j$, let v_{ij} be such that $\pi^{-1}(e_i + e_j) = e_i + e_j + v_{ij}$. Then for all $i < j < k$, $\text{TOF}_{i,j,k}$ appears in the staircase form of π if and only if there is an e_k term in v_{ij} .

Proof. Note that v_{ij} is the vector of the positions containing a $a_i a_j$ monomial in the polynomial form of π^{-1} , so this follows directly from Lemma 30. ◀

Proof of Proposition 40. Take $U_k = \pi$ in Proposition 43. We know from the proof of Proposition 35 that $v_{ij} = \pi^{-1}(e_i e_j)$ for all $i < j$. In the notation of indexing qubits as sets, this becomes $v_{ST} = \pi^{-1}(e_S e_T)$ for $S < T$. Now note that $e_{S \cap T}$ is $e_{S \sqcup T}$ or 0, and in either case $\pi^{-1}(e_{S \cap T}) = e_{S \cap T}$. Thus $v_{ST} = e_{S \cap T}$ in any case, so Proposition 43 implies the desired, by the definition of $e_{S \cap T}$. ◀

5.2 Proof of Theorem 42

► **Proposition 44.** Given a descending multiplication and its corresponding $\mathcal{C}_3$ permutation π , for any nonempty set $S \subseteq [n]$, the value of $\Pi_{i \in S} e_i$ is exactly the vector corresponding to the positions at which an $\Pi_{i \in S} a_i$ term appears in the polynomial representation of π .

Proof. For any S , let p_S be the vector corresponding to the positions at which an $\Pi_{i \in S} a_i$ term appears in the polynomial representation of π . Then we can see, for any set S , that $\pi(\sum_{i \in S} e_i) = \sum_{T \subseteq S} p_T$. From Equation (2), we have that $\sum_{T \subseteq S; T \neq \emptyset} p_T = \sum_{T \subseteq S; T \neq \emptyset} \prod_{i \in T} e_i$ for any nonempty set S . It then easily follows by strong induction on $|S|$ that $p_S = \prod_{i \in S} e_i$ for all nonempty S . ◀

We prove Theorem 42 through the polynomial representations of U_k and U_k^{-1} .

► **Lemma 45.** For each integer $k \geq 3$, let us denote the polynomial representations (Definition 16) of U_k and U_k^{-1} respectively as $(\pi_S)_{S \subseteq [k], S \neq \emptyset}$ and $(\pi'_S)_{S \subseteq [k], S \neq \emptyset}$. Then

$$\pi_S(a) = \sum_{m=1}^{|S|} \sum_{\substack{\sqcup_{i=1}^m T_i = S, \\ T_i \neq \emptyset}} a_{T_1} a_{T_2} \dots a_{T_m}, \quad (4)$$

where the sum is over unordered non-empty subsets $T_1, \dots, T_m$, in other words, all partitions of S , and

$$\pi'_S(a) = a_S + \sum_{T_1 \neq \emptyset, T_1 \sqcup T_2 = S} a_{T_1} a_{T_2},$$

where the sum is over unordered pairs T_1, T_2 .

Proof. The polynomial form of π^{-1} follows directly from Proposition 40 and Lemma 30. For the polynomial form of π , observe that, for any $T_1, \dots, T_m$, we have $e_{T_1} \dots e_{T_m}$ is $e_{T_1 \sqcup \dots \sqcup T_m}$ if the T_i are pairwise disjoint, and 0 if not, so the desired follows from Proposition 44. ◀

Proof of Theorem 42. We know $U_k \in \mathcal{C}_3$ by definition. Also, by Lemma 45, we know that $\pi_{[k]}$ is a polynomial of degree k because it contains the monomial $a_{\{1\}} a_{\{2\}} \dots a_{\{k\}}$. This implies that $U_k^{-1} \notin \mathcal{C}_k$ using Lemma 17. In particular, $U_k^{-1} \notin \mathcal{C}_3$, so U_k is not semi-Clifford by Proposition 9. ◀

5.3 Example: $k = 3$

Let us examine the case of $k = 3$, the simplest gate in the family. U_3 is a 7-qubit permutation gate with 6 Toffoli gates in staircase form (see circuit in Figure 1):

$$U_3 = \text{TOF}_{1,6,7} \text{TOF}_{2,5,7} \text{TOF}_{3,4,7} \text{TOF}_{2,4,6} \text{TOF}_{1,4,5} \text{TOF}_{1,2,3}.$$

Recall the Gottesman–Mochon 7-qubit gate (see Figure 2 for the circuit)

$$\mathcal{C}_3 \ni G = \text{CSWAP}_{7,1,6} \text{CSWAP}_{7,2,5} \text{CSWAP}_{7,4,3} \cdot \text{CCZ}_{1,2,3} \text{CCZ}_{1,4,5} \text{CCZ}_{2,4,6} \text{CCZ}_{3,5,6},$$

which is not semi-Clifford, as in the proof of Lemma 10. Now define a 7-qubit Clifford gate F :

$$F = H_3 H_5 H_6 \text{CNOT}_{6,1} \text{CNOT}_{5,2} \text{CNOT}_{3,4} H_7.$$

► **Proposition 46.** U_3 is a non-semi-Clifford permutation in $\mathcal{C}_3$ on 7 qubits and $FGF^{-1} = U_3$.

Proof. The fact that U_3 is a non-semi-Clifford permutation on 7 qubits is a special case of Theorem 42. Checking that $FGF^{-1} = U_3$ is a direct computation. ◀

In particular, this implies that for all $n \geq 7$, $\mathcal{C}_3$ contains a non-semi-Clifford permutation (see Lemma A.3 in the full version [24]).

5.4 Lower bound on the number of qubits required for a $\mathcal{C}_3$ permutation

From Theorem 42, we know that U_k is a $\mathcal{C}_3$ permutation on $2^k - 1$ qubits that contains a degree- k monomial in its polynomial representation. Our next result shows that any such permutation in $\mathcal{C}_3$ must be supported on at least $2^k - 1$ qubits.

► **Theorem 47** (Lower bound on the number of qubits required for a $\mathcal{C}_3$ permutation). *If π is a $\mathcal{C}_3$ permutation such that there is a degree- k monomial somewhere in the polynomial representation of π , then $n \geq 2^k - 1$; this bound is sharp for all $k \geq 3$.*

We prove Theorem 47 by taking advantage of the one-to-one correspondence between descending multiplications (Proposition 48) and $\mathcal{C}_3$ permutations in staircase form (Proposition 49).

► **Proposition 48.** *Given any descending multiplication, if $\Pi_{i \in S} e_i$ is nonzero for some k -element set S , then $n \geq 2^k - 1$.*

Proof. Suppose $\Pi_{i \in S} e_i$ is nonzero. For any nonempty subset T of S , let $p_T = \Pi_{i \in T} e_i$. We shall show that the vectors p_T , over all nonempty subsets T of S , are linearly independent.

Suppose for contradiction they are linearly dependent. Take a family F of nonempty subsets of S such that $\sum_{T \in F} p_T = 0$. Take a minimal element U of F (i.e. such that no proper subset of U is in F). Let $V = S \setminus U$. Note that for any $T \in F$ with $T \neq U$, we have $T \not\subseteq U$, which means $T \cap V \neq \emptyset$. From Definition 33, we see that $p_T p_V = 0$. Therefore, $\sum_{T \in F} p_T p_V = p_U p_V = p_S$. On the other hand, $\sum_{T \in F} p_T = 0$ implies $(\sum_{T \in F} p_T) p_V = 0$. Thus $p_S = 0$, which is a contradiction. We conclude that the vectors p_T must be linearly independent. Since there are $2^k - 1$ such vectors in the vector space $\mathbb{F}_2^n$, we must have $n \geq 2^k - 1$. ◀

► **Proposition 49.** *If π is a staircase form $\mathcal{C}_3$ permutation such that there is a degree- k monomial somewhere in the polynomial form of π , then $n \geq 2^k - 1$.*

Proof. Suppose $\Pi_{i \in S} a_i$ appears somewhere in the polynomial form of π , for some k -element set S . Then, for the descending multiplication corresponding to π , we have $\Pi_{i \in S} e_i$ is nonzero, by Proposition 44; then Proposition 48 yields the desired. ◀

We are now ready to prove Theorem 47.

Proof of Theorem 47. Assume without loss of generality that $k \geq 2$. By Theorem 25, we can write $\pi = \phi_1 \mu \phi_2$, for Clifford permutations ϕ_1 and ϕ_2 and staircase form $\mu \in \mathcal{C}_3$. Now all terms in the polynomial forms of ϕ_1 and ϕ_2 are degree at most 1; then if all terms in the polynomial form of μ have degree less than k , then all terms in the polynomial form of $\phi_1 \mu \phi_2 = \pi$ have degree less than k , a contradiction. Thus there exists some term in the polynomial form of μ with degree $d \geq k$. Then Proposition 49 applied to μ yields that $n \geq 2^d - 1 \geq 2^k - 1$, as desired. The example of U_k shows that the bound is sharp. ◀

5.5 Rejection of Anderson's conjectures

We now disprove Anderson's two conjectures.

► **Lemma 50.** *Two multi-controlled NOT gates commute if and only if they have no mismatch (that is, there is no qubit that is used as a target in one and a control in the other).*

Proof. The “if” direction is clear. Let us prove the “only if” direction. Assume for the sake of contradiction that they have mismatch. Without loss of generality, let the gates be A , with qubit 1 as a control and qubit 2 as target, and B , with qubit 1 as target. Then $AB|1^n\rangle = A|01^{n-1}\rangle = |01^{n-1}\rangle$, while $BA|1^n\rangle = B|101^{n-2}\rangle$, which is either $|101^{n-2}\rangle$ or $|001^{n-2}\rangle$; in either case $BA|1^n\rangle \neq AB|1^n\rangle$, so they do not commute. ◀

► **Lemma 51.** *Any mismatch-free product μ of C^*X gates is semi-Clifford.*

Proof. Consider an X gate on every target qubit and a Z gate on every non-target qubit. These gates generate a maximal abelian subgroup of $\mathcal{P}_n$ up to phase, and μ will commute with every element of this subgroup. The claim follows from Lemma 8. $\blacktriangleleft$

► **Proposition 52.** *Conjecture 22 and Conjecture 23 are false.*

Proof. Conjecture 23 is false by Theorem 42. For Conjecture 22, suppose it holds. Then by Lemma 50, every permutation in $\mathcal{C}_3$ is a mismatch-free product of Toffoli gates, up to Clifford permutations on the left and right. This implies that every permutation in $\mathcal{C}_3$ is semi-Clifford by Lemma 51, which results in a contradiction, as we know U_3 is a non-semi-Clifford permutation in $\mathcal{C}_3$ for $n = 7$. $\blacktriangleleft$

References

- 1 Jonas T. Anderson. On groups in the qubit Clifford hierarchy. *Quantum*, 8:1370, June 2024. doi:10.22331/q-2024-06-13-1370.
- 2 Jonas T Anderson and Andrew Connelly. Affine equivalence in the clifford hierarchy. *arXiv preprint*, 2025. arXiv:2507.14370.
- 3 Jonas T Anderson and Tomas Jochym-O'Connor. Classification of transversal gates in qubit stabilizer codes. *arXiv preprint arXiv:1409.8320*, 2014. doi:10.48550/arxiv.1409.8320.
- 4 Jonas T Anderson and Matthew Weippert. Controlled gates in the clifford hierarchy. *arXiv preprint*, 2024. arXiv:2410.04711.
- 5 Salman Beigi and Peter W. Shor. C_3 , semi-Clifford and generalized semi-Clifford operations, 2009. arXiv:0810.5108.
- 6 Ingemar Bengtsson, Kate Blanchfield, Earl Campbell, and Mark Howard. Order 3 symmetry in the clifford hierarchy. *Journal of Physics A: Mathematical and Theoretical*, 47(45):455302, 2014.
- 7 Sergey Bravyi and Jeongwan Haah. Magic-state distillation with low overhead. *Physical Review A*, 86(5):052329, November 2012. doi:10.1103/physreva.86.052329.
- 8 Sergey Bravyi and Alexei Kitaev. Universal quantum computation with ideal Clifford gates and noisy ancillas. *Physical Review A*, 71(2):022316, 2005. URL: <https://journals.aps.org/pr/abstract/10.1103/PhysRevA.71.022316>.
- 9 Sergey Bravyi and Robert König. Classification of topologically protected gates for local stabilizer codes. *Physical review letters*, 110(17):170503, 2013.
- 10 Earl T Campbell, Hussain Anwar, and Dan E Browne. Magic-state distillation in all prime dimensions using quantum reed-muller codes. *Physical Review X*, 2(4):041021, 2012.
- 11 Chi-Fang Chen, Jeongwan Haah, Jonas Haferkamp, Yunchao Liu, Tony Metger, and Xinyu Tan. Incompressibility and spectral gaps of random circuits. *arXiv preprint arXiv:2406.07478*, 2024. doi:10.48550/arXiv.2406.07478.
- 12 Imin Chen and Nadish de Silva. Characterising semi-clifford gates using algebraic sets. *Communications in Mathematical Physics*, 405(9):201, 2024.
- 13 Shawn X Cui, Daniel Gottesman, and Anirudh Krishna. Diagonal gates in the Clifford hierarchy. *Physical Review A*, 95(1):012329, 2017. URL: [S.X.Cui,D.Gottesman,andA.Krishna,DiagonalgatesintheCliffordhierarchy,PhysicalReviewA](https://arxiv.org/abs/1608.03912).
- 14 Alexander M Dalzell, Sam McArdle, Mario Berta, Przemyslaw Bienias, Chi-Fang Chen, András Gilyén, Connor T Hann, Michael J Kastoryano, Emil T Khabiboulline, Aleksander Kubica, et al. Quantum algorithms: A survey of applications and end-to-end complexities. *arXiv preprint*, 2023. arXiv:2310.03011.
- 15 Christopher M Dawson and Michael A Nielsen. The solovay-kitaev algorithm. *arXiv preprint quant-ph/0505030*, 2005.
- 16 Nadish de Silva. Efficient quantum gate teleportation in higher dimensions. *Proceedings of the Royal Society A*, 477(2251):20200865, 2021.

- 17 Nadish de Silva and Oscar Lautsch. The clifford hierarchy for one qubit or qudit. *arXiv preprint*, 2025. [arXiv:2501.07939](#).
- 18 Jeroen Dehaene and Bart De Moor. Clifford group, stabilizer states, and linear and quadratic operations over $gf(2)$. *Physical Review A*, 68(4):042318, 2003.
- 19 D. Gottesman. Stabilizer Codes and Quantum Error Correction. *arXiv:quant-ph/9705052*, 1997. doi:10.48550/arxiv.quant-ph/9705052.
- 20 Daniel Gottesman. The heisenberg representation of quantum computers, 1998. [arXiv:quant-ph/9807006](#).
- 21 Daniel Gottesman and Isaac L. Chuang. Demonstrating the viability of universal quantum computation using teleportation and single-qubit operations. *Nature*, 402(6760):390–393, November 1999. doi:10.1038/46503.
- 22 David Gross and Maarten Nest. The lu-lc conjecture, diagonal local operations and quadratic forms over $gf(2)$. *arXiv preprint*, 2007. [arXiv:0707.4000](#).
- 23 Zhiyang He, Luke Robitaille, and Xinyu Tan. Permutation gates in the third level of the clifford hierarchy. *arXiv preprint*, 2024. [arXiv:2410.11818](#).
- 24 Zhiyang He, Luke Robitaille, and Xinyu Tan. Characterization of permutation gates in the third level of the clifford hierarchy. *arXiv preprint*, 2025. [arXiv:2510.04993](#).
- 25 Zhiyang He, Vinod Vaikuntanathan, Adam Wills, and Rachel Yun Zhang. Quantum Codes with Addressable and Transversal Non-Clifford Gates. *arXiv preprint arxiv:2502.01864*, 2025. doi:10.48550/arxiv.2502.01864.
- 26 Jingzhen Hu, Qingzhong Liang, and Robert Calderbank. Climbing the diagonal clifford hierarchy, 2021. [arXiv:2110.11923](#).
- 27 Tomas Jochym-O'Connor, Aleksander Kubica, and Theodore J Yoder. Disjointness of stabilizer codes and limitations on fault-tolerant logical gates. *Physical Review X*, 8(2):021047, 2018.
- 28 Anirudh Krishna and Jean-Pierre Tillich. Towards Low Overhead Magic State Distillation. *Physical Review Letters*, 123(7):070507, August 2019. doi:10.1103/physrevlett.123.070507.
- 29 Tony Metger, Alexander Poremba, Makrand Sinha, and Henry Yuen. Simple constructions of linear-depth t-designs and pseudorandom unitaries. *arXiv preprint arXiv:2404.12647*, 2024. doi:10.48550/arXiv.2404.12647.
- 30 Tefjol Pllaha, Narayanan Rengaswamy, Olav Tirkkonen, and Robert Calderbank. Un-weyl-ing the clifford hierarchy. *Quantum*, 4:370, 2020. doi:10.22331/Q-2020-12-11-370.
- 31 Heydar Radjavi and Peter Rosenthal. *Simultaneous Triangularization*. Springer Science & Business Media, 2000. doi:10.1007/978-1-4612-1200-3.
- 32 Narayanan Rengaswamy, Robert Calderbank, and Henry D Pfister. Unifying the clifford hierarchy via symmetric matrices over rings. *Physical Review A*, 100(2):022304, 2019.
- 33 Peter W Shor. Scheme for reducing decoherence in quantum computer memory. *Physical review A*, 52(4):R2493, 1995.
- 34 P.W. Shor. Fault-tolerant quantum computation. In *Proceedings of 37th Conference on Foundations of Computer Science*, pages 56–65, 1996. doi:10.1109/SFCS.1996.548464.
- 35 Adam Wills, Min-Hsiu Hsieh, and Hayata Yamasaki. Constant-overhead magic state distillation. *arXiv preprint*, 2024. [arXiv:2408.07764](#).
- 36 Bei Zeng, Xie Chen, and Isaac L Chuang. Semi-Clifford operations, structure of C_k hierarchy, and gate complexity for fault-tolerant quantum computation. *Physical Review A*, 77(4):042313, 2008. URL: <https://journals.aps.org/pr/abstract/10.1103/PhysRevA.77.042313>.
- 37 Xinlan Zhou, Debbie W Leung, and Isaac L Chuang. Methodology for quantum logic gate construction. *Physical Review A*, 62(5):052316, 2000.

Quantum Search with Generalized Wildcards

Arjan Cornelissen   

Simons Institute, University of California Berkeley, CA, USA

Nikhil S. Mande   

School of Computer Science and Informatics, University of Liverpool, UK

Subhasree Patro   

Department of Mathematics and Computer Science, Eindhoven University of Technology,
The Netherlands

Nithish Raja   

Department of Mathematics and Computer Science, Eindhoven University of Technology,
The Netherlands

Swagato Sanyal   

School of Computer Science, University of Sheffield, UK

Abstract

In the “search with wildcards” problem [Ambainis, Montanaro, Quantum Inf. Comput.’14], one’s goal is to learn an unknown bit-string $x \in \{-1, 1\}^n$. An algorithm may, at unit cost, test equality of any subset of the hidden string with a string of its choice. Ambainis and Montanaro showed a quantum algorithm of cost $O(\sqrt{n} \log n)$ and a near-matching lower bound of $\Omega(\sqrt{n})$. Belovs [Comput. Comp.’15] subsequently showed a tight $O(\sqrt{n})$ upper bound.

We consider a natural generalization of this problem, parametrized by a subset $\mathcal{Q} \subseteq 2^{[n]}$, where an algorithm may test whether $x_S = b$ for an arbitrary $S \in \mathcal{Q}$ and $b \in \{-1, 1\}^S$ of its choice, at unit cost. We show the following:

- For all $k \in [n]$, when $\mathcal{Q}$ is the collection of all sets of size at most k , the quantum query complexity is $\Theta(n/\sqrt{k})$. In particular when $k = n$, this corresponds to the standard search with wildcards setting. This recovers and generalizes the tight characterization of Belovs, and Ambainis and Montanaro, using completely different techniques.
- When $\mathcal{Q}$ is the collection of contiguous blocks, the quantum query complexity is $\tilde{\Theta}(n)$.
- When $\mathcal{Q}$ is the collection of prefixes, the quantum query complexity is $\Theta(n)$.

All of these results are derived using a framework that we develop. We apply a symmetry reduction to the *primal* version of the negative-weight adversary bound, and show that the quantum query complexity of learning x is characterized, up to a constant factor, by a particular optimization program, which can be succinctly described as follows: ‘maximize over all odd functions $f : \{-1, 1\}^n \rightarrow \mathbb{R}$ the ratio of the maximum value of f to the maximum (over $T \in \mathcal{Q}$) standard deviation of f on a subcube whose free variables are exactly T ’

To the best of our knowledge, ours is the first work to use the primal version of the negative-weight adversary bound (which is a *maximization* program typically used to show lower bounds) to show new quantum query *upper* bounds without explicitly resorting to SDP duality.

2012 ACM Subject Classification Theory of computation → Quantum query complexity

Keywords and phrases quantum algorithms, quantum query complexity, adversary bound, symmetry reduction, substring queries

Digital Object Identifier 10.4230/LIPIcs.TQC.2026.3

Related Version *Full Version*: <https://arxiv.org/abs/2511.04669>

Funding *Arjan Cornelissen*: AC is supported by a Simons-CIQC postdoctoral fellowship through NSF QLCI Grant No. 2016245.

Nikhil S. Mande: NSM gratefully acknowledges support from the London Mathematical Society through a Scheme 7 grant, which enabled preliminary work that sparked this project.

Subhasree Patro: SP acknowledges the support from the Dutch Ministry of Education, Culture, and Science through Gravitation project “Challenges in Cyber Security – 024.006.037” for this work.



© Arjan Cornelissen, Nikhil S. Mande, Subhasree Patro, Nithish Raja, and Swagato Sanyal;
licensed under Creative Commons License CC-BY 4.0

21st Conference on the Theory of Quantum Computation, Communication and Cryptography (TQC 2026).

Editors: Rotem Arnon and Aram W. Harrow; Article No. 3; pp. 3:1–3:20

Leibniz International Proceedings in Informatics



LIPICs Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

Nithish Raja: NR acknowledges the support from the Dutch Ministry of Education, Culture, and Science through Gravitation project “Challenges in Cyber Security – 024.006.037” for this work.

Acknowledgements The authors gratefully acknowledge OpenAI’s ChatGPT (GPT-5), which assisted in developing several crucial ideas presented in this work. The authors bear full responsibility for any mistakes in the work. The authors also thank Nadezhda Voronova for pointing out that one of our lower bounds (when the query set is contiguous blocks) was already known, and for pointing us to the relevant references.

1 Introduction

Reconstructing an unknown string from limited access to its substrings is a natural and widely studied task across many areas of science and engineering. In computational biology, for example, genome assembly and DNA sequencing rely on reconstructing a long unknown strand from local substring measurements (see, for example, [22, 21, 9, 16] and the references therein). Beyond applied motivations, the problem provides a clean and mathematically tractable model for understanding the complexity of learning a hidden string via substring queries.

Most relevant to our work is the *search with wildcards* problem, introduced by Ambainis and Montanaro [1]. In this problem, the goal is to learn a hidden bitstring $x \in \{-1, 1\}^n$ (i.e., an algorithm must correctly output x with high probability) using the minimum number of queries in the following *wildcard query* model. At unit cost, an algorithm may “check its guess for x ” on any subset of the inputs. Formally, at unit cost, an algorithm may make a query (S, b) where $S \subseteq [n]$, $b \in \{-1, 1\}^S$. The query returns 1 if $x_S = b$, and 0 otherwise. Wildcard queries are a generalization of the usual query model, which only allows for $|S| = 1$. Classically even in the wildcard model every query provides at most 1 bit of information, yielding an information-theoretic lower bound of $\Omega(n)$ for learning x . In the usual quantum query model with $|S| = 1$, even though roughly a factor-2 speedup is possible [23], this is essentially the best speedup one can get [13].

Ambainis and Montanaro [1] showed that in the wildcard model, quantum algorithms can learn x using only $O(\sqrt{n} \log n)$ queries, exhibiting roughly a quadratic quantum speedup over the classical lower bound. However, their analysis and algorithm crucially rely on the ability to query *all subsets* of coordinates. In many practical settings, this assumption can be unrealistic. For instance, in *de novo* genome sequencing and related problems in computational biology [21], one has no promise on the input string, and the goal is to reconstruct it. In practical sequencing settings, the accessible measurements are limited to contiguous segments of the underlying strand; one cannot, for instance, simultaneously probe all odd positions at unit cost. Likewise, in more general settings one might be restricted to a fixed family of allowed subsets $\mathcal{Q} \subseteq 2^{[n]}$, and the structure of $\mathcal{Q}$ can significantly affect the power of the algorithm.

In this work, we study the quantum query complexities of learning an unknown string when the allowed query subsets are drawn from an arbitrary fixed collection $\mathcal{Q}$. This unifies previously studied models, including the standard query model $\mathcal{Q} = \{\{i\} : i \in [n]\}$, and the search-with-wildcards model $\mathcal{Q} = 2^{[n]}$ under a single framework.

1.1 Our Results

We refer the reader to Section 2 for formal definitions. We summarize here the notions that we need to state our main results. For an arbitrary collection of sets $\mathcal{Q} \subseteq 2^{[n]}$ and a function $F : \{-1, 1\}^n \rightarrow E$, let $\mathbf{Q}^{\mathcal{Q}}(F)$ denote the quantum query complexity of computing

F , where an algorithm can make unit-cost queries of the form $\mathbb{I}[x_S = b_S]$ for an arbitrary $S \in \mathcal{Q}$ and $b \in \{-1, 1\}^S$ of its choice. We call $\mathcal{Q}^{\mathcal{Q}}(F)$ the *quantum $\mathcal{Q}$ -substring query complexity* of F . Let $\oplus(\cdot)$ denote the parity function, i.e., $\oplus(x) = \prod_{i=1}^n x_i$ for $x \in \{-1, 1\}^n$. A function $f : \{-1, 1\}^n \rightarrow \mathbb{R}$ is said to be *odd* if $f(x) = -f(-x)$ for all $x \in \{-1, 1\}^n$. We write $\text{Var}(f) := \text{Var}_{\mathbf{x} \sim \{-1, 1\}^n}[f(\mathbf{x})]$ for the variance of the random variable $f(\mathbf{x})$, where $\mathbf{x}$ is sampled uniformly at random from $\{-1, 1\}^n$.

Our main technical contribution is to show that the quantum $\mathcal{Q}$ -substring query complexity of parity is characterized by the optimal value of an abstract analytic optimization program. Below, $f_{S|b}$ refers to the function f restricted to S by setting the bits in $\bar{S}$ to b .

► **Theorem 1.** *Let n be a positive integer and $\mathcal{Q} \subseteq 2^{[n]}$. Then $\mathcal{Q}^{\mathcal{Q}}(\oplus) = \Theta(\text{val}_{\mathcal{Q}})$, where*

$$\text{val}_{\mathcal{Q}} := \left(\max_{\substack{f: \{-1, 1\}^n \rightarrow \mathbb{R} \\ f \text{ is an odd function}}} \max_{\substack{S \in \mathcal{Q}, \\ b \in \{-1, 1\}^{\bar{S}}}} \frac{\|f\|_{\infty}}{\sqrt{\text{Var}(f_{S|b})}} \right). \quad (1)$$

In order to show this, we use the characterization of quantum query complexity by the negative-weight adversary bound [15, 20]. We show that for all $\mathcal{Q}$, the corresponding adversary bound *equals* $\text{val}_{\mathcal{Q}}$ as defined above. We next analyze $\text{val}_{\mathcal{Q}}$, as defined in Equation (1) for various interesting and natural settings of $\mathcal{Q}$ such as bounded-size sets, contiguous blocks, prefixes, and only the full set.

► **Theorem 2.** *Let n be a positive integer. Then, the following bounds hold for $\text{val}_{\mathcal{Q}}$ for various classes of $\mathcal{Q} \subseteq 2^{[n]}$.*

Query Set	$\mathcal{Q}$	$\text{val}_{\mathcal{Q}}$	Reference
Bounded-size sets	$\{S \subseteq [n] : S \leq k\}$	$\Theta\left(\frac{n}{\sqrt{k}}\right)$	Theorem 16
Contiguous blocks	$\{S = [i, j] : i \leq j \in [n]\}$	$\tilde{\Theta}(n)$	Theorem 18
Prefixes	$\{[1, i] : i \in [n]\}$	$\Theta(n)$	Full version of the paper
Only full set	$\{[n]\}$	$2^{(n-1)/2}$	Full version of the paper

These results have immediate implications for the query complexity of the string-recovery problem in these query models. Let $\text{rec} : \{-1, 1\}^n \rightarrow \{-1, 1\}^n$ be the (string recovery/learning) function defined by $\text{rec}(x) = x$. Using Theorem 1, Theorem 2, and the Bernstein-Vazirani algorithm [6] to recover a bitstring in one query given query access to arbitrary parities of it, we obtain the following corollary.¹

► **Corollary 3.** *Let n be a positive integer. Then, the quantum $\mathcal{Q}$ -substring query complexity of learning an input string is characterized up to constants by $\text{val}_{\mathcal{Q}}$.*

In particular, setting $k = n$ in the first bound above tightens the $O(\sqrt{n} \log n)$ algorithm of Ambainis and Montanaro [1] for quantum search with wildcards, and improves it to the optimal $O(\sqrt{n})$. We note that the optimal $O(\sqrt{n})$ bound in this case has already been shown by Belovs [3] using completely different techniques from ours. Additionally, our results in the prefix-query model prove that recovering any bitstring cannot be done with a sublinear

¹ The upper bounds for contiguous and prefix queries follow from the naive upper bound of querying one bit at a time (or an increasingly long prefix in the prefix query model) to learn an input string in $O(n)$ queries. The upper bound for full set queries follows from [14].

number of prefix queries, which resolves an open question raised in [24]. The result in the only full set case recovers the well-known tight $\Omega(\sqrt{N})$ lower bound for searching for a marked element in a database with N items [7] and the matching upper bound is given by [14].

A lower bound of $\Omega(n/\log(n))$, for learning the entire string, was shown by [11] and [8] when the query set is restricted to contiguous blocks. We recover this lower bound using our framework.

1.2 Our Techniques

We now discuss the techniques that we use to show Theorems 1 and 2. We then make a comparison with related work.

1.2.1 Sketch of Proof of Theorem 1

Our analysis begins from the characterization of quantum query complexity by the negative-weight adversary bound [20]. This characterization considers the quantum query complexity of a function $F : D \rightarrow E$, with respect to a query set $\mathcal{R}$, where every query $q \in \mathcal{R}$ is a function mapping D to a set of query outcomes. Reichardt [20] expresses the query complexity of F w.r.t. $\mathcal{R}$, up to a constant factor, in terms of the optimum value of a certain semidefinite optimization program known as the general adversary bound, which we refer to as the *primal adversary bound*, defined as

$$\text{ADV}^{\mathcal{R},\pm}(F) := \max_{\Gamma \neq 0} \frac{\|\Gamma\|}{\max_{q \in \mathcal{R}} \|\Gamma_q\|}. \quad (2)$$

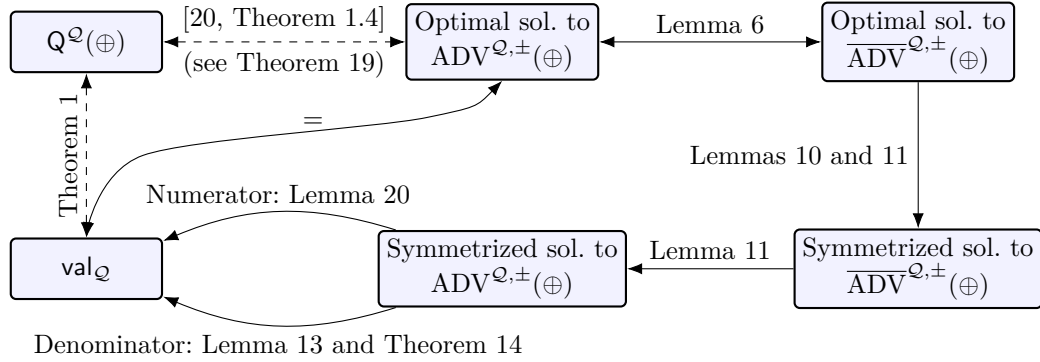
Here the maximization is over all matrices $\Gamma \in \mathbb{R}^{D \times D}$ with $F(x) = F(y) \implies \Gamma[x, y] = 0$, and the maximization in the denominator is over all available queries $\mathcal{R}$. For a query $q \in \mathcal{R}$, the matrix Γ_q is defined to be equal to Γ , with all those $[x, y]$ entries zeroed out where the outcome of the query q is the same on inputs x and y . This quantity was defined in [15], where it was shown to be a lower bound on quantum query complexity. Reichardt [20] subsequently showed that the query complexity of F with respect to query set $\mathcal{R}$ is equal to $\text{ADV}^{\mathcal{R},\pm}(F)$ up to a constant factor.

It is well-known (see e.g., [19, Theorem 1.2]) that the negative-weight adversary bound can be written as a semidefinite program (SDP). We show in Lemma 6 that the feasible solutions of the maximization version of this SDP are directly related to the choices of Γ in Equation (2). It then follows that symmetry reductions on the SDP-level can be translated into additional symmetry constraints on the matrix Γ in Equation (2), without changing the optimal objective value. Note that attempting to naively apply such a symmetrization to the formulation in Equation (2) does not immediately work since, for example, the numerator could potentially reduce after symmetrizing. This is why all our symmetrization is done at the SDP level.

For the specific case where $F = \oplus$ and $\mathcal{R} = \{(S, b) : S \in \mathcal{Q}, b \in \{-1, 1\}^S\}$, with $\mathcal{Q} \subseteq 2^{[n]}$, we subsequently characterize in Lemmas 10 and 11 the symmetries in the primal adversary bound SDP, from which we deduce in Corollary 12 that a maximizing Γ can be assumed to be the communication matrix of an XOR function. That is, we can assume that $\Gamma = M_{f \circ \text{XOR}}$ for some $f : \{-1, 1\}^n \rightarrow \mathbb{R}$, where $M_{f \circ \text{XOR}}(x, y) := f(x \oplus y)$, i.e., the (x, y) 'th entry only depends on the bitwise XOR of x and y . It is well known that such matrices are simultaneously diagonalizable by the Hadamard matrix, and hence diagonal in the Fourier basis, which motivates analyzing the numerator and denominators of Equation (2) using Fourier analysis.

It is known that the numerator of Equation (2) in this case, which is the spectral norm of $M_{f \circ \text{XOR}}$, equals the (appropriately scaled) maximum Fourier coefficient of f [5]. For the denominator, we analyze the sparsity structure of the matrix Γ_q , for any query $(S, b) = q \in \mathcal{R}$. Through a series of calculations involving Fourier analysis of restricted Boolean functions, we show that the matrix Γ_q 's spectral norm equals the (appropriately scaled) maximum, over all fixings of bits in $\bar{S}$, standard deviation of the restriction of f to the subcube that fixes these bits in $\bar{S}$. Putting everything together, we obtain $\text{val}_{\mathcal{Q}} = \text{ADV}^{\mathcal{R}, \pm}(\oplus)$, which together with Reichardt's characterization [20] implies Theorem 1.

We provide a schematic illustration of the proof overview for Theorem 1 in Figure 1.



■ **Figure 1** A schematic illustration to obtain our framework, which enables us to transform the problem of computing $Q^{\mathcal{Q}}(\oplus)$, for an arbitrary collection of sets $\mathcal{Q} \subseteq 2^{[n]}$, into the task of analyzing an abstract analytic optimization program $\text{val}_{\mathcal{Q}}$. The solid lines represent exact equalities, whereas the dashed lines represents characterizations up to constants.

1.2.2 Sketch of Proof of Theorem 2

Theorem 2 proves matching lower and upper bounds on $\text{val}_{\mathcal{Q}}$ for four different classes of sets $\mathcal{Q}$. Note that $\text{val}_{\mathcal{Q}}$ involves a maximization over all odd real-valued functions over the Boolean hypercube. The lower bounds are proven by exhibiting explicit functions for which every valid (for the class of $\mathcal{Q}$ considered) subcube restriction has low standard deviation with respect to the maximum absolute value of the function. The upper bounds are proven by showing that every odd real-valued function over the Boolean hypercube has a high standard deviation (with respect to its maximum absolute value) when restricted to some subcube that is valid for the class of $\mathcal{Q}$ considered. We now briefly discuss the proofs we give for various classes of $\mathcal{Q}$. In the main version of the paper we also discuss the “prefixes” and “only full set” classes, which are omitted in this version.

1.2.2.1 Bounded size

Theorem 16 deals with query sets of bounded size. The valid subcubes are the ones that fix at least $n - k$ variables. The lower bound is witnessed by the function that maps each bit-string to its Hamming weight minus $n/2$ (note that this is an odd function). For the upper bound, we first prove by an induction on the number of variables, that for every odd function there is some subcube (not necessarily valid) which simultaneously has a high variance and the absolute values of all its degree-1 Fourier coefficients are large. If this subcube is not valid (i.e., if it fixes less than $n - k$ variables), we argue via the probabilistic method and

Fourier analysis that the function can be further restricted by appropriately setting additional variables to a valid subcube, while simultaneously ensuring that the variance does not drop too much.

1.2.2.2 Contiguous blocks

Theorem 18 deals with query sets comprising consecutive variables (the same theorem statement holds true even if one is allowed wraparound). The valid subcubes are the ones that restrict consecutive variables. The upper bound of n follows from the observation that the query sets include all singletons, and Proposition 15 which verifies via the framework of Theorem 1 the obvious fact that if all the single bit queries are allowed, then the quantum query complexity is $O(n)$.

The lower bound is witnessed by a function f defined as follows. Given an input bit string of size n , view it as consisting of n/k disjoint blocks of size k , where $k = \log n - \log \log n + O(1)$. Let the blocks be denoted by $z^{(1)}, \dots, z^{(n/k)}$. First, each $z^{(i)}$ is subjected to a function g that maps the string 1^k to 1, $(-1)^k$ to -1 and maps all other strings to 0. Finally f is defined to be $\sum_{i=1}^{n/k} g(z^{(i)})$. Note that each subcube setting contiguous variables (even if wraparound is allowed) touches, i.e., the endpoints of the block of queried indices are contained in, at most two of the blocks, so the restricted function is essentially a reduced version of f on a smaller number of blocks plus a bounded function on at most $2k$ bits. Thus, up to an additive shift by a bounded function, the (random variable corresponding to the) restriction of f is essentially the sum of independent random variables taking values in $\{0, 1, -1\}$ (the independence is due to the disjointness of the blocks), each of which is 0 except with probability $1/2^{k-1}$. Furthermore, the additional bounded summand is also independent of this sum and has $O(1)$ variance. We can thus apply linearity of variance for independent random variables to finish our argument.

1.3 Comparison With Related Work

The quantum $\mathcal{Q}$ -substring query complexity of rec with $\mathcal{Q} = 2^{[n]}$ was introduced and studied by Ambainis and Montanaro [1], who dubbed this problem the “search with wildcards” problem and gave an $O(\sqrt{n} \log n)$ upper bound and a near matching $\Omega(\sqrt{n})$ lower bound. The algorithm by Ambainis and Montanaro uses the Pretty Good Measurement to iteratively learn $O(\sqrt{n})$ bits and fixes any errors made via binary search. In the search with wildcards setting, binary search can be performed with just $O(\log n)$ queries. Therefore, the algorithm learns the entire string in $O(\sqrt{n} \log n)$ queries. Setting $k = n$ in our Corollary 3 gives a tight $O(\sqrt{n})$ bound for this task. Ambainis and Montanaro also gave quantum query algorithms for the combinatorial group testing problem, which they observed to be a generalization of the $\mathcal{Q}$ -substring query complexity of rec with $\mathcal{Q} = 2^{[n]}$. Subsequently, Belovs [3] gave, among other results, tight bounds on the quantum query complexity of combinatorial group testing, which implies a tight $O(\sqrt{n})$ upper bound on the quantum $2^{[n]}$ -substring query complexity of rec . In contrast with the algorithms of Ambainis and Montanaro, Belovs showed a $O(\sqrt{n})$ upper bound on the dual of the negative-weight adversary bound. Since the negative-weight adversary bound is known to be asymptotically equal to quantum query complexity [20], this immediately gives an upper bound on $\mathcal{Q}^{\mathcal{Q}}(\text{rec})$.

In contrast to the work of Ambainis and Montanaro [1] who construct an explicit algorithm, and that of Belovs [3] who constructs an explicit solution to the dual adversary bound, we deal, perhaps counter-intuitively, completely with the *primal* version of the negative-weight adversary bound [15]. Our work does have the drawback that it does not give explicit

algorithms for any of our upper bounds. However, as far as we are aware, ours is the first work to prove novel quantum query *upper* bounds using the *primal* version of the negative-weight adversary bound (which is primarily used as a lower bound technique) without explicitly resorting to SDP duality.

When $\mathcal{Q}$ is the set of all contiguous blocks, a lower bound of $\Omega(n/\log(n))$ was previously shown by [11, Lemma 7] via the positive-weight adversary method. The same lower bound, even for computing Parity, was also shown by [8, Theorem 19], where they lower bound the approximate degree and obtain the lower bound via the polynomial method [2]. Furthermore, their lower bound holds even in the setting where the success probability of the algorithm is inverse polynomially close to $1/2$. In this work, we obtain the same $\Omega(n/\log(n))$ query lower bound (in the bounded-error setting) using our framework, using techniques different from the previous works.

Another work of relevance to ours is [4], who showed classical lower bounds for composed functions using quantum upper bounds for combinatorial group testing [3]. It is not immediately clear to us if our results have any new consequences for classical lower bounds using their framework, but our work can be viewed as partial progress towards answering their question [4, Section V] “Are there nontrivial quantum algorithms that use g -queries to learn x for some function $g \notin \{\text{OR}, \text{XOR}\}$? Are there nontrivial quantum algorithms that use g -queries to compute some other function $h(x)$ of the input?” Theorem 1 gives a complete framework that allows one to easily analyze the complexity of computing Parity (which is often equivalent to the complexity of learning the full input string) when g is restricted to subcube queries. Our framework gives provably tight bounds in this setting for all classes of allowed subcubes to query (with the constraint that if the algorithm is allowed to query if x is in a subcube whose fixed variables are S , all fixing of variables in S are allowed queries). The key advantage of our framework is that it replaces quantum and SDP-based reasoning with a simple, combinatorial analysis of real-valued functions on the Boolean hypercube. We justify this simpler viewpoint via Theorem 2, which establishes tight bounds within this framework for several natural classes of allowed subcube query sets.

1.4 Outlook

While our results do not apply in the combinatorial group testing setting, where only OR queries are allowed and the input string is promised to have bounded Hamming weight, we consider a generalization of the search with wildcards problem in a different direction, namely when a query algorithm is restricted on the support of the substring queries it can make. Our results are specifically of interest in the context of biological applications such as genome reconstruction, DNA sequencing, and RNA sequencing: In these models, a unit-cost task is often assumed to be observing a contiguous substring of the input string [22, 21, 9, 16]. Our second bound in Corollary 3 shows that quantum algorithms cannot provide an advantage over classical ones in this setting.

We view our work as an invitation to explore upper bounds through the *primal* adversary formulation itself, by studying the structural properties that optimal adversary matrices must satisfy. While this formulation can tend to be regarded as less convenient than its dual, our results illustrate that, with sufficient symmetrization and structural insight, the primal characterization can in fact yield tight bounds and reveal new patterns in how optimal Γ matrices arise. This suggests that it may be a more practical tool for upper bounds than previously thought.

2 Preliminaries

All logarithms in this paper are taken base 2. For a positive integer n and integers $i \leq j \in [n]$, we use the notation $[i, j] := \{k \in [n] : i \leq k \leq j\}$. For strings, $x, y \in \{-1, 1\}^n$, we use the notation $x \oplus y$ to denote the bitwise XOR of x and y , or equivalently, the string obtained by taking component-wise multiplication of x and y . For a string $x \in \{-1, 1\}^n$, we use the notation $|x|$ to denote the Hamming weight of x , i.e., the number of -1 's in x . For a string $x \in \{-1, 1\}^n$ and $S \subseteq [n]$, we use x_S to denote the string x restricted to the bits in S . For an expression X , we define $\mathbb{I}[X]$ to equal 1 if X is true, and 0 if X is false. For a string $x = (x_1, x_2, \dots, x_n) \in \{-1, 1\}^n$, we use the notation $-x$ to denote the string $(-x_1, -x_2, \dots, -x_n)$. We denote vectors with the ket-notation, $|v\rangle$, and refer to their conjugate transpose with the bra-notation, i.e., $\langle v|$.

A function $f : \{-1, 1\}^n \rightarrow \mathbb{R}$ is said to be an *odd function* if $f(x) = -f(-x)$ for all $x \in \{-1, 1\}^n$. We abuse notation and use f to interchangeably mean a function as well as a random variable corresponding to picking a value of f uniformly at random. That is, for $f : \{-1, 1\}^n \rightarrow \mathbb{R}$, we use $\mathbb{E}[f] := \mathbb{E}_{x \in \{-1, 1\}^n}[f(x)]$, $\mathbb{E}[f^2] := \mathbb{E}_{x \in \{-1, 1\}^n}[f(x)^2]$, and $\text{Var}(f) := \mathbb{E}[f^2] - \mathbb{E}[f]^2$. We use $\|f\|_\infty$ to denote $\max_{x \in \{-1, 1\}^n} |f(x)|$. For a positive integer n and a 2^n -dimensional real matrix M whose rows and columns are indexed by n -bit strings, we say that a M is the *communication matrix of an XOR function* if there exists a function $f : \{-1, 1\}^n \rightarrow \mathbb{R}$ such that $M(x, y) = f(x \oplus y)$. We use the notation $M_{f \circ \text{XOR}}$ to denote the communication matrix of $f \circ \text{XOR}$.

A collection of sets $\mathcal{Q} \subseteq 2^{[n]}$ is said to be *downward closed* if for all $Q \in \mathcal{Q}$, all subsets of Q are also in $\mathcal{Q}$. For an n -dimensional square matrix M , we denote its spectral norm by $\|M\|$. We provide a brief overview of quantum algorithms and their query models in Appendix A, and we state the results from Fourier analysis that we need in Appendix B.

3 The Framework

In this section we build our framework that allows us to analyze the $\mathcal{Q}$ -substring query complexity of learning input strings, for arbitrary $\mathcal{Q}$.

3.1 Learning vs. Parity

We first make the simple observation that for every collection of sets $\mathcal{Q} \subseteq 2^{[n]}$, the quantum query complexity of learning x is bounded from below by the quantum query complexity of computing the parity of all of the bits of x .

► **Observation 4.** *Let $\mathcal{Q} \subseteq 2^{[n]}$. Then $Q^{\mathcal{Q}}(\text{rec}) \geq Q^{\mathcal{Q}}(\oplus)$.*

This immediately follows from the fact that once a string has been learned, the parity of its input bits can be computed with no extra queries. We remark that the same argument in fact yields $Q^{\mathcal{Q}}(\text{rec}) \geq \max_{S \subseteq [n]} Q^{\mathcal{Q}}(\oplus_S)$, but we do not use this.

Next we observe that in the case when $\mathcal{Q}$ is downward closed, these complexities are actually equal.

► **Lemma 5.** *For all $\mathcal{Q} \subseteq 2^{[n]}$ that is downward closed, we have $Q^{\mathcal{Q}}(\text{rec}) = Q^{\mathcal{Q}}(\oplus)$.*

Proof. Observation 4 provides the lower bound on $Q^{\mathcal{Q}}(\text{rec})$, so it remains to prove the upper bound. Note that we can recover a bit string if we are able to compute the parity on each of the subsets in superposition, by the Bernstein-Vazirani algorithm. Thus, it remains to prove that $Q^{\mathcal{Q}}(\oplus_S) \leq Q^{\mathcal{Q}}(\oplus)$, for all $S \subseteq [n]$.

To that end, let $S \subseteq [n]$. For any $x \in \{-1, 1\}^n$, let $y(x) \in \{-1, 1\}^n$ be the input that is equal to x on S , and identically 1 on $\bar{S}$. Then, $\oplus_S(x) = \oplus(y(x))$, so it remains to run the algorithm on $y(x)$. To that end, note that we can simulate a substring query (T, b) to $y(x)$ as follows. First, we check if b is identically 1 on $T \setminus S$, else we simply output 1 (False). Otherwise, we perform the query $(S \cap T, b_S)$ on x , and we observe that $y(x)_T = b$ if and only if $y(x)_{S \cap T} = b_S$. Since $\mathcal{Q}$ is downward-closed and $S \cap T \subseteq T$, we indeed have access to this query, and so we can simulate the parity-algorithm on $y(x)$ without additional overhead. $\blacktriangleleft$

We conclude that in order to characterize the $\mathcal{Q}$ -complexity of bit string recovery, it suffices to characterize the $\mathcal{Q}$ -complexity of computing the parity function, as long as $\mathcal{Q}$ is downward closed.

3.2 Simplification by Symmetry Reduction

In this subsection we perform a symmetry reduction on the primal adversary bound for the quantum $\mathcal{Q}$ -substring quantum query complexity of computing parity, for an arbitrary $\mathcal{Q} \subseteq 2^{[n]}$. We show that the maximizing matrix Γ can be assumed to be the communication matrix of a XOR function, i.e., we may assume that $\Gamma = M_{f \circ \text{XOR}}$ for some $f : \{-1, 1\}^n \rightarrow \mathbb{R}$. Recall that $M_{f \circ \text{XOR}}[x, y] = f(x \oplus y)$ for all $x, y \in \{-1, 1\}^n$.

3.2.1 Alternate formulation of adversary bound for functions with Boolean output

Here we use an alternative formulation of the adversary bound that was shown by Cornelissen [12, Theorem 6.2.4]. This formulation allows us to average over optimal solutions without any loss in the optimal value. An important detail to note is that this formulation holds only for functions with Boolean output.²

Alternate formulation of primal adversary bound for functions with Boolean output

$$\begin{aligned} \overline{\text{ADV}}^{\mathcal{R}, \pm}(F) = & \max_{\Gamma, \beta} \sum_{x, y \in D} \Gamma[x, y] & (3) \\ \text{subject to } & \Gamma \in \mathbb{R}^{D \times D}, \beta \in \mathbb{R}_{\geq 0}^D, \\ & \Gamma \text{ is symmetric,} \\ & \Gamma[x, y] = 0, & (\forall x, y \in D \text{ with } F(x) = F(y)) \\ & \text{diag}(\beta) - \Gamma \circ \Delta_q \succeq 0, & (\forall q \in \mathcal{R}) \\ & \sum_{x \in f^{-1}(1)} \beta[x] = \sum_{x \in f^{-1}(-1)} \beta[x] = \frac{1}{2}. \end{aligned}$$

► **Lemma 6.** *Let $D \subseteq \{-1, 1\}^n$ and $F : D \rightarrow \{-1, 1\}$ be a Boolean function. Then $\text{ADV}^{\mathcal{R}, \pm}(F) = \overline{\text{ADV}}^{\mathcal{R}, \pm}(F)$. Moreover, if (β, Γ) is an optimal solution for $\overline{\text{ADV}}^{\mathcal{R}, \pm}(F)$, then Γ' , defined as³*

$$\Gamma'[x, y] = \frac{\Gamma[x, y]}{\sqrt{\beta[x]\beta[y]}},$$

is an optimal solution for $\text{ADV}^{\mathcal{R}, \pm}(F)$.

² A similar optimization program can be derived for functions with non-Boolean output. Since its presentation is slightly more cumbersome and we don't need it in this work, we merely present the Boolean version here.

³ Here, we use the convention that $0/0 = 0$.

The proof of this lemma essentially appears in the proof of [12, Theorem 6.2.4], but not with the exact formulation above. We provide a proof of Lemma 6 in the main version of the paper.

The benefit of this alternate formulation of the primal adversary bound is that it is a convex optimization problem. Therefore, it is always possible to take a linear combination of two feasible solutions, say with objective values c_1 and c_2 , and obtain a new feasible solution with an objective value that is at least as big as $\min\{c_1, c_2\}$. We will use this to perform a symmetry reduction on the level of the SDP. In particular, we show how symmetries in the function and in the query set translate to additional constraints that we can impose on the optimal solutions to the alternate formulation of the primal adversary bound.

Let $\sigma : D \rightarrow D$ be a permutation of the domain D . We abuse notation to use σ to refer to the permutation function as well as the corresponding permutation matrix. It will be clear based on context whether σ is treated as a function or a matrix. When treated as a matrix, right-multiplying by σ^T corresponds to permuting columns according to the permutation σ and left-multiplying by σ corresponds to permuting rows according to σ .

Next, we define two types of symmetry that our query problem could exhibit.

► **Definition 7** (Function symmetry). *A permutation $\sigma : D \rightarrow D$ is said to be function symmetric with respect to a function $F : D \rightarrow E$ if $F(\sigma(x)) = F(\sigma(y)) \iff F(x) = F(y)$, $\forall x, y \in D$.*

► **Definition 8** (Query symmetry). *A permutation $\sigma : D \rightarrow D$ is said to be query symmetric with respect to the set of queries $\mathcal{R}$ if $\{\sigma \Delta_q \sigma^T \mid q \in \mathcal{R}\} = \{\Delta_q \mid q \in \mathcal{R}\}$.*

We now define a group that we use for the rest of this section and in the appendices.

► **Definition 9.** *Let G denote the group generated by bit-flip permutations, i.e., $G = \{\sigma_y \mid y \in \{-1, 1\}^n\}$ where $\sigma_y : x \mapsto x \oplus y$.*

We now show that computing the parity function using $\mathcal{Q}$ -substring queries exhibits these symmetries. The proofs of the following lemmas can be found in the full version.

► **Lemma 10.** *Let $n \in \mathbb{N}$ and $\mathcal{Q} \subseteq 2^{[n]}$. Then, every $\sigma \in G$ is function-symmetric w.r.t. $\oplus$ and query-symmetric w.r.t. $\mathcal{Q}$.*

► **Lemma 11.** *Let n be a positive integer and let $\mathcal{R}$ be a query set. Let G be the group generated by bit flip permutations. If every permutation $\sigma \in G$ is query symmetric with respect to $\mathcal{R}$, then there exists an $f : \{-1, 1\}^n \rightarrow \mathbb{R}$ such that*

- $f(x) = 0$ for all x of even Hamming weight, and
- $M_{f \circ \text{XOR}}$ is an optimal solution for the adversary bound $\text{ADV}^{\mathcal{R}, \pm}(\oplus)$.

► **Corollary 12.** *Let $n \in \mathbb{N}$ and $\mathcal{Q} \subseteq 2^{[n]}$. Then*

$$\text{ADV}^{\mathcal{Q}, \pm}(\oplus) = \max_{\substack{f: \{-1, 1\}^n \rightarrow \mathbb{R} \\ |z| \text{ even} \implies f(z) = 0}} \frac{\|M_{f \circ \text{XOR}}\|}{\max_{\substack{S \in \mathcal{Q}, \\ b \in \{-1, 1\}^S}} \|M_{f \circ \text{XOR}} \circ \Delta_{S, b}\|}.$$

Proof. We first show that $M_{f \circ \text{XOR}}$ is a feasible solution to $\text{ADV}^{\mathcal{Q}, \pm}(\oplus)$, for all $f : \{-1, 1\}^n \rightarrow \mathbb{R}$ satisfying $f(x) = 0$ when $|x|$ is even. The matrix $M_{f \circ \text{XOR}}$ is symmetric since $M[x, y] = f(x \oplus y) = f(y \oplus x) = M[y, x]$ for all $x, y \in \{-1, 1\}^n$. We know that $|x \oplus y|$ is even if both $|x|, |y|$ are even or if both $|x|, |y|$ are odd. Therefore,

$$\oplus(x) = \oplus(y) \implies |x \oplus y| \text{ is even} \implies M_{f \circ \text{XOR}}[x, y] = 0.$$

Lemma 11 (which is applicable since Lemma 10 guarantees all elements of G to be query symmetric w.r.t. $\mathcal{Q}$ and function symmetric w.r.t. $\oplus$) yields the corollary. ◀

3.3 Simplification by Fourier Analysis

In this subsection we show that using the structure we derived in the last subsection, we can simplify the expression on the right-hand side of Corollary 12 to a concise analytic optimization problem over the Boolean hypercube. Lemma 20 says that the numerator equals $2^n \max_{S \subseteq [n]} |\widehat{f}(S)|$. For the denominator, we use the following lemma, whose proof is available in the main version of the paper.

► **Lemma 13.** *For all $f : \{-1, 1\}^n \rightarrow \mathbb{R}$, and all $S \subseteq [n]$ and $b \in \{-1, 1\}^S$, we have*

$$\|M_{f \circ \text{XOR}} \circ \Delta_{S,b}\| = 2^{n-|S|} \max_{x \in \{-1, 1\}^{\bar{S}}} \left\| \left(\widehat{f_{\bar{S}|y \oplus b}}(x) \right)_{y \in \{-1, 1\}^S \setminus \{b\}} \right\|_2.$$

Now, we can prove the required bound on the operator norm of the denominator in Corollary 12.

► **Theorem 14.** *For all $f : \{-1, 1\}^n \rightarrow \mathbb{R}$, $S \subseteq [n]$ and all $b \in \{-1, 1\}^S$, we have*

$$\|M_{f \circ \text{XOR}} \circ \Delta_{S,b}\| = 2^n \max_{x \in \{-1, 1\}^{\bar{S}}} \sqrt{\text{Var}(\hat{f}_{S|x})}.$$

Proof. For $x \in \{-1, 1\}^S$ and $y \in \{-1, 1\}^{\bar{S}}$, let xy denote the bit string that equals x on S and y on $\bar{S}$. For the rest of this proof we identify a bitstring with its characteristic set. Specifically, this means for every $f : \{-1, 1\}^n \rightarrow \mathbb{R}$, we use the notation $\hat{f}(x) := \hat{f}(\{j \in [n] : x_j = -1\})$. The spectral norm of a symmetric matrix with a single non-zero row is simply the norm of that row. We can thus use Lemma 13 to obtain that

$$\begin{aligned} \|M_{f \circ \text{XOR}} \circ \Delta_{S,b}\| &= 2^{n-|S|} \max_{x \in \{-1, 1\}^{\bar{S}}} \left\| \left(\widehat{f_{\bar{S}|y \oplus b}}(x) \right)_{y \in \{-1, 1\}^S \setminus \{b\}} \right\|_2 \\ &= 2^{n-|S|} \max_{x \in \{-1, 1\}^{\bar{S}}} \sqrt{\sum_{y \in \{-1, 1\}^S \setminus \{b\}} \widehat{f_{\bar{S}|y \oplus b}}(x)^2} \\ &= 2^{n-|S|} \max_{x \in \{-1, 1\}^{\bar{S}}} \sqrt{\sum_{y \in \{-1, 1\}^S \setminus \{1^S\}} \widehat{f_{\bar{S}|y}}(x)^2} \quad \text{by variable substitution} \\ &= 2^{n-|S|} \max_{x \in \{-1, 1\}^{\bar{S}}} \sqrt{\sum_{y \in \{-1, 1\}^S} \widehat{f_{\bar{S}|y}}(x)^2 - \widehat{f_{\bar{S}|1^S}}(x)^2} \\ &= 2^{n-|S|} \max_{x \in \{-1, 1\}^{\bar{S}}} \sqrt{2^{|S|} \mathbb{E}_{y \in \{-1, 1\}^S} [\widehat{f_{\bar{S}|y}}(x)^2] - \widehat{f_{\bar{S}|1^S}}(x)^2} \\ &= 2^{n-|S|} \max_{x \in \{-1, 1\}^{\bar{S}}} \sqrt{2^{|S|} \sum_{y \in \{-1, 1\}^S} \hat{f}(xy)^2 - \left(\sum_{y \in \{-1, 1\}^S} \hat{f}(xy) \right)^2} \\ &\quad \text{By Proposition 22 and Corollary 23} \\ &= 2^{n-|S|} \max_{x \in \{-1, 1\}^{\bar{S}}} \sqrt{2^{2|S|} \mathbb{E}[\hat{f}_{S|x}^2] - 2^{2|S|} \mathbb{E}[\hat{f}_{S|x}]^2} \\ &= 2^n \max_{x \in \{-1, 1\}^{\bar{S}}} \sqrt{\text{Var}(\hat{f}_{S|x})}. \quad \blacktriangleleft \end{aligned}$$

Observe that the RHS in the theorem above is independent of b . Using this derived expression for the denominator, we are able to now state and prove Theorem 1. We refer to the main version of the paper for the formal version of the proof.

4 Applications of the Framework

In this section we use the framework developed in the last section to analyze val_Q , and thereby the quantum Q -substring query complexity of computing parity, for various interesting collections Q . We consider a wider variety of collections Q in the main version of the paper.

4.1 Contains All Singletons

In the case that the allowed substring queries include all singletons, there is an easy quantum query upper bound of n : simply query all of the variables one at a time. As a sanity check, we first show how this can be recovered by our framework.

► **Proposition 15.** *Let $Q \supseteq \{\{i\} : i \in [n]\}$. Then $\text{val}_Q \leq n$.*

Proof. Since f is odd, there is always an x^* such that $f(x^*) = \|f\|_\infty > 0$. Let $M = \max_x f(x)$ be attained at x^* . As f is odd, $f(-x^*) = -M$.

Consider a shortest path from x^* to $-x^*$ in the hypercube, this has length n . Along this path the total value of f decreases by $2M$, thus there must exist an edge with a drop of at least $2M/n$. That edge corresponds to flipping one coordinate, say $i \in [n]$. This means there exists an $z \in \{-1, 1\}^n$ such that $|f(z) - f(z \oplus e_i)| \geq 2M/n$. Choose the subcube that fixes all variables but i to be consistent with z , and leave the i 'th variable free. Note that this is a valid subcube to consider as Q contains all singletons. For this choice of subcube we have the variance (using the formula $\text{Var}(g) = \frac{1}{2}\mathbb{E}[(g(x) - g(y))^2]$) to be $\frac{(f(z) - f(z \oplus e_i))^2}{4} \geq \frac{4M^2}{4n^2}$. Thus $\text{val}_Q \leq \frac{M}{\sqrt{4M^2/4n^2}} = n$. ◀

4.2 Sets of Bounded Size

The setting $Q = \{S \subseteq 2^{[n]} : |S| \leq k\}$ corresponds to the model where the allowed substrings to be queried are bounded in size.

In particular, $k = n$ corresponds to the usual search with wildcards setting, where an algorithm is allowed to query OR's of arbitrary sets of literals.

► **Theorem 16.** *Let $1 \leq k \leq n$ be an integer, and let $Q = \{S \subseteq 2^{[n]} : |S| \leq k\}$. Then,*

$$\frac{n}{\sqrt{k}} \leq \text{val}_Q \leq \frac{2n}{\sqrt{k}}.$$

We show the lower bound using the (appropriately shifted) Hamming weight function. We prove the upper bound by induction on n . We first state a convenient induction hypothesis, then show how it implies the upper bound, and finally prove the induction hypothesis.

► **Lemma 17.** *For all integers $n > 0$ and all functions $f : \{-1, 1\}^n \rightarrow \mathbb{R}$ with $\mathbb{E}[f(x)] = 0$, there exists a $S \subseteq [n]$ and $b \in \{-1, 1\}^{\bar{S}}$ where:*

- $\text{Var}(f_{S|b}) \geq \frac{\|f\|_\infty^2}{4n}$, and
- $\left| \widehat{f_{S|b}}(\{i\}) \right| \geq \frac{\|f\|_\infty}{2n}$ for all $i \in S$.

In other words, there exists a subcube where the restricted function has large variance, as well as large degree-1 Fourier coefficients. We feel this is an independently interesting result in Fourier analysis. We first show how Theorem 16 follows from Lemma 17 (note that Lemma 17 gives no guarantee on $|S|$, which we require for proving Theorem 16).

Proof of Theorem 16. We first show the lower bound on $\text{val}_{\mathcal{Q}}$, and then the upper bound.

- **Lower bound:** Consider the function $f : \{-1, 1\}^n \rightarrow \mathbb{R}$ defined by $f(x) = |x| - (n/2)$. It is clear that f is an odd function since $f(-x) = |-x| - (n/2) = (n - |x|) - (n/2) = n/2 - |x| = -f(x)$. Let $m \leq k$. For any subcube that fixes all but a collection of m variables, it is easy to see that the random variable obtained by restricting f to this subcube and picking a uniformly random function value in the restricted subcube is simply a constant additive shift of the binomial random variable $\text{Bin}(m, 1/2)$. Since such shifts do not affect variance, the variance in this subcube equals $m/4 \leq k/4$, since the only subcubes under consideration are those where the number of free variables is at most k . Since $\max |f(x)| = n/2$, this function f witnesses

$$\text{val}_{\mathcal{Q}} \geq \frac{n/2}{\sqrt{k/4}} = \frac{n}{\sqrt{k}}.$$

- **Upper bound:** Let $M := \|f\|_{\infty} = \max_{x \in \{-1, 1\}^n} |f(x)|$. Let $S \subseteq [n]$ and $b \in \{-1, 1\}^{\bar{S}}$ be obtained from Lemma 17 (which is applicable since f is odd, and hence balanced). If $|S| \leq k$, then this is a valid subcube that can be chosen in the denominator of $\text{val}_{\mathcal{Q}}$, and we get $\text{val}_{\mathcal{Q}} \leq \frac{M}{\sqrt{M^2/4n}} = 2\sqrt{n} \leq \frac{2n}{\sqrt{k}}$, which proves the theorem. So for the rest of the proof we may assume that $|S| > k$. For convenience, define $g := f_{S|b} : \{-1, 1\}^S \rightarrow \mathbb{R}$. Let T be any subset of S of size k , and define $\bar{T} := S \setminus T$ (i.e., the complement of T with respect to the universe S). For an arbitrary $x \in \{-1, 1\}^{\bar{T}}$, consider the function $g_{T|x} : \{-1, 1\}^T \rightarrow \mathbb{R}$ obtained by restricting g further by setting the variables in $\bar{T}$ to x . We have for each $i \in T \subseteq S$,

$$\mathbb{E}_x[\widehat{g_{T|x}}(\{i\})^2] = \sum_{A \subseteq \bar{T}} \widehat{g}(A \cup \{i\})^2 \geq \widehat{g}(\emptyset \cup \{i\})^2 = \widehat{g}(\{i\})^2 \geq \frac{M^2}{4n^2},$$

where the equality is due to Corollary 23 and the last inequality holds by Lemma 17. By linearity of expectation, and since $|T| = k$, we have

$$\mathbb{E}_x \left[\sum_{i \in T} \widehat{g_{T|x}}(\{i\})^2 \right] = \sum_{i \in T} \mathbb{E}_x [\widehat{g_{T|x}}(\{i\})^2] \geq \frac{M^2 k}{4n^2}.$$

Therefore there exists a $x^* \in \{-1, 1\}^{\bar{T}}$ such that $\sum_{i \in T} (\widehat{g_{T|x^*}}(\{i\})^2) \geq \frac{M^2 k}{4n^2}$. Since $\text{Var}(h) = \sum_{S \neq \emptyset} \widehat{h}(S)^2$ for all functions h by Lemma 25, we have

$$\text{Var}(g_{T|x^*}) = \sum_{S \subseteq T, S \neq \emptyset} \widehat{g_{T|x^*}}(S)^2 \geq \sum_{i \in T} \widehat{g_{T|x^*}}(\{i\})^2 \geq \frac{M^2 k}{4n^2}.$$

Note that $g_{T|x^*}$ is the restriction of f to the subcube T , by setting the variables in $\bar{S}$ and $\bar{T}$ to b and x^* , respectively. This is a valid subcube for the denominator of $\text{val}_{\mathcal{Q}}$ (see Equation (1)), and so this subcube witnesses

$$\text{val}_{\mathcal{Q}} \leq \frac{M}{\sqrt{\frac{M^2 k}{4n^2}}} = \frac{2n}{\sqrt{k}},$$

which completes the proof. ◀

It only remains to prove our induction hypothesis, Lemma 17, which we do now.

Proof of Lemma 17. We prove the lemma by induction on n . Let $\|f\|_\infty = M$ and assume that $|f(x^*)| = M$.

- **Base case:** $n = 1$. Thus we have $f(-1) = M$ and $f(1) = -M$ (or the opposite, in which case essentially the same proof works). We have the variance over the whole cube as $\mathbf{Var}(f) = M^2 > \frac{M^2}{4}$. We also have $\widehat{f}(\{1\}) = \text{either } M \text{ or } -M$. In either case, $|\widehat{f}(\{1\})| = M \geq \frac{M}{2}$. This proves the base case.
- **Assumption for inductive step:** We assume the induction hypothesis to be true for $n = \ell$, i.e., for every balanced function f on ℓ variables, there exists a $S \subseteq [\ell]$ and $b \in \{-1, 1\}^{\bar{S}}$ such that $\mathbf{Var}(f_{S|b}) \geq \frac{M^2}{4\ell}$, and $|\widehat{f_{S|b}}(\{i\})| \geq \frac{M}{2\ell}$ for all $i \in S$.
- **Inductive step:** For the inductive step, we assume that the hypothesis holds for $n = \ell$ (see above) and show it is true for $n = \ell + 1$. Define $\mu_{i,b} := \mathbb{E}_{x: x_i=b}[f(x)]$ for $b \in \{-1, 1\}$. We consider two cases.
 - **Case 1:** $\exists i \in [\ell + 1]$ such that $|\mu_{i,x_i^*} - \mu_{i,-x_i^*}| \leq \frac{M}{\ell+1}$. Since we know f to be balanced, we have $\mu_{i,x_i^*} + \mu_{i,-x_i^*} = 0$, and hence by the triangle inequality

$$|\mu_{i,x_i^*}| = \left| \frac{\mu_{i,x_i^*} - \mu_{i,-x_i^*}}{2} + \frac{\mu_{i,x_i^*} + \mu_{i,-x_i^*}}{2} \right| \leq \frac{|\mu_{i,x_i^*} - \mu_{i,-x_i^*}|}{2} + \frac{|\mu_{i,x_i^*} + \mu_{i,-x_i^*}|}{2} \leq \frac{M}{2(\ell+1)}.$$

Define $f_i := f|_{x_i=x_i^*} - \mu_{i,x_i^*}$. Note that we have $\mathbb{E}[f_i] = 0$ and $\|f_i\|_\infty \geq M - \frac{M}{2(\ell+1)}$, where we used that the maximum absolute value of f is attained in the domain of f_i by construction (i.e., x^* is in the domain of f_i). Applying the induction hypothesis to f_i , we get a $S \subseteq [\ell + 1]$ and $b \in \{-1, 1\}^{\bar{S}}$ (with $i \in \bar{S}$ and $b_i = x_i^*$) such that the following two properties hold.

* We write $b' := b|_{\bar{S} \setminus \{i\}}$, i.e., the bit string b without the bit at position i , and obtain

$$\mathbf{Var}((f_i)_{S|b'}) \geq \frac{\left(M \left(1 - \frac{1}{2(\ell+1)}\right)\right)^2}{4\ell} \geq \frac{\left(M \left(\sqrt{1 - \frac{1}{\ell+1}}\right)\right)^2}{4\ell} = \frac{M^2}{4\ell} \cdot \frac{\ell}{\ell+1} = \frac{M^2}{4(\ell+1)}.$$

Here we are using $1 - \frac{1}{2\alpha} \geq \sqrt{1 - \frac{1}{\alpha}}$ for all $\alpha > 1$, which can easily be verified by squaring both sides and comparing terms. Since $f_{S|b} = (f_i)_{S|b'} + \mu_{i,x_i^*}$ and variance is unaffected by a constant additive shift, this implies $\mathbf{Var}(f_{S|b}) \geq \frac{M^2}{4(\ell+1)}$ as well.

* First note that for any function $g : \{-1, 1\}^n \rightarrow \mathbb{R}$ and any constant c , all Fourier coefficients of g are equal to the corresponding Fourier coefficients of $g - c$, except for the empty coefficient which gets subtracted by c . Thus, we again write $b' := b|_{\bar{S} \setminus \{i\}}$, to find that for all $j \in S$,

$$|\widehat{f_{S|b}}(\{j\})| = |\widehat{(f_i)_{S|b'}}(\{j\})| \geq \frac{M \left(1 - \frac{1}{2(\ell+1)}\right)}{2\ell} = \frac{M \cdot \frac{2\ell+1}{2(\ell+1)}}{2\ell} > \frac{M \cdot 2\ell}{2(\ell+1)} \cdot \frac{1}{2\ell} = \frac{M}{2(\ell+1)}.$$

- **Case 2:** For all $i \in [\ell + 1]$, we have $|\mu_{i,x_i^*} - \mu_{i,-x_i^*}| > \frac{M}{\ell+1}$. In that case, we choose $S = [\ell + 1]$ and b becomes the empty string, so that $f_{S|b} = f$. Then, we find
 - * By definition of Fourier coefficients, we have for all $i \in [\ell + 1]$,

$$|\widehat{f}(\{i\})| = |\mathbb{E}_x[f(x) \cdot x_i]| = \left| \frac{x_i^*}{2} (\mu_{i,x_i^*} - \mu_{i,-x_i^*}) \right| > \frac{M}{2(\ell+1)}.$$

* By Lemma 25, we have the variance of the whole subcube as

$$\mathbf{Var}(f) = \sum_{S \neq \emptyset} \widehat{f}(S)^2 \geq \sum_{i \in [\ell+1]} \widehat{f}(\{i\})^2 > (\ell+1) \frac{M^2}{4(\ell+1)^2} = \frac{M^2}{4(\ell+1)}. \quad \blacktriangleleft$$

4.3 Contiguous Blocks

The setting $\mathcal{Q} = \{[i, j] : i \leq j \in [n]\}$ corresponds to the model where the allowed substrings to be queried form contiguous blocks. While we don't phrase it as such, our results in this subsection also hold when the notion of contiguity allows for wraparound.

The following is our main result of this subsection.

► **Theorem 18.** *Let n be a positive integer and let $\mathcal{Q} = \{[i, j] : i \leq j \in [n]\}$. Then there exists a universal constant c such that*

$$\frac{cn}{\log n} \leq \text{val}_{\mathcal{Q}} \leq n.$$

Proof. Since $\mathcal{Q}$ contains all singletons, Proposition 15 implies the upper bound. For the lower bound, define $f : \{-1, 1\}^{mk} \rightarrow \mathbb{R}$ as $f(z^{(1)}, \dots, z^{(m)}) = \sum_{i=1}^m g(z^{(i)})$. Here each $z^{(i)} \in \{-1, 1\}^k$, and

$$g(z^{(i)}) = \begin{cases} 1, & z^{(i)} = 1^k, \\ -1, & z^{(i)} = (-1)^k, \\ 0, & \text{otherwise.} \end{cases}$$

First note that f is odd and $\max |f(x)| = m = n/k$. We show that every valid subcube in the denominator of the RHS of the expression of $\text{val}_{\mathcal{Q}}$ (see Equation (1)) has small variance, which proves the theorem.

For any subcube (S, b) where S is a contiguous block, since a contiguous block can partially intersect at most 2 blocks (let the free variables in these blocks be denoted by strings $y^{(1)}$ and $y^{(2)}$), we can assume that the restriction of f to the subcube (S, b) has the following form:

$$f_{S|b}(y^{(1)}, y^{(2)}, z^{(1)}, z^{(2)}, \dots, z^{(t)}) = h_1(y^{(1)}) + h_2(y^{(2)}) + \sum_{i=1}^t g(z^{(i)}) + c, \quad (4)$$

where $h_1, h_2 \in \{r_1, r_{-1}, r_0\}$ depending on the values of the set variables in the corresponding blocks, where for all $b \in \{-1, 0, 1\}$,

$$r_b(y) := \begin{cases} b, & \text{if } y \text{ is the all-}b \text{ string,} \\ 0, & \text{otherwise.} \end{cases}$$

In particular, r_0 is the constant 0 function. Since each block in the restricted function contains disjoint variables, the variance of the sum on the RHS of Equation (4) equals the sum of variances of the individual terms, which equals

$$= \text{Var}(h_1) + \text{Var}(h_2) + \sum_{i=1}^t \text{Var}(g) + 0 = O(1) + t \cdot \mathbb{E}[g^2] = O(1) + m \cdot \frac{1}{2^{k-1}},$$

where the second equality follows since h_i is a $\{-1, 0, 1\}$ -valued function, and each g is a balanced function, and the last equality follows since $t \leq m$ and g^2 takes value 1 at precisely two points and 0 everywhere else.

If we set n to be the number of variables for f , which equals mk , then the above variance equals $O(1) + \frac{n/k}{2^{k-1}}$. Thus we have

$$\text{val}_{\mathcal{Q}} = \frac{\|f\|_{\infty}}{\sqrt{\max_{S \in \mathcal{Q}, b \in \{-1, 1\}^S} \text{Var}(f_{S|b})}} = \frac{n/k}{\sqrt{O(1) + \frac{n/k}{2^{k-1}}}} = \Omega\left(\min\left\{\frac{n}{k}, \sqrt{\frac{n}{k}} \cdot 2^{\frac{k-1}{2}}\right\}\right).$$

Setting k to maximize this quantity asymptotically, we get $\frac{n}{k} = 2^{k-1}$, which implies $n = k \cdot 2^{k-1}$, implying $k = \log n - \log \log n + O(1)$, and hence $\text{val}_{\mathcal{Q}} = \Omega(n/\log n)$, proving the theorem. $\blacktriangleleft$

5 Conclusions

This work introduces a general framework for analyzing query algorithms for learning input strings in models that extend the search with wildcards setting [1]. As applications of our framework we are able to recover existing upper bounds [1, 3] in the search-with-wildcards setting and also show new bounds in generalized search-with-wildcards settings. Interestingly, our upper bounds in this paper are all shown (without explicitly resorting to SDP duality) using the *primal* version of the negative-weight adversary bound [15], which is a maximization SDP, typically used to show *lower bounds*. To the best of our knowledge, ours is the first work to use the primal adversary bound to establish novel upper bounds. Even though some of our results were already known from previous works [1, 3, 11, 8] through various differing techniques, we are able to recover the same bounds through a unifying framework. Our work makes progress towards answering an open question of [4], who asked if one could devise algorithms for learning input strings or computing some function of the input string, assuming non-standard query access to the input. It would be interesting to see if our results can be used to show any classical lower bounds using the framework developed in [4].

An interesting related direction is that of the AND/OR-query complexity of computing Boolean functions rather than learning an input string. It is not hard to show that the query model in the search-with-wildcards setting is equivalent to allowing unit-cost query access to arbitrary ORs and arbitrary ANDs of inputs. It was recently shown [10] that randomness does not give a superpolynomial advantage over determinism for computing total Boolean functions in this model. It would be interesting to see if ideas from our work could be generalized to show that quantumness also does not provide a superpolynomial advantage over randomness/determinism. Our framework relies on the optimizing matrix for the primal adversary bound being the communication matrix of an XOR function, which crucially uses symmetries of Parity (the function being computed), so it is not clear how to generalize our framework in this direction easily.

References

- 1 Andris Ambainis and Ashley Montanaro. Quantum algorithms for search with wildcards and combinatorial group testing. *Quantum Inf. Comput.*, 14(5-6):439–453, 2014. doi:10.26421/QIC14.5-6-4.
- 2 Robert Beals, Harry Buhrman, Richard Cleve, Michele Mosca, and Ronald de Wolf. Quantum lower bounds by polynomials. *J. ACM*, 48(4), 2001. doi:10.1145/502090.502097.
- 3 Aleksandrs Belovs. Quantum algorithms for learning symmetric juntas via the adversary bound. *Comput. Complex.*, 24(2):255–293, 2015. doi:10.1007/S00037-015-0099-2.
- 4 Shalev Ben-David, Adam Boulund, Ankit Garg, and Robin Kothari. Classical lower bounds from quantum upper bounds. In *2018 IEEE 59th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 339–349. IEEE, 2018. doi:10.1109/FOCS.2018.00040.
- 5 Anna Bernasconi and Bruno Codenotti. Spectral analysis of boolean functions as a graph eigenvalue problem. *IEEE transactions on computers*, 48(3):345–351, 2002.
- 6 Ethan Bernstein and Umesh V. Vazirani. Quantum complexity theory. *SIAM Journal on Computing*, 26(5):1411–1473, 1997. Earlier version in STOC’93. doi:10.1137/S0097539796300921.
- 7 Michel Boyer, Gilles Brassard, Peter Høyer, and Alain Tapp. Tight bounds on quantum searching. *Fortschritte der Physik: Progress of Physics*, 46(4-5):493–505, 1998.

- 8 Mark Bun and Nadezhda Voronova. Approximate degree lower bounds for oracle identification problems, 2023. doi:10.48550/arXiv.2303.03921.
- 9 Mark JP Chaisson, Richard K Wilson, and Evan E Eichler. Genetic variation and the de novo assembly of human genomes. *Nature Reviews Genetics*, 16(11):627–640, 2015.
- 10 Arkadev Chattopadhyay, Yogesh Dahiya, Nikhil S. Mande, Jaikumar Radhakrishnan, and Swagato Sanyal. Randomized versus deterministic decision tree size. In *Proceedings of the 55th Annual ACM Symposium on Theory of Computing (STOC)*, pages 867–880. ACM, 2023. doi:10.1145/3564246.3585199.
- 11 Richard Cleve, Kazuo Iwama, François Le Gall, Harumichi Nishimura, Seiichiro Tani, Junichi Teruyama, and Shigeru Yamashita. Reconstructing strings from substrings with quantum queries. In Fedor V. Fomin and Petteri Kaski, editors, *Algorithm Theory – SWAT 2012*. Springer Berlin Heidelberg, 2012.
- 12 Arjan Cornelissen. *Quantum multivariate estimation and span program algorithms*. University of Amsterdam, 2023.
- 13 Edward Farhi, Jeffrey Goldstone, Sam Gutmann, and Michael Sipser. Bound on the number of functions that can be distinguished with k quantum queries. *Physical Review A*, 60(6):4331, 1999.
- 14 Lov K Grover. A fast quantum mechanical algorithm for database search. In *Proceedings of the twenty-eighth annual ACM Symposium on Theory of Computing (STOC)*, pages 212–219, 1996. doi:10.1145/237814.237866.
- 15 Peter Høyer, Troy Lee, and Robert Špalek. Negative weights make adversaries stronger. In *Proceedings of the thirty-ninth annual ACM Symposium on Theory of Computing (STOC)*, pages 526–535, 2007.
- 16 Heng Li and Richard Durbin. Genome assembly in the telomere-to-telomere era. *Nature Reviews Genetics*, 25(9):658–670, 2024.
- 17 Nikhil Shekhar Mande. *Communication complexity of XOR functions*. PhD thesis, PhD thesis, Tata Institute of Fundamental Research Mumbai, 2018.
- 18 Ryan O’Donnell. *Analysis of boolean functions*. Cambridge University Press, 2014.
- 19 Ben W. Reichardt. Span programs and quantum query complexity: the general adversary bound is nearly tight for every Boolean function. In *2009 50th Annual IEEE Symposium on Foundations of Computer Science—FOCS*, pages 544–551. IEEE Computer Soc., Los Alamitos, CA, 2009. doi:10.1109/FOCS.2009.55.
- 20 Ben W. Reichardt. Reflections for quantum query algorithms. In *Proceedings of the Twenty-Second Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 560–569. SIAM, Philadelphia, PA, 2011. doi:10.1137/1.9781611973082.44.
- 21 Gordon Robertson, Jacqueline Schein, Readman Chiu, Richard Corbett, Matthew Field, Shaun D Jackman, Karen Mungall, Sam Lee, Hisanaga Mark Okada, Jenny Q Qian, et al. De novo assembly and analysis of RNA-seq data. *Nature methods*, 7(11):909–912, 2010.
- 22 Steven S Skiena and Gopalakrishnan Sundaram. Reconstructing strings from substrings. *Journal of Computational Biology*, 2(2):333–353, 1995. doi:10.1089/CMB.1995.2.333.
- 23 Wim van Dam. Quantum oracle interrogation: Getting all information for almost half the price. In *Proceedings 39th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 362–367. IEEE, 1998. doi:10.1109/SFCS.1998.743486.
- 24 Yongzhen Xu, Shihao Zhang, and Lvzhou Li. Quantum algorithm for learning secret strings and its experimental demonstration. *Physica A: Statistical Mechanics and its Applications*, 609:128372, 2023.

A Quantum Query Complexity

Let $\mathcal{H}$ be a finite-dimensional Hilbert space. Any quantum algorithm $\mathcal{A}$ can be represented as a sequence of unitary operators $\{U_i\}_{i=0}^T \in \mathcal{L}(\mathcal{H})$ on an initial state $|\psi_0\rangle \in \mathcal{H}$. The algorithm is terminated by a measurement on $|\phi\rangle = U_T U_{T-1} \cdots U_1 U_0 |\psi_0\rangle$ to observe an outcome.

Let D and E be finite sets and $F : D \rightarrow E$ be a function. For every $x \in D$, let O_x be a unitary operator that depends on x , referred to as the *oracle*. A quantum query algorithm $\mathcal{A}$ alternates input-independent unitary operators with calls to the oracle, i.e., it applies the operations U_0, O_x, U_1, O_x , etc. The oracle calls are referred to as *queries* to the input x . The algorithm $\mathcal{A}$ is a bounded-error, quantum query algorithm for computing F if $\forall x \in D$ the output of $\mathcal{A}$ is $F(x)$ with probability at least $2/3$. The quantum query complexity of computing F is the minimum number of queries required to compute F with probability $2/3$. This is denoted by $Q(F)$.

In this work, we consider algorithms with access to different *query sets* and therefore use $Q^{\mathcal{R}}(F)$ to refer to the quantum query complexity of computing F using query set $\mathcal{R}$. Formally a query set $\mathcal{R}$ is a set consisting of functions $r : \{-1, 1\}^n \rightarrow \mathcal{O}$ for some discrete range $\mathcal{O}$. The usual query model is captured by the query set $\mathcal{R} = \{r_i : \{-1, 1\}^n \rightarrow \{-1, 1\} \mid i \in [n]\}$ where $r_i(x) := x_i$. The final state of a quantum query algorithm with query set $\mathcal{R}$ looks like

$$|\phi\rangle = U_T O_x \dots U_2 O_x U_1 O_x U_0 |\psi_0\rangle$$

where the action of O_x on each basis state is

$$O_x : |r\rangle |b\rangle \rightarrow |r\rangle |b + r(x) \bmod |\mathcal{O}|\rangle \quad \forall r \in \mathcal{R}, b \in \{0, 1, \dots, |\mathcal{O}|\}.$$

Let $x \in \{-1, 1\}^n$ be an input string and $\mathcal{Q} \subseteq 2^{[n]}$. We abuse notation and use $\mathcal{Q}$ to be the query set consisting of all queries $q = (S, b_S)$ of the form $q(x) = \mathbb{I}[x_S = b_S]$ where $S \in \mathcal{Q}$, $b_S \in \{-1, 1\}^S$ and x_S is the substring of x restricted to indices in S . Throughout this work, we only work with queries of this form. Thus, for $\mathcal{Q} \subseteq 2^{[n]}$, we abuse notation and use $\mathcal{Q}$ to also denote the query set $\{q = (S, b) : S \in \mathcal{Q}, b \in \{-1, 1\}^S\}$.

The primal adversary bound is a semi-definite program that characterizes the quantum query complexity of computing function $F : D \rightarrow E$, relative to a query set $\mathcal{R}$. To that end, $\forall q \in \mathcal{R}, \Delta_q \in \{0, 1\}^{D \times D}$ is the matrix defined as:

$$\Delta_q[x, y] = \begin{cases} 1, & q(x) \neq q(y), \\ 0, & \text{otherwise.} \end{cases}$$

That is, $\Delta_q[x, y] = 1$ if and only if the query q can distinguish between inputs x and y . Throughout this work, we use the notation $\text{ADV}^{\mathcal{R}, \pm}$ to denote the adversary bound relative to query set $\mathcal{R}$. Now we state the primal version of the general adversary bound, which we will refer to as the *primal adversary bound*.

Primal adversary bound

$$\begin{aligned} \text{ADV}^{\mathcal{R}, \pm}(F) = \max_{\Gamma} \quad & \frac{\|\Gamma\|}{\max_{q \in \mathcal{R}} \|\Gamma \circ \Delta_q\|} \\ \text{subject to} \quad & \Gamma \in \mathbb{R}^{D \times D}, \\ & \Gamma \text{ is symmetric,} \\ & \Gamma[x, y] = 0. \end{aligned} \tag{5} \quad (\forall x, y \in D \text{ with } F(x) = F(y))$$

Since the objective function is invariant under multiplying Γ with a constant factor, we can without loss of generality assume that the denominator is equal to 1. Thus, we can equivalently write the SDP computing $\text{ADV}^{\mathcal{R}, \pm}$ with objective function $\max_{\Gamma} \|\Gamma\|$, and the additional constraint that $\forall q \in \mathcal{R}, \|\Gamma \circ \Delta_q\| \leq 1$.

Reichardt [20] showed that the primal adversary bound is a tight characterization of quantum query complexity.

► **Theorem 19** ([20, Theorem 1.4]). *Let D and E be finite sets and $F : D \rightarrow E$ be a function. The bounded-error, quantum query complexity of computing F using query set $\mathcal{R}$ is characterized by the adversary bound.*

$$Q^{\mathcal{R}}(F) = \Theta(\text{ADV}^{\mathcal{R}, \pm}(F)).$$

Although [20, Theorem 1.4] states the adversary bound for index queries, it can be observed that the adversary bound characterizes the quantum query complexity for any query set and therefore we present it as such. Let $\mathcal{R}$ be a query set (assume for simplicity that all queries output values in $\{-1, 1\}$). The argument below can easily be generalized if this is not the case). For every $x \in D$, define the string $\mathcal{R}(x) = (r(x))_{r \in \mathcal{R}}$. Define the partial function $G : \{-1, 1\}^{\mathcal{R}} \rightarrow E$ on the input domain $\{z \in \{-1, 1\}^{\mathcal{R}} : z = \mathcal{R}(x) \text{ for some } x \in D\}$ as $G(\mathcal{R}(x)) := F(x)$. Observe that querying a input variable, say $R \in \mathcal{R}$, to G on input $\mathcal{R}(x)$ is exactly the same as making the query R on input x . Thus, $Q^{\mathcal{R}}(F) = Q(G)$ and $\text{ADV}^{\mathcal{R}, \pm}(F) = \text{ADV}^{\pm}(G)$, and therefore the result follows by applying [20, Theorem 1.4] to G .

When $\mathcal{Q} \subseteq 2^{[n]}$, and the query set consists of all queries of the form $\mathbb{I}[x_S = b]$ where $S \in \mathcal{Q}$ and $b \in \{-1, 1\}^S$ are arbitrary, recall that we abuse notation and use $\mathcal{Q}$ to also denote the query set $\{q = (S, b) : S \in \mathcal{Q}, b \in \{-1, 1\}^S\}$. Thus $\text{ADV}^{\mathcal{Q}, \pm}(\cdot)$ and $Q^{\mathcal{Q}}(\cdot)$ denote the negative-weight adversary bound and the quantum query complexity, respectively, with respect to this query set.

B Fourier Analysis of Boolean Functions and Restrictions

Below are some relevant formal definitions and Fourier-analytic properties of restricted functions, adapted from [18].

Consider the vector space of functions from $\{-1, 1\}^n$ to $\mathbb{R}$, equipped with the following inner product.

$$\langle f, g \rangle := \mathbb{E}_{x \in \{-1, 1\}^n} [f(x)g(x)] = \frac{1}{2^n} \sum_{x \in \{-1, 1\}^n} f(x)g(x).$$

For a set $S \subseteq [n]$, define the parity function $\chi_S : \{-1, 1\}^n \rightarrow \mathbb{R}$ by $\chi_S(x) = \prod_{i \in S} x_i$. It is not hard to show that the set of all parities $\{\chi_S : S \subseteq [n]\}$ forms an orthonormal basis of the space above. Thus, every function $f : \{-1, 1\}^n \rightarrow \mathbb{R}$ has a unique expansion with respect to this basis, $f = \sum_{S \subseteq [n]} \widehat{f}(S) \chi_S$. This expansion is called the Fourier expansion of f and the coefficients $\{\widehat{f}(S) : S \subseteq [n]\}$ are called the Fourier coefficients of f .

We list several properties of Fourier coefficients.

► **Lemma 20** ([5], also see [17, Lemma 2.2.3]). *Let $f : \{-1, 1\}^n \rightarrow \mathbb{R}$ be a function and let M be the communication matrix of $f \circ \text{XOR}$. Then,*

$$\|M\| = 2^n \max_{S \subseteq [n]} |\widehat{f}(S)|.$$

► **Definition 21.** *Let $f : \{-1, 1\}^n \rightarrow \mathbb{R}$ be a function, and let $(J, \bar{J})$ be a partition of $[n]$. Let $z \in \{-1, 1\}^{\bar{J}}$. Then we write $f_{J|z} : \{-1, 1\}^J \rightarrow \mathbb{R}$ (pronounced “the restriction of f to J using z ”) for the subfunction of f obtained by fixing the coordinates in $\bar{J}$ to the bit values z , i.e.,*

$$\forall x \in \{-1, 1\}^J, f_{J|z}(x) := f(x_J z_{\bar{J}})$$

where $x_J z_{\bar{J}}$ represents the string that equals x on J and z on $\bar{J}$.

► **Proposition 22** ([18, Proposition 3.21]). *Let $f : \{-1, 1\}^n \rightarrow \mathbb{R}$ and let $(J, \bar{J})$ be a partition of $[n]$. For $z \in \{-1, 1\}^{\bar{J}}$ and $S \subseteq J$ we have*

$$\widehat{f_{J|z}}(S) = \sum_{T \subseteq \bar{J}} \widehat{f}(S \cup T) \chi_T(z).$$

► **Corollary 23** ([18, Corollary 3.22]). *Let $(J, \bar{J})$ be a partition of $[n]$ and fix $S \subseteq J$. Suppose $z \sim \{-1, 1\}^{\bar{J}}$ is chosen uniformly at random. Then,*

$$\mathbb{E}_z[\widehat{f_{J|z}}(S)^2] = \sum_{T \subseteq \bar{J}} \widehat{f}(S \cup T)^2.$$

The proof of the following lemma is available in the full version of the paper.

► **Lemma 24.** *For $f : \{-1, 1\}^n \rightarrow \mathbb{R}$, the following conditions are equivalent:*

- *For all x with $|x|$ even, $f(x) = 0$.*
- *For all $S \subseteq [n]$, we have $\widehat{f}(S) = -\widehat{f}(\bar{S})$.*

We require the following well-known expression for the variance of a function. See, for example, [18, Proposition 1.13] for a simple proof.

► **Lemma 25.** *Let $f : \{-1, 1\}^n \rightarrow \mathbb{R}$ be a function. Then,*

$$\text{Var}(f) = \sum_{S \neq \emptyset} \widehat{f}(S)^2.$$

Provable Speedups in Convex Optimization via Quantum Dynamics

Shouvanik Chakrabarti¹  

Global Technology Applied Research,
JPMorganChase, New York, NY, USA

Jacob Watkins¹  

Global Technology Applied Research,
JPMorganChase, New York, NY, USA

Brandon Augustino  

Global Technology Applied Research,
JPMorganChase, New York, NY, USA

Marco Pistoia 

Global Technology Applied Research,
JPMorganChase, New York, NY, USA

Dylan Herman¹  

Global Technology Applied Research,
JPMorganChase, New York, NY, USA

Enrico Fontana  

Global Technology Applied Research,
JPMorganChase, New York, NY, USA

Junhyung Lyle Kim  

Global Technology Applied Research,
JPMorganChase, New York, NY, USA

Abstract

This work investigates the possibility of quantum speedups for continuous optimization through quantum Hamiltonian simulation. We establish the first rigorous query complexity bounds for unconstrained convex optimization via digital quantum annealing, based on the non-adiabatic Quantum Hamiltonian Descent (QHD) framework. In the process, we derive rigorous resource estimates for digital quantum simulation of Schrödinger operators depending only on input simulation parameters, given black-box evaluation access to separable, Lipschitz continuous potential $b(t)f(x)$. We apply these simulation bounds to assess the complexity of high-dimensional convex optimization.

Our annealing schedule achieves *arbitrarily fast* convergence rates in the evolution time, with computational time determined solely by the cost of discretization. We show that a G -Lipschitz convex function can be optimized to an error of ϵ with $\tilde{O}(d^{1.5}G^2R^2/\epsilon^2)$ queries, given a starting point that is Euclidean distance R from optimal. Under reasonable assumptions such as the complexity of simulating Schrödinger operators, we show that $\tilde{\Omega}(d/\epsilon^2)$ queries are necessary. This suggests QHD does not offer improvements over classical methods in the noiseless zeroth order setting.

However, we show that the QHD algorithm can tolerate $\tilde{O}(\epsilon^3/d^{1.5}G^2R^2)$ noise in function evaluation, and as a result, provides a super-quadratic query advantage over the best existing noise-tolerant classical algorithms in the high-dimensional setting. We leverage this to design a quantum algorithm for stochastic convex optimization that offers a super-quadratic speedup over all known classical and quantum algorithms in this regime. To our knowledge, these results represent the first rigorous quantum speedups for convex optimization obtained through a dynamical algorithm.

2012 ACM Subject Classification Theory of computation $\rightarrow$ Quantum complexity theory

Keywords and phrases Convex optimization, Hamiltonian simulation, zeroth order

Digital Object Identifier 10.4230/LIPIcs.TQC.2026.4

Related Version *Full Version:* <https://arxiv.org/abs/2503.24332> [27]

Acknowledgements The authors thank Aram Harrow, Jiaqi Leng, and Xiaodi Wu for insightful discussions about Quantum Hamiltonian Descent and the simulation of real-space dynamics. They also thank their colleagues at the Global Technology Applied Research center of JPMorganChase for their support and helpful discussions.

¹ These authors contributed equally and are listed in alphabetical order.



© Shouvanik Chakrabarti, Dylan Herman, Jacob Watkins, Enrico Fontana, Brandon Augustino, Junhyung Lyle Kim, and Marco Pistoia;
licensed under Creative Commons License CC-BY 4.0

21st Conference on the Theory of Quantum Computation, Communication and Cryptography (TQC 2026).
Editors: Rotem Arnon and Aram W. Harrow; Article No. 4; pp. 4:1–4:25



Leibniz International Proceedings in Informatics
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

1 Introduction

Quantum computers have the potential to solve complex optimization problems more efficiently than classical methods. The degree of quantum algorithmic speedup in optimization has been the subject of many studies spanning over two decades (for recent surveys see [1, 30]). Interest in this topic is driven both by the mathematical richness of problems in the field and by the many practical applications to machine learning, statistics, finance, and engineering.

Perhaps the most natural optimization algorithms are those that directly simulate the continuous time dynamics of some physical system. These algorithms seek to leverage the fact that such systems experience forces in the direction of minimal potential energy or related quantities. If a system can be identified where these quantities correspond to an objective function, then the simulation of that system yields an optimization algorithm. These connections have been leveraged to great success in classical optimization theory [72, 76, 47]. Their most notable success has been in explaining the so-called *acceleration phenomenon*, [62, 58, 34] wherein certain finely-tuned gradient methods have been shown to achieve asymptotically faster convergence than standard gradient descent in a variety of domains.

From the point of view of quantum algorithms, the study of optimization through simulating quantum dynamics has been ongoing since the beginning of the field. Perhaps the most famous example is the quantum adiabatic algorithm for combinatorial optimization [37], which leverages *adiabaticity* [14]. This is the property in which slowly changing Hamiltonian systems remain in their ground state (optimum) even as the Hamiltonian varies. Due to the flexibility of this framework, the adiabatic algorithm and a heuristic variant called quantum annealing [46] have been heavily studied from both theoretical and empirical perspectives [18, 19, 36, 74, 63, 4, 6, 9, 13]. For a comprehensive treatment of adiabatic quantum computing, we refer the reader to [5] and the references therein. Other optimization algorithms were originally proposed in the discrete time regime but subsequently analyzed from a dynamical/optimal control perspective, such as the Quantum Approximate Optimization Algorithm (QAOA) [35, 77, 17, 75] and the Variational Quantum Eigensolver (VQE) [61, 53, 54, 29]. However, these algorithms are primarily intended for discrete optimization.

In contrast, this work considers the continuous setting, specifically unconstrained convex optimization.

► **Problem 1 (Unconstrained Convex Optimization).** *Let $f: \mathbb{R}^d \rightarrow \mathbb{R}$ be a convex¹ G -Lipschitz continuous function with global minimum $f_\star$ and set of global minimizers $\mathcal{X}_\star$. Given an input point x_0 such that $\ell_2(x_0, \mathcal{X}_\star) \leq R$, find a point $\tilde{x} \in \mathbb{R}^d$ such that*

$$f(\tilde{x}) - f_\star \leq \epsilon.$$

Thus, R bounds the known distance from solution, and ϵ sets the acceptable error tolerance. Our focus on convex optimization stems from both practical and theoretical considerations. Practical, because unconstrained convex optimization has plentiful applications to statistics and machine learning [16]. Theoretical, because the rigorous study of the complexity of dynamical algorithms for optimization is an emerging topic, and it is instructive to focus on a domain where classical complexities are well understood [59]. Unconstrained convex optimization provides such a domain, where the optimal query complexities in the standard regimes have been characterized for decades [57]. There is also extensive research activity on variants of this setting, such as erroneous oracular access to the function or its derivatives [71, 31, 11], stochastic optimization [24, 56, 32, 69], and bandit optimization [2, 3, 21, 68, 33].

¹ Our results also hold for *star*-convex functions, all other assumptions kept the same. We only require convexity conditions to hold between a fixed global minimizer $x_\star$ and any point y in the domain.

A dynamical approach for characterizing acceleration in convex optimization was introduced by Su, Boyd, and Candès [72], and subsequently generalized by Wibisono, Wilson, and Jordan [76]. In the latter work, the authors define a class of (classical) dynamical systems in terms of a so-called Bregman Lagrangian

$$\mathcal{L}(x, v, t) = e^{\alpha_t + \gamma_t} (D_h(x + e^{-\alpha_t} v, x) - e^{\beta_t} f(x)),$$

where x denotes the position, $v = \dot{x}$ denotes the velocity, and D_h is the Bregman divergence $D_h(x, y) := h(x) - h(y) - \langle \nabla h(x), y - x \rangle$, where h is convex and essentially smooth. Later, Leng et al. [48] quantized these dynamics in the Euclidean setting $h(x) = \|x\|^2/2$, which they term *Quantum Hamiltonian Descent (QHD)*. These dynamics are described by a time dependent Schrödinger equation, with Hamiltonian

$$H_{\text{QHD}}(t) := -e^{\alpha_t - \gamma_t} \frac{1}{2} \Delta + e^{\alpha_t + \beta_t + \gamma_t} f(x). \quad (\text{QHD})$$

In both the classical and quantum frameworks, $\alpha_t, \beta_t, \gamma_t$ are differentiable real-valued functions, which we call *schedules*. Wibisono et al. [76] show that if the schedules satisfy so-called *ideal scaling* conditions, then for convex f the classical dynamics converge to a global minimum at rate $O(e^{-\beta_t})$. Leng et al. [48] show a similar result for quantum dynamics. What is remarkable about these protocols is that they offer a *non-adiabatic* pathway to annealing-based optimization, whose convergence is relatively insensitive to both initial state and spectral properties of the Hamiltonian.

Ideal scaling schedules are invariant under a reparameterization of time, and in principle, the rate of convergence can be arbitrarily fast in t . Clearly, however, the query complexity of convex optimization cannot be made arbitrarily small, as this would contradict known lower bounds in the classical [57, 59] and quantum [26, 73, 40, 39] settings. Wibisono et al. [76] offer an explanation of this phenomenon by noting that while the convergence of continuous dynamics can be arbitrarily fast, algorithms for discretizing the dynamics become unstable beyond a speed limit. They additionally argue that the maximum speed where discretization is possible corresponds to the optimal (accelerated) classical query complexities. Leng et al. [48] hint at a similar speed/discretization tradeoff for quantum algorithms, but this tradeoff has yet to be theoretically characterized.

Our main technical contribution is to give a precise theoretical characterization of the dependence of the query complexity on schedules. This leads to rigorous bounds on the query complexity of convex optimization through the simulation of QHD dynamics. We identify precise speed-discretization tradeoffs and show that our results cannot be improved under believable assumptions about initial state and the complexity of simulating this class of Schrödinger operators (e.g., Assumption 9). As a consequence, we identify regimes where QHD simulation does not offer any speedup over comparable classical algorithms, and yet others where (perhaps surprisingly) large speedups over the best known classical algorithms are possible. The next section discusses our results in detail.

2 Input Model

We study an oracular problem in which the objective function f is accessed via evaluation (“zeroth-order”) queries. Our analysis is primarily concerned with the query complexity, namely the number of calls to such oracles. Quantum computing allows for several different natural access models for f ; here we define the ones of relevance to this work. The number of qubits, and hence precision to which x is encoded in a finite-dimensional basis state, will be specified when necessary.

► **Definition 2** (ϵ_f -accurate evaluation oracle). *Let $f : \mathcal{X} \rightarrow \mathbb{R}$, where $\mathcal{X} \subseteq \mathbb{R}^d$. The unitary O_f is an ϵ_f -accurate evaluation, or binary, oracle for f if for all computational basis states $|x\rangle|y\rangle$,*

$$O_f |x\rangle|y\rangle = |x\rangle|y \oplus \tilde{f}(x)\rangle,$$

and $\|\tilde{f}(x) - f(x)\|_\infty < \epsilon_f$.

► **Definition 3** (ϵ_p -accurate phase oracle). *For $\mathcal{X} \subseteq \mathbb{R}^d$, let $f : \mathcal{X} \rightarrow \mathbb{R}$ be a bounded function with bound $\Lambda \geq \|f\|_\infty$. The unitary O_f is an ϵ_p -accurate phase oracle for f if for all computational basis states $|x\rangle$,*

$$O_f |x\rangle = e^{i\tilde{h}(x)}|x\rangle,$$

where $\|\tilde{h} - f/\Lambda\|_\infty < \epsilon_p$.

Conversions from a binary to phase oracle can be made via phase kickback, and the reverse conversion can be done with quantum phase estimation. Finally, we will make use of a “qsample” oracle for the setting of stochastic convex optimization.

► **Definition 4** (quantum stochastic oracle). *Let f be a real-valued function with domain $\mathcal{X} \subseteq \mathbb{R}^d$. For each $x \in \mathcal{X}$, let p_x be a probability density on $\mathbb{R}$ with zero mean and bounded variance. The unitary O_f is an ϵ_f -accurate stochastic oracle for f if it acts (in superposition) on a joint register $|x\rangle|y\rangle$ of domain points $x \in \mathcal{X}$ and values y as*

$$O_f |x\rangle|y\rangle = |x\rangle \int \sqrt{p_x(\xi)} |y \oplus (f(x) + \xi)\rangle d\xi. \quad (1)$$

In many ways, the quantum stochastic oracle above is a natural generalization of a stochastic classical oracle, in which $f(x) + \xi$ is returned with probability $p_x(\xi)$. For example, see [70] for a comparable setup in the first-order setting. However, we acknowledge that, in some respects, the quantum and classical oracles are less directly comparable than for more familiar deterministic evaluation oracles. For example, it’s unclear how O_f could be constructed “on the fly” using a stream of classically random samples, given the need for superposition. On the other hand, if the data is loaded into quantum read-only memory (QROM), then the qsampling of this data can be done in analogous fashion to the classical case. See Remark 13 for a discussion of the consequences on query complexity, if we only have classical access to the noise.

3 Results

We present our results in two parts. First, we informally state our main theorem regarding the simulation of Schrödinger operators, given evaluation access to a potential f . Then, based on these resource estimates, we discuss results for optimization using the QHD framework.

3.1 Real space Hamiltonian simulation

Classically, the simulation of dynamical systems may be performed using various numerical integration techniques [42, 12]. In the case where (closed) quantum dynamics are simulated on a quantum computer, these integrators correspond to algorithms for *time dependent Hamiltonian simulation*, which synthesize the unitary time evolution operator as a quantum circuit. Such algorithms are well-studied in the discrete setting. A careful study of real

space simulation complexity has only been recently undertaken [28], though this analysis focuses primarily on Fourier truncation errors. In order to characterize the query complexity of optimization with QHD, we require optimized and explicit resource estimates for the complexity of simulation with black box potentials, making as few assumptions as possible and accounting for all sources of error.

We present and analyze a quantum algorithm for real-space simulation based on a pseudo-spectral method known as collocation. The main novelty of the analysis is handling the dynamical propagation of discretization error. Existing results on Schrödinger operators and partial differential equations (PDEs) are often tailored to contexts in mathematical physics, e.g. applicable only to a few spatial dimensions, and are thus not immediately useful for complexity-theoretic statements. Our work fills this gap and provides a *computational* result regarding the digital simulation of Schrödinger equations in Euclidean space, which can be applied out-of-the-box. In particular, the result applies when the dimension of the problem is taken to be an asymptotic parameter, which is of crucial importance to high-dimensional settings such as optimization.

The following informally summarizes our main simulation theorem.

► **Theorem 5** (Informal version of Thm 4.2 in Main Version [27]). *Consider the Schrödinger equation*

$$i\partial_t \Phi(x, t) = [-a(t)\Delta + b(t)f(x)]\Phi(x, t),$$

subject to initial data $\Phi(x, 0) = \Phi_0(x)$, where a, b are sufficiently smooth functions of time and f is a G -Lipschitz function with bound $\Lambda_f \geq \|f\|_\infty$, accessed through a ϵ_f -noisy binary quantum oracle O_f . If $\epsilon_f = \tilde{O}(\epsilon/\|b\|_1)$, then there is a digital quantum algorithm that outputs an ϵ -approximation $|\Psi_t\rangle$ to $\Phi(\cdot, t)$ (in an appropriate sense defined later) using $\mathcal{O}(\Lambda\|b\|_1)$ queries to O_f , $\mathcal{O}(d^2 \text{polylog}(1/\epsilon, \|a\|_1, \|b\|_1))$ qubits and $\tilde{O}(\text{poly}(d, \Lambda, \|b\|_1))$ gates.

To the best of our knowledge, this is the first rigorous result on the quantum computational complexity of real-space quantum dynamics given solely in terms of basic simulation parameters. We prove Theorem 5 using multi-dimensional Fourier analysis and bounding various sources of error that typically occur in signal processing applications, such as spectral truncation and aliasing. These errors are connected to the so-called Sobolev norms of the functions involved, and occur because the digital quantum algorithm is unable to fully represent the infinite spectrum of the continuous problem. Our results apply to separable potentials $b(t)f(x)$ where b is differentiable and f is G -Lipschitz. These weak assumptions on smoothness are possible through a mollification argument, a common technique in signal processing and functional analysis. Further details are provided in Section 4.2 and Appendix A. As these simulation results are concerned with general classes of Schrödinger dynamics, they may be of interest beyond optimization.

3.2 Convex optimization

Our main theorem on unconstrained convex optimization via Hamiltonian simulation is as follows.

► **Theorem 6.** *There exists a quantum algorithm that solves Problem 1 using $\tilde{O}(d^{3/2}(GR/\epsilon)^2)$ queries to an evaluation oracle for f (Definition 2) with error $\epsilon_f = \tilde{O}\left(\frac{\epsilon^3}{d^{3/2}G^2R^2}\right)$.*

In Section 4.3, we prove this result as a simple consequence of a more general theorem. A full proof of the general result is given in the Section 5 of the Full Version [27], but here we outline a sketch of the derivation. With the simulation results of Theorem 5, we can calculate the

query complexity of optimization via QHD once we provide convergence results in continuous time. Theorem 14 provides this convergence under our modified ideal scaling conditions, which relaxes the conditions on the potential from being differentiable, as in [48], to merely Lipschitz continuous. We note that this relaxation is for *any* schedule that obeys the ideal scaling conditions. Since our goal is to completely determine the query complexity, we keep track of (possibly dimension-dependent) factors in addition to the convergence rate. Our complexity scales as $d^{3/2}$ instead of d because of a $\sqrt{d}$ factor that arises from the mismatch between the $p = \infty$ and $p = 2$ norm. We expect this may be improved with refined analysis.

Our analysis considers a modified parametrization of the QHD Hamiltonian from (QHD). In what follows we write

$$H_{\text{QHD}}(t) = c_t \left(\frac{\hat{p}^2}{2m_t} + m_t \omega_t^2 f(\hat{x}) \right), \quad (2)$$

where m_t, ω_t are positive differentiable functions satisfying the ideal scaling conditions

$$\frac{\dot{m}_t}{m_t} = \lambda c_t, \quad \frac{\dot{\omega}_t}{\omega_t} \leq \frac{1}{2} \lambda c_t \quad (\text{Ideal Scaling})$$

for some $\lambda \in \mathbb{R}_+$. Besides providing greater physical intuition, this parametrization also ensures dimensional consistency, whereas prior expressions for Ideal Scaling left this point unclear. Evidently, the parameter $c_t = e^{\alpha t}$ merely plays the role of time parametrization. While such parametrization can affect algorithmic performance, the time dependent simulation algorithm we employ, based on the Dyson series method, will be robust to such reparametrizations.

Note that the query complexity in Theorem 6 has a dependence of $1/\epsilon^2$ on the precision, which is due to the query cost of discretization. In fact, the same complexity can be obtained from a schedule with *arbitrarily fast* convergence (and consequently, arbitrarily small simulation time). As specific examples, we now define schedules with arbitrary exponential or polynomial convergence rates. These schedules were proposed in [76] and considered in the quantum setting in [48].

► **Definition 7** (Exponential Schedules with Convergence Rate e^{-ct}). *Let $c \geq 0$ be a parameter that determines the rate of convergence, and $m_0, \omega_0 \in \mathbb{R}_+$ be arbitrary. A family of QHD Hamiltonians with exponential convergence rate e^{-ct} is defined by schedules $c_t = c$, $m_t = m_0 e^{ct}$, and $\omega_t = \omega_0 e^{ct/2}$. The resulting Hamiltonian is*

$$H_c(t) = c \left(-e^{-ct} \frac{1}{2m_0} \Delta + e^{2ct} m_0 \omega_0^2 f \right) \quad (\text{QHD-exponential})$$

for $t \in [0, \infty)$.

► **Definition 8** (Polynomial Schedules with Convergence Rate t^{-k}). *Let $k > 0$ be a parameter that determines the rate of convergence, and $m_0, \omega_0, t_0 \in \mathbb{R}_+$ be arbitrary. A family of QHD Hamiltonians with polynomial convergence rate t^{-k} is defined by schedules $c_t = k/t$, $m_t = m_0 (t/t_0)^k$, and $\omega_t = \omega_0 (t/t_0)^{k/2}$. The resulting Hamiltonian is*

$$H_k(t) = \frac{k}{t} \left(-(t/t_0)^{-k} \frac{1}{2m_0} \Delta + (t/t_0)^{2k} m_0 \omega_0^2 f \right) \quad (\text{QHD-polynomial})$$

for $t \in [t_0, \infty)$.

The rate of convergence for these schedules, here provided within the definitions for clarity, are simple corollaries of Theorem 14. We also show how any ideal-scaling schedule can be discretized to obtain the query complexity stated in Theorem 6. Finally, we note that, while

the choice of convergence rate is relatively arbitrary and leads to the same query complexity, the remaining parameters of the schedule such as m_0, ω_0 must be chosen carefully. This is because the convergence of QHD is shown via a Lyapunov (“nonincreasing”) function $\mathcal{E}_t$ which loosely tracks the “energy” of the system. The convergence bounds are directly proportional to $\mathcal{E}_0$, which may be dimension dependent. In fact, using QHD schedules already stated in previous works [48, 49] result in a dimension dependence of $\tilde{\mathcal{O}}(d^{2.5})$ in the query complexity, compared to the $\tilde{\mathcal{O}}(d^{1.5})$ obtained here.

3.2.1 Schedule invariance and lower bounds on query complexity

We now consider the possibility of improvements over our upper bounds within the QHD framework. The algorithmic results we provide are independent of ideal-scaling schedule, but perhaps there are optimizations in terms of the initial parameters, the initial state, or in terms of the domain of simulation. As discussed in the previous section, we can choose schedules that converge in continuous time at arbitrarily fast rates, and thus simulate for an arbitrarily short amount of time to reach the target precision. Lower bounds on the query complexity arise from the increasing cost of simulating these arbitrarily fast schedules. We make the following assumption on the complexity of simulating Schrödinger Dynamics.

► **Assumption 9** (No Fast-Forwarding for Time Dependent Schrödinger Operator). *A potential $f : \mathbb{R}^d \rightarrow \mathbb{R}$ satisfies the “no fast-forwarding” assumption if simulating the time dependent Hamiltonian $H(t) = -a(t)\Delta + b(t)f(x)$ for $t \in [t_1, t_2]$ requires $\Omega\left(\Lambda_f \cdot \int_{t_1}^{t_2} |b(t)| dt\right)$ evaluation queries to f , where $\Lambda_f \geq 0$ is some functional of f , independent of t , which satisfies $\Lambda_{cf} \geq |c| \Lambda_f$ for any $c \in \mathbb{R}$.*

We call this assumption “no fast-forwarding” since, with constant $b(t)$, it reduces to the familiar notion for finite-dimensional, time-independent Hamiltonians, and because it is the most reasonable and natural generalization of this notion. It is possible in principle that, by restricting to convex f and for some particular choices of schedules $b(t)$, this assumption may be violated. However, we argue that it is very likely to be true for simulation schemes that work in a black box setting, for several reasons. First, the assumption is satisfied by all general algorithms for time-dependent Hamiltonian simulation of Schrödinger operators that we are aware of, including the one in this paper. Second, this dependence is inherent in the methods for time-dependent Hamiltonian simulation via the interaction picture, that these works are based on. Finally, Assumption 9 reflects intuition in the following (non-rigorous) sense: in each time slice $[t, t + dt)$ where dt is small enough that $b(t)$ is nearly constant, it follows from the no fast forwarding theorem for time independent potentials that the cost of simulating the Schrödinger operator in this time slice is at least $b(t)f dt$. Approximating the whole evolution by these time slices results in the dependence encoded in Assumption 9.

We are now ready to state our lower bounds on the query complexity of convex optimization by simulating QHD.

► **Theorem 10.** *Let f be a potential satisfying the no fast-forwarding assumption (Assumption 9), and the conditions of Problem 1. Then any optimization algorithm that solves Problem 1 via the simulation of QHD under ideal scaling conditions, starting from a real initial state, must make $\Omega(d\Lambda_f GR/\epsilon^2)$ queries to an evaluation oracle for f , if the continuous-time convergence is as predicted by Theorem 14.*

The additional $\sqrt{d}$ factor in our upper bounds (Theorem 6) arises from a bound on Λ_f over the hypercube. Together, our upper and lower bounds almost completely characterize the query complexity of convex optimization with QHD.

We point out that we make no assumptions beyond Lipschitz continuity of the convex functions considered in Problem 1. As such, Problem 1 gives rise to the “nonsmooth” setting in convex optimization theory [65]. It is also common to study settings with more assumptions on the function class. Concretely, it is standard to additionally assume Lipschitz continuity of gradients (this is often referred to as “smoothness” in optimization theory [59]),² and strong convexity. It is known that gradient methods are more efficient when we assume Lipschitz continuous gradients, or Lipschitz continuous gradients and strong convexity. In contrast, our results are equally applicable to all the settings and only require the assumptions corresponding to the nonsmooth setting.

3.2.2 Prospects for quantum speedup for convex optimization

We now compare our query complexity bounds to classical algorithms for unconstrained optimization. Our results are summarized in Table 1.

Comparison to classical algorithms for noiseless convex optimization

The first-order query complexity for the nonsmooth, smooth, and smooth and strongly convex settings is well-understood [20, 59]. When f is G -Lipschitz, one can determine an ϵ -approximate minimizer using $\mathcal{O}((GR/\epsilon)^2)$ queries. If f has L -Lipschitz continuous gradients the bound improves to $\mathcal{O}(LR^2/\epsilon)$, and when f is additionally μ -strongly convex the number of queries is at most $\mathcal{O}(\kappa \log(1/\epsilon))$, where $\kappa := L/\mu$. Note that these bounds are for *unaccelerated* methods. If one employs Nesterov’s acceleration [58], these results improve to $\mathcal{O}(R\sqrt{L}/\epsilon)$ for L -smooth functions, and $\mathcal{O}(\sqrt{\kappa} \log(1/\epsilon))$ for L -smooth, μ -strongly convex functions.

The *zeroth-order* complexity in these settings has only been established more recently. Most of the results along this line concern algorithms that construct gradient estimators by evaluating the function value at a fixed number of random points. The number of permitted points may vary, though the most frequent setups involve 2-point estimators [60, 33, 68] and single-point estimators [38, 3, 10, 11, 22]. Compared to the first-order query complexities, the upper bounds obtained for zeroth-order optimization suffer a $\text{poly}(d)$ slow-down [41]. Specifically, for our setting of nonsmooth optimization, classical zeroth-order methods achieve an oracle complexity $\mathcal{O}(d(GR/\epsilon)^2)$, see, e.g., [33]. In comparison, we show query upper bounds of $\tilde{\mathcal{O}}(d^{1.5}(GR/\epsilon)^2)$, and if the fast forwarding assumption holds and the continuous time convergence rates match those of QHD (Theorem 14), a lower bound of $\tilde{\Omega}(d(GR/\epsilon)^2)$ which matches the classical upper bound. Thus, if the potential corresponding to the objective function satisfies Assumption 9, and if the QHD convergence bound of Theorem 14 is sharp in general, *there is no quantum speedup for zeroth order noiseless convex optimization by simulating QHD.*

Comparison to classical algorithms for noisy convex optimization

While our results for noiseless convex optimization are largely negative, we have a more positive outlook for the case of convex optimization with noisy oracles. By noisy, we assume that evaluations of f are promised to be accurate to precision ϵ_f (see for instance Definition 2). We do not assume that ϵ_f follows some probability distribution, and we will continue to use the term “noisy” for this general setting.

² Note that this definition of smoothness is incomparable to the typical one in analysis, where smoothness implies infinite differentiability.

■ **Table 1** Complexity of algorithms for noisy unconstrained zero-order convex optimization. Admissible noise is an upper bound on ϵ_f , the error in the evaluation of f using a binary oracle (see Definition 2).

Classical Algorithms	Reference	Query complexity	Admissible noise
Stochastic Gradient Estimator	[64]	$\tilde{\mathcal{O}}(d^4(GR/\epsilon)^6)$	$\max\left\{\frac{\epsilon^2}{\sqrt{d}GR}, \frac{\epsilon}{d}\right\}$
Simulated annealing	[11]	$\tilde{\mathcal{O}}(d^{4.5})$	ϵ/d
Quantum Algorithms	Reference	Query complexity	Admissible noise
Quantum simulated annealing	[51]	$\tilde{\mathcal{O}}(d^3)$	ϵ/d
QHD simulation (this paper)	Theorem 6	$\tilde{\mathcal{O}}(d^{1.5}(GR/\epsilon)^2)$	$\tilde{\mathcal{O}}\left(\frac{\epsilon^3}{d^{1.5}G^2R^2}\right)$
Quantum subgradient method	[7]	$\tilde{\mathcal{O}}((GR/\epsilon)^2)$	$\mathcal{O}\left(\frac{\epsilon^5}{d^{4.5}G^4R^4}\right)$

The best known classical algorithms with erroneous oracles are due to Belloni et al. [11] and Ristetski and Li [64]. We discuss both results, since each presents advantages over the other. [11] presents an algorithm based on simulated annealing using a hit and run walk, which requires $\tilde{\mathcal{O}}(d^{4.5} \log(1/\epsilon))$ evaluations of the objective, with error $\mathcal{O}(\epsilon/d)$. On the other hand, [64] presents an algorithm based on a stochastic gradient estimator due to [38]. Their algorithm requires $\mathcal{O}(d^4/\epsilon^6)$ evaluations of the objective with error at most $\mathcal{O}(\max(\epsilon^2/\sqrt{d}, \epsilon/d))$. We note that Ristetski and Li only show the query complexity of their algorithm to be polynomial scaling in d, ϵ but do not determine the polynomial. To make an accurate comparison, we determine this scaling in Appendix C.

In contrast, our algorithm is shown to require $\mathcal{O}(d^{1.5}/\epsilon^2)$ quantum evaluation queries, each with error $\tilde{\mathcal{O}}(\epsilon^3/d^{1.5})$. We note that if $\epsilon = \Theta(d^{-\varrho})$ for some $\varrho > 0$, our query complexities are polynomially lower than the classical algorithms. Up to polylogarithmic factors, the exponent of the speedup over [11] is $1/3 + 4\varrho/9$ and that over [64] is $(3 + 4\varrho)/(8 + 12\varrho)$. Therefore, if ϱ is small, the speedup is super-quadratic over both classical algorithms. We note that the classical algorithms are robust to higher levels of noise than ours, so it is natural to ask if they can be improved if a lower noise threshold is taken into account. We argue in Appendix D that such an improvement is not immediate from existing considerations. It is of course possible that such an improvement can be attained with more effort as there are few applicable lower bounds in general, so it is important to note that our claimed speedups are over *best known* classical algorithms, and not the best possible ones.

Comparison to quantum algorithms for noisy convex optimization

In the quantum model of computation, the best known algorithms in the noisy setting are attributed to Li and Zhang [51] and Augustino et al. [7]. The algorithm in [51] can be viewed as a quantum analogue of the simulated annealing algorithm from Belloni et al. [11] in that the classical hit and run walk is replaced with a quantum hit and run walk originally proposed in [25]. The quantum hit and run walk requires quadratically fewer samples per iteration compared to its classical counterpart, which improves the overall query complexity from $\tilde{\mathcal{O}}(d^{4.5} \log(1/\epsilon))$ to $\tilde{\mathcal{O}}(d^3 \log(1/\epsilon))$. As in [11], their algorithm can tolerate error up to $\mathcal{O}(\epsilon/d)$.

Augustino et al. [7] introduce a zero-order quantum subgradient method, wherein the subgradients are estimated via a generalization of Jordan's algorithm for gradient estimation [45] to subgradient estimation [73, 26]. The quantum subgradient method proposed in [7] achieves a query complexity of $\tilde{O}((GR/\epsilon)^2)$, which is exponentially faster than all known zero-order methods in the problem dimension. This algorithm works in both the noiseless and noisy settings, tolerating function evaluation error up to $\mathcal{O}(\epsilon^5/(d^{4.5}G^4R^4))$ in the latter.

These results and our analysis highlight a tradeoff between the query complexity and the robustness of quantum algorithms for noisy convex optimization. For the high-dimensional regime where $d \gg \text{poly}(G, R, \epsilon^{-1})$, it is clear that the quantum subgradient method is the best performing algorithm with respect to query complexity, but is the least robust in terms of the amount of noise it can tolerate in the function evaluations. Meanwhile, QHD achieves a superior query complexity compared to quantum simulated annealing [51], and is significantly more robust than the quantum subgradient method [7].

Comparison to algorithms for stochastic convex optimization

QHD's combination of the superior query complexity compared to more robust algorithms, and higher noise tolerance compared to faster algorithms, leads to an additional speedup for the problem of *unconstrained stochastic convex optimization*.

► **Problem 11** (Unconstrained Stochastic Convex Optimization). *Let $f: \mathbb{R}^d \rightarrow \mathbb{R}$ be a (star) convex G -Lipschitz continuous function with global minimum $f_\star$ and set of global minimizers $\mathcal{X}_\star$. Suppose f is accessed via a random oracle (quantum or classical) which, given input x , returns $f(x) + \xi$ where ξ has zero mean and bounded variance. Given an input point x_0 such that $\ell_2(x_0, \mathcal{X}_\star) \leq R$, an algorithm solves the stochastic convex optimization problem if it returns a (random) point $\tilde{x} \in \mathbb{R}^d$ such that*

$$\mathbb{E}[f(\tilde{x})] - f_\star \leq \epsilon,$$

where $\mathbb{E}$ is taken over the randomness of the i.i.d. oracle values.

The following result is an immediate consequence of combining the result in Theorem 6 with quantum mean estimation [55].

► **Theorem 12.** *There exists a quantum algorithm that solves Problem 11 using $\tilde{O}(d^3(GR/\epsilon)^5)$ queries to a stochastic evaluation oracle for f in Definition 4.*

A proof is provided in Section 6 of the full version [27], and can be informally summarized as follows. Our algorithm for convex optimization evaluates f up to error $\tilde{O}\left(\frac{\epsilon^3}{d^{3/2}G^2R^2}\right)$. On the other hand, Montanaro [55] proves that if one has access to an oracle O_f for evaluating f up to error ϵ_f , then $\tilde{O}(1/\epsilon_f)$ applications of O_f suffice to estimate $\mathbb{E}[f(x)]$.

Accordingly, the performance of QHD is favorable over the simulated annealing algorithm of Belloni et al. [11] and the quantum algorithms [51, 7] in the zero-order stochastic setting, as highlighted in Table 2. Indeed, QHD achieves a super-quadratic speedup in the dimension over the algorithm in [11], and a sub-quadratic speedup in d over quantum simulated annealing [51]. The query complexity of QHD in this setting also outperforms that of the quantum subgradient method [7] in every parameter. We further remark that the dependence on dimension is close to optimal, due to an $\Omega(d^2(GR/\epsilon)^2)$ lower bound from Shamir [67].

■ **Table 2** Complexity of algorithms for unconstrained zero-order stochastic convex optimization. See Problem 11 for a statement of the problem .

Classical Algorithms	Reference	Query complexity
Simulated annealing	[11]	$\tilde{O}(d^{7.5}(GR/\epsilon)^2)$
Quantum Algorithms	Reference	Query complexity
Quantum simulated annealing	[51]	$\tilde{O}(d^5 GR/\epsilon)$
QHD simulation (this paper)	Theorem 12	$\tilde{O}(d^3(GR/\epsilon)^5)$
Quantum subgradient method	[7]	$\tilde{O}(d^{4.5}(GR/\epsilon)^7)$

► **Remark 13.** As discussed earlier, the assumption that qsamples are available for the noise ξ may not be justified in all settings. In case we have only a classical sampler for ξ , we must resort to classical mean estimation to produce a noisy oracle. In this case, QHD simulation results in a query complexity of $\tilde{O}\left(d^{4.5}\left(\frac{GR}{\epsilon}\right)^8\right)$ whereas the quantum subgradient method incurs a query complexity of $\tilde{O}\left(d^9\left(\frac{GR}{\epsilon}\right)^{12}\right)$. QHD simulation still provides the fastest stochastic convex optimization in this regime, but this regime is much more limited than under qsample access for the noise. The quantum subgradient method no longer provides any speedup.

Related work

Several recent works also consider the use of Hamiltonian dynamics for continuous optimization without adiabaticity. For example, [23] considers several approaches for quantum dynamics-based optimization, including a similar time-dependent Hamiltonian scheme (see Sec. IV.B and V.B therein). While similar, our work has slightly different emphasis and scope. In [23], the authors analyze local convergence in nonconvex optimization by analyzing quadratic approximations to local minima. Although the assumption of convexity is dropped, this analysis tacitly assumes f is twice differentiable and strongly convex in a neighborhood of local optima. In contrast, our work only applies in the convex setting, but makes global convergence claims (via a Lyapunov argument) and only requires f be Lipschitz continuous. Overall, we believe our work and that of [23] both shed light on the potential for quantum-based optimization beyond the computation of gradients of f .

4 Overview of Methods

4.1 Lyapunov convergence

Our first result proves the continuous time convergence of QHD under ideal scaling conditions. The proof is a slight strengthening of [48, Theorem 1]. In addition to the generalization of ideal scaling, we remark on two additional points: firstly, we retain the pre-factor of the error dependence in addition to the convergence rate. This is crucial in order to determine the dimension dependence of the final query complexity (as we do in the next section). Secondly, we show that the convergence requires only the Lipschitz continuity of the potential, not necessarily differentiability.

► **Theorem 14** (QHD convergence for Lipschitz Potentials). *Let $f : \mathbb{R}^d \rightarrow \mathbb{R}$ be convex and Lipschitz continuous, with global minimum $f_\star$ and minimizer $x_\star \in \mathcal{X}_\star$. Let $H_{\text{QHD}}(t)$ be the QHD Hamiltonian with schedules c_t, m_t, ω_t obeying the ideal scaling conditions (Ideal Scaling). Then, for any $\Phi_t \in \text{Dom}(-i\nabla, f(x))$ solving the Schrödinger equation with Hamiltonian H_{QHD} and initial condition Φ_0 ,*

$$\mathbb{E}_{|\Phi_t|^2} f(x) - f_\star \leq \mathcal{E}_0 \omega_t^{-2},$$

where

$$\mathcal{E}_0 := \frac{1}{2} \langle (-im_0^{-1}\nabla + \lambda(x - x_\star))^2 \rangle_{\Phi_0} + \omega_0^2 \langle f(x) - f_\star \rangle_{\Phi_0}.$$

See Appendix B for a proof. Theorem 14 implies that QHD for nonsmooth convex optimization converges at an arbitrarily fast rate in t . To see this, suppose that $r(t)$ is a monotonically increasing function such that $e^{-r(t)}$ is the desired convergence rate. Simply set $\omega_t^2 = e^{r(t)}$ and c_t, m_t according to ideal scaling with equality.

4.2 Hamiltonian simulation

Implementing QHD can be abstracted as the task of simulating $\Phi(x, t) \in \mathcal{H} = L_2([-1/2, 1/2]^d)$ on a Euclidean space, evolved under a kinetic-plus-potential type Hamiltonian

$$i\partial_t \Phi(x, t) = [-a(t)\Delta + V(x, t)]\Phi(x, t) = H(t)\Phi(x, t) \quad (\text{Real Space Schrödinger})$$

from some initial state $\Phi(x, 0) = \Phi_0(x)$ under periodic boundary conditions (PBC). Particularly, the QHD case only considers separable potentials of the form $V(x, t) = b(t)g(x)$. Here g is a natural restriction of some Lipschitz function f to the hypercube $S := [-1/2, 1/2]^d$, defined by $g(x) := f(x_0 + 2Rx)$. For the case of QHD, f is the function we want to optimize.

The first step is to approximately map the spatially-continuous evolution onto a qubit Hamiltonian, which is done via a *spectral method*. Spectral methods begin by imposing some truncation scheme on the solution, such as

$$\Psi(x, t) = \sum_{\mathbf{n} \in \mathcal{N}} \hat{\Psi}_{\mathbf{n}}(t) \chi_{\mathbf{n}}(x) \quad (3)$$

and stipulating some conditions on the modified coefficients $\hat{\Psi}_{\mathbf{n}}$ so that Ψ approximates Φ for the observable of interest. Here $\mathcal{N} := \{-N, -N+1, \dots, N-2, N-1\}^d$, $\chi_{\mathbf{n}}(x) = e^{i2\pi\mathbf{n}\cdot x}$ with $\mathbf{n} \in \mathbb{Z}^d$. We let $\mathcal{H}_N := \text{span}\{\chi_{\mathbf{n}}\}_{\mathbf{n} \in \mathcal{N}}$.

One popular and practical pseudo-spectral method is the so-called *collocation* method, which seeks to replicate the Schrödinger equation exactly on a predefined set of grid points $\mathcal{G} \subset S$. The collocation method seeks a $\Psi \in \mathcal{H}_N$ such that the residual is zero on $\mathcal{G}$:

$$[(i\partial_t - H(t))\Psi](x, t) = 0, \forall x_{\mathbf{j}} \in \mathcal{G}. \quad (4)$$

We take a uniform rectangular grid $\mathcal{G} = \{x_{\mathbf{j}}\}$ such that $x_{\mathbf{j}} = \mathbf{j}/2N$, $\mathbf{j} \in \mathcal{N}$. Then one can show [27, Lemma 4.1] that the solution to (4) can be obtained by simulating the following $d \log(N)$ -qubit quantum Hamiltonian:

$$\hat{H}(t) = a(t)(2\pi)^2 \text{QFT}^{\otimes d} \hat{\mathbf{n}}^2 \text{QFT}^{\dagger \otimes d} + b(t)G = K(t) + V(t), \quad (5)$$

where $G = \sum_{\mathbf{j} \in \mathcal{N}} g(x_{\mathbf{j}}) |\mathbf{j}\rangle \langle \mathbf{j}|$ is the discretized potential. The second component is the time-dependent digital simulation of (5), which is performed utilizing the interaction picture (IP) and rescaled Dyson series approach of [52].

Let U_K stand for the unitary propagators under $K(t)$. Given a digital Schrödinger wavefunction $|\Psi_t\rangle$ generated by $\tilde{H}(t)$, we define the $|\Psi_t^K\rangle$ in the IP as

$$|\Psi_t^K\rangle := U_K^\dagger(t, 0) |\Psi_t\rangle. \quad (6)$$

From here on we drop the 0 argument to U_K . It can be checked that

$$i\partial_t |\Psi_t^K\rangle = H_I(t) |\Psi_t^K\rangle, \quad H_I(t) := U_K^\dagger(t) V(t) U_K(t). \quad (7)$$

By solving (7), the original solution of interest $|\Psi_t\rangle$ can be obtained by applying $U_K(t)$, which is generally presumed efficient, and is indeed so in our case. In fact,

$$U_K(t) = \text{QFT}^{\otimes d} \exp \left\{ -i(2\pi\hat{n})^2 \int_0^t a(\tau) d\tau \right\} \text{QFT}^{\dagger \otimes d} \quad (8)$$

which can be implemented using d QFTs and inverses, along with standard techniques for diagonal phase operations. The rescaled approach also includes a time-reparameterization $t \mapsto \xi(t) = \int_0^t \Lambda b(\tau) d\tau$ applied to $H_I(t)$, which results in

$$\tilde{H}_I(t) = U_K^\dagger(t) (G/\Lambda) U_K(t), \quad (9)$$

where $\Lambda \geq \|g\|_\infty$. This acts as a preconditioning step. Following [52], a truncated Dyson series (Trotterization plus linear combination of unitaries) approach is used to simulate the evolution under $\tilde{H}_I(t)$, which can be done with $\tilde{O}(\Lambda T)$ queries to a $\text{Ham-}T(\tilde{H}_I)$ oracle.

We rigorously show (see proof of [27, Theorem 4.2]) that under the resources specified in Theorem 5, the digital state outputted by our algorithm, $|\Psi_T\rangle$, satisfies:

$$|\langle \Psi_T | G | \Psi_T \rangle - \langle \Phi(\cdot, T) | g | \Phi(\cdot, T) \rangle| \leq \|f\|_\infty \epsilon, \quad (10)$$

where $\Phi(\cdot, T)$ is the continuum solution to (Real Space Schrödinger).

4.3 Combining for full algorithm

We now prove Theorem 6 by combining the Lyapunov convergence results of Section 4.1 with the Hamiltonian simulation results of Section 4.2. These results together lead to a complete resource analysis of discretized QHD. We will actually prove the following stronger theorem.

► **Theorem 15.** *Let $f : \mathbb{R}^d \rightarrow \mathbb{R}$ be a G -Lipschitz convex function with $\mathcal{X}_\star$ its (non-empty, convex) set of global minima. Let there further be a known point $x_0 \in \mathbb{R}^d$ and parameters $R, R_\infty, \Lambda, \Lambda_\infty \in \mathbb{R}_+$ such that for some minimum $x_\star \in \mathcal{X}_\star$ it holds that $\|x_0 - x_\star\| \leq R$, $\|x_0 - x_\star\|_\infty \leq R_\infty$, $f(x_0) - f(x_\star) \leq \Lambda$, and $\max_{x \in B_\infty(x_0, 4R_\infty)} f(x) \leq \Lambda_\infty$.³ There exists then a quantum algorithm (Algorithm 1) that solves Problem 1 using $\tilde{O}\left(\frac{dGR\Lambda_\infty}{\epsilon^2}\right)$ queries to a $\tilde{O}\left(\frac{\epsilon^3}{dGR\Lambda_\infty}\right)$ -accurate evaluation oracle, or $\tilde{O}\left(\left[\frac{dGR\Lambda_\infty}{\epsilon^2}\right]^2 \frac{1}{\epsilon}\right)$ queries to a $\tilde{O}\left(\frac{\epsilon^3}{dGR\Lambda_\infty^2}\right)$ -accurate phase oracle.*

Let us first prove Theorem 6 as a consequence of the above.

³ Not all of $R, R_\infty, \Lambda, \Lambda_\infty$ must be supplied in advance, as bounds on these parameters can be obtained from others and Lipschitz continuity. We state the result in terms of the parameters to accommodate cases where stronger bounds than these are available.

Proof of Thm 6. We note that f satisfies all the conditions of Theorem 15. By the conditions of Problem 1 we are given a point x_0 such that $\|x_0 - x_\star\| \leq R$, hence, $\|x_0 - x_\star\|_\infty \leq R$. Furthermore, since f is G -Lipschitz, we have that $f(x_0) - f(x_\star) \leq GR$ and $\max_{x \in B_\infty(x_0, 4R_\infty)} f(x) \leq GR\sqrt{d}$. We can therefore apply Theorem 15 with $R_\infty = R$, $\Lambda = GR$, and $\Lambda_\infty = GR\sqrt{d}$. The result follows. $\blacktriangleleft$

Algorithm 1 gives pseudocode for our protocol.

■ **Algorithm 1** Convex Optimization via Simulation of Quantum Hamiltonian Descent.

Input: Evaluation oracle O_f for f satisfying conditions of Theorem 15 with parameters $R, R_\infty, \Lambda, \Lambda_\infty$; Schedule c_t, m_t, ω_t satisfying Ideal Scaling with equality, with initial conditions $\omega_0 = 1$, $m_0 = R^{-1}\sqrt{d/(GR + \Lambda)}$ and $\lambda = G^{-1}R^{-2}(\Lambda + GR)^{3/2}$ (See Definitions 7,8 for examples); Target error parameter ϵ .

Output: With probability at least $2/3$, a point $\tilde{x}$ such that $f(\tilde{x}) - f_\star \leq \epsilon, \forall x_\star \in \mathcal{X}_\star$.

Procedure:

- 1: Compute minimum time T_ϵ such that $\omega_{T_\epsilon}^{-2} \leq \epsilon/24$.
- 2: Prepare initial state $|\Psi_0\rangle$ as an appropriately discretized symmetric Gaussian wavefunction

$$\Phi_0(x) = \frac{1}{(2\pi R^2)^{d/4}} e^{-(x-x_0)^2/4R^2}.$$

- 3: Simulate, using Theorem 5, centered at x_0 with simulation accuracy $\epsilon/6\Lambda_\infty$ and $R = 4R_\infty$, the evolution of $|\Psi_0\rangle$ under H_{QHD} as in (QHD), for time T_ϵ . Obtain state $|\Psi_{T_\epsilon}\rangle$.
 - 4: Measure $|\Psi_{T_\epsilon}\rangle$ in the computational basis to obtain $\tilde{x}$; return $\tilde{x}$.
-

We must refer to the Main Version [27] of this paper for a full proof of Theorem 15. Nonetheless, we outline the requisite steps. First, in Lemma 5.1 of [27], we show that for an initial state taken as gaussian wavepacket

$$\Phi_0(x) = (2\pi\sigma^2)^{-d/4} e^{-(x-x_0)^2/4\sigma^2}$$

the Lyapunov constant is bounded as

$$\mathcal{E}_0 \leq \frac{d}{(2m_0\sigma)^2} + 2\lambda^2(d\sigma^2 + R^2) + \omega_0^2(GR + \Lambda).$$

Next, we bound the simulation cost, showing that for ideal scaling parameters, our interaction picture algorithm has complexity

$$\int_0^T c_t m_t \omega_t^2 dt = \frac{m_0 \omega_0^2}{2\lambda} (\omega_T^4 / \omega_0^4 - 1). \quad (11)$$

The result then follows from judicious choice of the initial parameters m_0, ω_0, c_0 .

4.4 Conditional lower bound

In this section, we investigate the dependence of the query complexity derived in previous sections on choice of dynamics within the QHD framework. In particular, is there room to reduce the complexity, by careful optimization of the schedule functions c_t, m_t, ω_t ? We answer this question in the negative, under some reasonable assumptions on the query complexity of Hamiltonian simulation and the initial state, in the above Theorem 10. These lower

bounds, however, are highly conditional on the convergence rate matching the one from the Lyapunov argument. With faster convergence rates than ω_T^{-2} , lower complexities may indeed be achievable.

We use the following helper lemma, which is a specific version of the Heisenberg uncertainty principle in d dimensions.

► **Lemma 16.** *Let $\phi \in \text{Dom}(x_j p_k) \cap \text{Dom}(p_j x_k) \subseteq L_2(\mathbb{R}^d)$ for all $j, k \in [d]$, with $p = -i\nabla$. Suppose $\langle x \rangle_\phi = x_0$ for some $x_0 \in \mathbb{R}$ and suppose $\langle p \rangle_\phi = 0$. Let $\sigma^2 = \langle (x - x_0)^2 \rangle_\phi$ be the variance of ϕ . Then the variance of the momentum can be bounded as $\langle p^2 \rangle_\phi = \Omega(d^2/\sigma^2)$.*

See Appendix for proof. We are now able to give a proof of the conditional lower bound.

Proof of Thm 10. Assume, without loss of generality, that Φ_0 has expected value x_0 in position. Using Eq.(11) and Assumption 9, the total query complexity of simulation is given by

$$\Lambda_f \int_0^T c_t m_t \omega_t^2 dt = \Lambda_f \frac{m_0 \omega_0^2}{2\lambda} ((\omega_T/\omega_0)^4 - 1).$$

Since the convergence in continuous-time follows from Theorem 14, in order to ensure error $\leq \epsilon$, the schedule and simulation time must be concurrently chosen to ensure that $\mathcal{E}_0 \omega_T^{-2} \leq \epsilon$. Note that throughout the argument, $\mathcal{E}_0$ has a possible dependence on problem parameters including d, ϵ . As a consequence, the total query complexity can be lower bounded by

$$\frac{\Lambda_f m_0 \omega_0^2}{2\lambda} \left(\frac{\mathcal{E}_0^2}{\epsilon^2 \omega_0^4} - 1 \right). \quad (12)$$

We recall that for a state satisfying our conditions, and with initial variance $\sigma^2 = O(R^2)$ (all expectations with respect to Φ_0):

$$\begin{aligned} \mathcal{E}_0 &:= \frac{1}{2} \langle (-im_0^{-1}\nabla + \lambda(x - x_\star))^2 \rangle + \omega_0^2 \langle f(x) - f_\star \rangle \\ &= \frac{1}{2} m_0^{-2} \langle p^2 \rangle + \frac{1}{2} \lambda^2 \langle (x - x_0)^2 \rangle + \omega_0^2 \langle f(x) - f_\star \rangle \\ &= \Omega \left(\max \left(m_0^{-2} \langle p^2 \rangle, \lambda^2 \langle (x - x_\star)^2 \rangle, \omega_0^2 \langle f(x) - f_\star \rangle \right) \right) \\ &= \Omega \left(\max \left(d/(m_0 \sigma)^2, \lambda^2 \langle (x - x_\star)^2 \rangle, \omega_0^2 \langle f(x) - f_\star \rangle \right) \right). \end{aligned}$$

Let $l(d, \epsilon)$ be a desired lower bound on $\mathcal{E}_0$. In order for this bound to be satisfied for a general f satisfying only the conditions of Problem 1, we must ensure that:

- $d/(m_0 \sigma)^2 = \mathcal{O}(l(d, \epsilon))$, therefore $m_0 = \Omega(\sigma \sqrt{d/l(d, \epsilon)})$.
- $\lambda^2 \langle (x - x_\star)^2 \rangle = \mathcal{O}(l(d, \epsilon))$, therefore $\lambda^{-1} = \Omega(\sqrt{(R^2 + \sigma^2)/l(d, \epsilon)})$.
- $\omega_0^2 \langle f(x) - f_\star \rangle = \mathcal{O}(l(d, \epsilon))$, therefore $\omega_0^2 = \mathcal{O}(l(d, \epsilon)/GR)$.

Plugging these requirements into Equation 12 we obtain the lower bound

$$\Omega \left(\Lambda_f \cdot \frac{1}{\sigma} \sqrt{\frac{d}{l(d, \epsilon)}} \cdot \frac{\sqrt{R^2 + \sigma^2}}{\sqrt{l(d, \epsilon)}} \cdot \frac{GR}{l(d, \epsilon)} \cdot \left(\frac{l(d, \epsilon)^2}{\epsilon^2} - \frac{l(d, \epsilon)^2}{G^2 R^2} \right) \right).$$

We observe that the second term in the parentheses is dominated when $\epsilon \rightarrow 0$, and the remaining term is minimized when $\sigma = \Theta(R)$. Thus simplifying, we obtain a query complexity lower bound of $\Omega(\sqrt{dGR}\Lambda_f/\epsilon^2)$. ◀

Although the assumption of $\Psi_0(x)$ having real amplitudes appears natural, the reader might wonder whether other choices can avert the lower bound. A hint in this direction is that, domain questions briefly aside, the operator $(p + x)^2$ in the Lyapunov operator $\hat{\mathcal{E}}_0$ has eigenfunction $e^{ix^2/2}$ with eigenvalue 0, independent of d . This suggests there may be some proper, normalized wavefunction which minimizes the Lyapunov constant. This possibility may be worth exploring in future research; however, we argue that such an approach may amount to shifting complexity from time evolution to state preparation. The state in question appears to be highly oscillatory, whereas a real, log-concave initial state such as a Gaussian admits a number of well-studied methods for preparing.

Disclaimer

This paper was prepared for informational purposes by the Global Technology Applied Research center of JPMorgan Chase & Co. This paper is not a product of the Research Department of JPMorgan Chase & Co. or its affiliates. Neither JPMorgan Chase & Co. nor any of its affiliates makes any explicit or implied representation or warranty and none of them accept any liability in connection with this paper, including, without limitation, with respect to the completeness, accuracy, or reliability of the information contained herein and the potential legal, compliance, tax, or accounting effects thereof. This document is not intended as investment research or investment advice, or as a recommendation, offer, or solicitation for the purchase or sale of any security, financial instrument, financial product or service, or to be used in any way for evaluating the merits of participating in any transaction.

References

- 1 Amira Abbas, Andris Ambainis, Brandon Augustino, Andreas Bärtschi, Harry Buhrman, Carleton Coffrin, Giorgio Cortiana, Vedran Dunjko, Daniel J. Egger, Bruce G. Elmegreen, et al. Challenges and opportunities in quantum optimization. *Nature Reviews Physics*, pages 1–18, 2024.
- 2 Alekh Agarwal, Ofer Dekel, and Lin Xiao. Optimal algorithms for online convex optimization with multi-point bandit feedback. In *COLT*, pages 28–40. Citeseer, 2010. URL: <http://colt2010.haifa.il.ibm.com/papers/COLT2010proceedings.pdf#page=36>.
- 3 Alekh Agarwal, Dean P. Foster, Daniel J. Hsu, Sham M. Kakade, and Alexander Rakhlin. Stochastic convex optimization with bandit feedback. *Advances in Neural Information Processing Systems*, 24, 2011.
- 4 Dorit Aharonov, Wim Van Dam, Julia Kempe, Zeph Landau, Seth Lloyd, and Oded Regev. Adiabatic quantum computation is equivalent to standard quantum computation. *SIAM Review*, 50(4):755–787, 2008. doi:10.1137/080734479.
- 5 Tameem Albash and Daniel A. Lidar. Adiabatic quantum computation. *Reviews of Modern Physics*, 90(1):015002, 2018.
- 6 Boris Altshuler, Hari Krovi, and Jérémie Roland. Anderson localization makes adiabatic quantum optimization fail. *Proceedings of the National Academy of Sciences*, 107(28):12446–12450, 2010.
- 7 Brandon Augustino, Dylan Herman, Enrico Fontana, Junhyung Lyle Kim, Jacob Watkins, Shouvanik Chakrabarti, and Marco Pistoia. Fast convex optimization with quantum gradient methods. *arXiv preprint*, 2025. arXiv:2503.17356.
- 8 Brandon Augustino, Jiaqi Leng, Giacomo Nannicini, Tamás Terlaky, and Xiaodi Wu. A quantum central path algorithm for linear optimization. *arXiv preprint arXiv:2311.03977*, 2023. doi:10.48550/arXiv.2311.03977.
- 9 Ryan Babbush, Peter J. Love, and Alán Aspuru-Guzik. Adiabatic quantum simulation of quantum chemistry. *Scientific Reports*, 4(1):6603, 2014.

- 10 Francis Bach and Vianney Perchet. Highly-smooth zero-th order online optimization. In *Conference on Learning Theory*, pages 257–283. PMLR, 2016. URL: <http://proceedings.mlr.press/v49/bach16.html>.
- 11 Alexandre Belloni, Tengyuan Liang, Hariharan Narayanan, and Alexander Rakhlin. Escaping the local minima via simulated annealing: Optimization of approximately convex functions. In *Conference on Learning Theory*, pages 240–265. PMLR, 2015. URL: <http://proceedings.mlr.press/v40/Belloni15.html>.
- 12 Michael Betancourt, Michael I. Jordan, and Ashia C. Wilson. On symplectic optimization. *arXiv preprint*, 2018. [arXiv:1802.03653](https://arxiv.org/abs/1802.03653).
- 13 Sergio Boixo, Troels F. Rønnow, Sergei V. Isakov, Zhihui Wang, David Wecker, Daniel A. Lidar, John M. Martinis, and Matthias Troyer. Evidence for quantum annealing with more than one hundred qubits. *Nature Physics*, 10(3):218–224, 2014.
- 14 Max Born and Vladimir Fock. Beweis des adiabatsatzes. *Zeitschrift für Physik*, 51(3):165–180, 1928.
- 15 Jean Bourgain. On growth of Sobolev norms in linear Schrödinger equations with smooth time dependent potential. *Journal d’Analyse Mathématique*, 77(1):315–348, 1999.
- 16 Stephen P. Boyd and Lieven Vandenberghe. *Convex optimization*. Cambridge University Press, 2004.
- 17 Lucas T. Brady, Christopher L. Baldwin, Aniruddha Bapat, Yaroslav Kharkov, and Alexey V. Gorshkov. Optimal protocols in quantum annealing and quantum approximate optimization algorithm problems. *Physical Review Letters*, 126(7):070505, 2021.
- 18 J. Brooke, David Bitko, Rosenbaum, and Gabriel Aeppli. Quantum annealing of a disordered magnet. *Science*, 284(5415):779–781, 1999.
- 19 J. Brooke, T.F. Rosenbaum, and G. Aeppli. Tunable quantum tunnelling of magnetic domain walls. *Nature*, 413(6856):610–613, 2001.
- 20 Sébastien Bubeck. Convex optimization: Algorithms and complexity. *Foundations and Trends® in Machine Learning*, 8(3-4):231–357, 2015. doi:10.1561/22000000050.
- 21 Sébastien Bubeck and Nicolo Cesa-Bianchi. Regret analysis of stochastic and nonstochastic multi-armed bandit problems. *Foundations and Trends® in Machine Learning*, 5(1):1–122, 2012. doi:10.1561/22000000024.
- 22 Sébastien Bubeck, Ronen Eldan, and Yin Tat Lee. Kernel-based methods for bandit convex optimization. *Journal of the ACM (JACM)*, 68(4):1–35, 2021. doi:10.1145/3453721.
- 23 Ahmet Burak Catli, Sophia Simon, and Nathan Wiebe. Efficient quantum optimization via dynamical simulation. *arXiv preprint*, 2026. URL: <https://arxiv.org/abs/2502.04285>, [arXiv:2502.04285](https://arxiv.org/abs/2502.04285).
- 24 Nicolo Cesa-Bianchi, Alex Conconi, and Claudio Gentile. On the generalization ability of on-line learning algorithms. *IEEE Transactions on Information Theory*, 50(9):2050–2057, 2004. doi:10.1109/TIT.2004.833339.
- 25 Shouvanik Chakrabarti, Andrew M Childs, Shih-Han Hung, Tongyang Li, Chunhao Wang, and Xiaodi Wu. Quantum algorithm for estimating volumes of convex bodies. *ACM Transactions on Quantum Computing*, 4(3):1–60, 2023. doi:10.1145/3588579.
- 26 Shouvanik Chakrabarti, Andrew M. Childs, Tongyang Li, and Xiaodi Wu. Quantum algorithms and lower bounds for convex optimization. *Quantum*, 4:221, 2020. doi:10.22331/Q-2020-01-13-221.
- 27 Shouvanik Chakrabarti, Dylan Herman, Jacob Watkins, Enrico Fontana, Brandon Augustino, Junhyung Lyle Kim, and Marco Pistoia. On speedups for convex optimization via quantum dynamics. *arXiv preprint arXiv:2503.24332*, 2025. doi:10.48550/arXiv.2503.24332.
- 28 Andrew M. Childs, Jiaqi Leng, Tongyang Li, Jin-Peng Liu, and Chenyi Zhang. Quantum simulation of real-space dynamics. *Quantum*, 6:860, 2022. doi:10.22331/Q-2022-11-17-860.
- 29 Alexandre Choquette, Agustin Di Paolo, Panagiotis Kl Barkoutsos, David Sénéchal, Ivano Tavernelli, and Alexandre Blais. Quantum-optimal-control-inspired ansatz for variational quantum algorithms. *Physical Review Research*, 3(2):023092, 2021.

- 30 Alexander M. Dalzell, Sam McArdle, Mario Berta, Przemyslaw Bienias, Chi-Fang Chen, András Gilyén, Connor T. Hann, Michael J. Kastoryano, Emil T. Khabiboulline, Aleksander Kubica, et al. Quantum algorithms: A survey of applications and end-to-end complexities. *arXiv preprint*, 2023. [arXiv:2310.03011](#).
- 31 Olivier Devolder, François Glineur, and Yurii Nesterov. First-order methods of smooth convex optimization with inexact oracle. *Mathematical Programming*, 146(1):37–75, 2014. doi:10.1007/S10107-013-0677-5.
- 32 John C. Duchi. Introductory lectures on stochastic optimization. *The mathematics of data*, 25:99–186, 2018.
- 33 John C. Duchi, Michael I. Jordan, Martin J. Wainwright, and Andre Wibisono. Optimal rates for zero-order convex optimization: The power of two function evaluations. *IEEE Transactions on Information Theory*, 61(5):2788–2806, 2015. doi:10.1109/TIT.2015.2409256.
- 34 Alexandre d’Aspremont, Damien Scieur, and Adrien Taylor. Acceleration methods. *Foundations and Trends in Optimization*, 5(1-2):1–245, 2021. doi:10.1561/24000000036.
- 35 Edward Farhi, Jeffrey Goldstone, and Sam Gutmann. A quantum approximate optimization algorithm, 2014. [arXiv:1411.4028](#).
- 36 Edward Farhi, Jeffrey Goldstone, Sam Gutmann, Joshua Lapan, Andrew Lundgren, and Daniel Preda. A quantum adiabatic evolution algorithm applied to random instances of an NP-complete problem. *Science*, 292(5516):472–475, 2001.
- 37 Edward Farhi, Jeffrey Goldstone, Sam Gutmann, and Michael Sipser. Quantum computation by adiabatic evolution, 2000. [arXiv:quant-ph/0001106](#).
- 38 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*, SODA ’05, pages 385–394, USA, 2005. Society for Industrial and Applied Mathematics. URL: <http://dl.acm.org/citation.cfm?id=1070432.1070486>.
- 39 Ankit Garg, Robin Kothari, Praneeth Netrapalli, and Suhail Sherif. Near-optimal lower bounds for convex optimization for all orders of smoothness. *Advances in Neural Information Processing Systems*, 34:29874–29884, 2021. URL: <https://proceedings.neurips.cc/paper/2021/hash/fa6c94460e902005a0b660266190c8ba-Abstract.html>.
- 40 Ankit Garg, Robin Kothari, Praneeth Netrapalli, and Suhail Sherif. No Quantum Speedup over Gradient Descent for Non-Smooth Convex Optimization. In James R. Lee, editor, *12th Innovations in Theoretical Computer Science Conference (ITCS 2021)*, volume 185 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 53:1–53:20, Dagstuhl, Germany, 2021. Schloss Dagstuhl – Leibniz-Zentrum für Informatik. doi:10.4230/LIPIcs.ITCS.2021.53.
- 41 Alexander V. Gasnikov, Darina Dvinskikh, Pavel Dvurechensky, Eduard Gorbunov, Aleksandr Beznosikov, and Alexander Lobanov. Randomized gradient-free methods in convex optimization. In *Encyclopedia of Optimization*, pages 1–15. Springer, 2023.
- 42 Ernst Hairer, Christian Lubich, and Gerhard Wanner. Structure-preserving algorithms for ordinary differential equations. *Geometric numerical integration*, 31, 2006.
- 43 Brian C. Hall. *Quantum theory for mathematicians*. Springer, 2013.
- 44 Samuel B Hopkins. Mean estimation with sub-gaussian rates in polynomial time. *The Annals of Statistics*, 48(2):1193–1213, 2020.
- 45 Stephen P. Jordan. Fast quantum algorithm for numerical gradient estimation. *Physical Review Letters*, 95(5):050501, 2005.
- 46 Tadashi Kadowaki and Hidetoshi Nishimori. Quantum annealing in the transverse Ising model. *Physical Review E*, 58(5):5355, 1998.
- 47 Walid Krichene, Alexandre Bayen, and Peter L. Bartlett. Accelerated mirror descent in continuous and discrete time. *Advances in Neural Information Processing Systems*, 28, 2015.
- 48 Jiaqi Leng, Ethan Hickman, Joseph Li, and Xiaodi Wu. Quantum Hamiltonian Descent, 2023. [arXiv:2303.01471](#). doi:10.48550/arXiv.2303.01471.
- 49 Jiaqi Leng, Yufan Zheng, Zhiyuan Jia, Chaoyue Zhao, Yuxiang Peng, and Xiaodi Wu. Quantum Hamiltonian Descent for Non-smooth Optimization", 2025. [arXiv:2503.15878](#).

- 50 Jiaqi Leng, Yufan Zheng, and Xiaodi Wu. A quantum-classical performance separation in nonconvex optimization, 2023. [arXiv:2311.00811](https://arxiv.org/abs/2311.00811). doi:10.48550/arXiv.2311.00811.
- 51 Tongyang Li and Ruizhe Zhang. Quantum speedups of optimizing approximately convex functions with applications to logarithmic regret stochastic convex bandits. *Advances in Neural Information Processing Systems*, 35:3152–3164, 2022.
- 52 Guang Hao Low and Nathan Wiebe. Hamiltonian simulation in the interaction picture. *arXiv preprint*, 2018. [arXiv:1805.00675](https://arxiv.org/abs/1805.00675).
- 53 Alicia B. Magann, Christian Arenz, Matthew D. Grace, Tak-San Ho, Robert L. Kosut, Jarrod R. McClean, Herschel A. Rabitz, and Mohan Sarovar. From pulses to circuits and back again: A quantum optimal control perspective on variational quantum algorithms. *PRX Quantum*, 2(1):010101, 2021.
- 54 Oinam Romesh Meitei, Bryan T Gard, George S Barron, David P Pappas, Sophia E Economou, Edwin Barnes, and Nicholas J Mayhall. Gate-free state preparation for fast variational quantum eigensolver simulations. *npj Quantum Information*, 7(1):155, 2021.
- 55 Ashley Montanaro. Quantum speedup of Monte Carlo methods. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 471(2181):20150301, 2015.
- 56 Arkadi Nemirovski, Anatoli Juditsky, Guanghui Lan, and Alexander Shapiro. Robust stochastic approximation approach to stochastic programming. *SIAM Journal on Optimization*, 19(4):1574–1609, 2009. doi:10.1137/070704277.
- 57 Arkadi Semenovič Nemirovskii and David Borisovich Yudin. *Problem complexity and method efficiency in optimization*. Wiley-Interscience, 1983.
- 58 Yurii Nesterov. A method for solving the convex programming problem with convergence rate $\mathcal{O}(1/k^2)$. In *Dokl akad nauk Sssr*, volume 269, page 543, 1983.
- 59 Yurii Nesterov. *Lectures on convex optimization*, volume 137. Springer, 2018.
- 60 Yurii Nesterov and Vladimir Spokoiny. Random gradient-free minimization of convex functions. *Foundations of Computational Mathematics*, 17(2):527–566, 2017. doi:10.1007/S10208-015-9296-2.
- 61 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. *Nature Communications*, 5(1):4213, 2014.
- 62 Boris T. Polyak. Some methods of speeding up the convergence of iteration methods. *USSR Computational Mathematics and Mathematical Physics*, 4(5):1–17, 1964.
- 63 Ben W. Reichardt. The quantum adiabatic optimization algorithm and local minima. In *Proceedings of the thirty-sixth annual ACM Symposium on Theory of Computing*, pages 502–510, 2004. doi:10.1145/1007352.1007428.
- 64 Andrej Risteski and Yuanzhi Li. Algorithms and matching lower bounds for approximately-convex optimization. *Advances in Neural Information Processing Systems*, 29, 2016.
- 65 Terry Rockafellar. *Nonsmooth optimization*, 1994.
- 66 Walter Rudin. *Real and complex analysis*. McGraw-Hill, Inc., 1987.
- 67 Ohad Shamir. On the complexity of bandit and derivative-free stochastic convex optimization. In *Conference on learning theory*, pages 3–24. PMLR, 2013. URL: <http://proceedings.mlr.press/v30/Shamir13.html>.
- 68 Ohad Shamir. An optimal algorithm for bandit and zero-order convex optimization with two-point feedback. *Journal of Machine Learning Research*, 18(52):1–11, 2017. URL: <https://jmlr.org/papers/v18/16-632.html>.
- 69 Alexander Shapiro, Darinka Dentcheva, and Andrzej Ruszczyński. *Lectures on stochastic programming: modeling and theory*. SIAM, 2021.
- 70 Aaron Sidford and Chenyi Zhang. Quantum speedups for stochastic optimization. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine, editors, *Advances in Neural Information Processing Systems*, volume 36, pages 35300–35330. Curran Associates, Inc., 2023. URL: https://proceedings.neurips.cc/paper_files/paper/2023/file/6ed9931d6e1fb6a85efa1b2c014a47e1-Paper-Conference.pdf.

- 71 Yaron Singer and Jan Vondrák. Information-theoretic lower bounds for convex optimization with erroneous oracles. *Advances in Neural Information Processing Systems*, 28, 2015.
- 72 Weijie Su, Stephen Boyd, and Emmanuel J. Candes. A differential equation for modeling Nesterov’s accelerated gradient method: Theory and insights. *Journal of Machine Learning Research*, 17(153):1–43, 2016. URL: <https://jmlr.org/papers/v17/15-084.html>.
- 73 Joran van Apeldoorn, András Gilyén, Sander Gribling, and Ronald de Wolf. Convex optimization using quantum oracles. *Quantum*, 4:220, 2020. doi:10.22331/Q-2020-01-13-220.
- 74 Wim Van Dam, Michele Mosca, and Umesh Vazirani. How powerful is adiabatic quantum computation? In *Proceedings 42nd IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 279–287. IEEE, 2001. doi:10.1109/SFCS.2001.959902.
- 75 Lorenzo Campos Venuti, Domenico D’Alessandro, and Daniel A. Lidar. Optimal control for quantum optimization of closed and open systems. *Physical Review Applied*, 16(5):054023, 2021.
- 76 Andre Wibisono, Ashia C. Wilson, and Michael I. Jordan. A variational perspective on accelerated methods in optimization. *Proceedings of the National Academy of Sciences*, 113(47):E7351–E7358, 2016.
- 77 Zhi-Cheng Yang, Armin Rahmani, Alireza Shabani, Hartmut Neven, and Claudio Chamon. Optimizing variational quantum algorithms using Pontryagin’s minimum principle. *Physical Review X*, 7(2):021027, 2017.
- 78 Chenyi Zhang, Jiaqi Leng, and Tongyang Li. Quantum algorithms for escaping from saddle points. *Quantum*, 5:529, August 2021. doi:10.22331/q-2021-08-20-529.

A Additional Details for Hamiltonian Simulation

Simulation by truncated Dyson series (previously mentioned in Section 4.2) requires a certain block encoding of the interaction picture Hamiltonian $\tilde{H}_I$, which we now define.

► **Definition 17** (Ham-T Oracle [52]). *Let $H(t)$ be a time-dependent Hamiltonian such that for some $\Lambda \in \mathbb{R}_+$, $\|H(t)\| \leq \Lambda$ for all $t \in [0, T]$. Let $\{t_j\}_{j=1}^M \subset [0, T]$ be a discrete set of points. The associated Ham-T oracle is a unitary block encoding such that*

$$(\langle 0|_a \otimes I) \text{Ham-T}(H) (|0\rangle_a \otimes I) = C(H)/\Lambda,$$

where

$$C(H) := \sum_{j=0}^M |j\rangle \langle j| \otimes H(t_j)$$

is the time-controlled Hamiltonian and a is an auxiliary register.

Our Hamiltonian (9) admits such a block encoding almost immediately.

► **Lemma 18** (Ham-T Construction, Main Version [27] Lemma 4.2). *Let $\tilde{H}_I$ be the interaction picture Hamiltonian of equation (9) with $G|\mathbf{j}\rangle := g(x_{\mathbf{j}})|\mathbf{j}\rangle$, and let $\Lambda \geq \|g\|_\infty$ be a promised upper bound. Then there exists an ϵ_H -accurate block encoding U_g of G , such that*

$$\left\| \text{Ham-T}(\tilde{H}_I) - C(U_K)^\dagger U_g C(U_K) \right\| \leq \epsilon_H, \quad (13)$$

with probability of failure at most δ , and where $\text{Ham-T}(\tilde{H}_I)$ is given in Definition 17. The total costs are as follows.

- **Queries:** Q_f queries to a quantum oracle O_f , where

$$Q_f = \begin{cases} 2 & \text{if } O_f \text{ is a } (\Lambda\epsilon_H)\text{-accurate binary oracle (Definition 2),} \\ \mathcal{O}(\log(1/\delta)/\epsilon_H) & \text{if } O_f \text{ is a } \epsilon_H\text{-accurate controlled phase oracle (Definition 3).} \end{cases} \quad (14)$$

- **Qubits:** $\mathcal{O}(\log(1/\epsilon_H) + \log(M) + \log(1/\delta))$ auxiliary qubits.
- **Gates:** $\mathcal{O}(\log(1/\delta) \log^2(1/\epsilon_H) + d \log(N)^2 + d \log(N) \log(M))$ additional elementary quantum gates.

The overall resource count for simulating $\tilde{H}_I$ (Main Version [27, Theorem 4.3]) then follows from plugging in the various parameters of our setting into the results of [52] and accounting for the cost to implement $\text{Ham-T}(\tilde{H}_I)$. This is roughly $\tilde{\mathcal{O}}(T\Lambda Q_f)$ queries to the function quantum oracle O_f , as long as $\epsilon_f^{-1} = \tilde{\mathcal{O}}\left(\frac{\|b\|_1}{\epsilon}\right)$ (Main Version [27, Theorem 4.4]) under overall desired error ϵ .

We also sketch the procedure for analyzing the discretization error, i.e. how does the digital state produced by simulating $\hat{H}(t)$ compare to Real Space Schrödinger. In Main Version [27, Corollary 4.1], we show that digital output of our algorithm takes the form

$$|\Psi_t\rangle = \frac{1}{\sqrt{(2N)^d}} \sum_{\mathbf{j} \in \mathcal{N}} \Psi_{\mathbf{j}}(t) |\mathbf{j}\rangle.$$

We can use this to define a continuous wave function over S , using the notion of *Fourier interpolant*:

$$\Psi(x, t) = \sum_{\mathbf{n} \in \mathcal{N}} \left[\frac{1}{(2N)^d} \sum_{\mathbf{j} \in \mathcal{N}} \bar{\chi}_{\mathbf{n}}(x_{\mathbf{j}}) \Psi_{\mathbf{j}}(t) \right] \chi_{\mathbf{n}}(x) \in \mathcal{H}_N. \quad (15)$$

In Main Version [27, Theorem 4.6], we show that the d -dimensional pseudospectral method applied to (Real Space Schrödinger) with N^d grid points satisfies:

$$\|\Phi(T) - \Psi(T)\| \lesssim \|a\|_1 \cdot \max_{t \in [0, T]} \left[\left(\frac{\pi}{4} \right)^{d/4} \frac{2|\Phi(t)|_{\mathcal{H}^{m+2}} + |\Phi(t)|_{\mathcal{H}^m}}{\sqrt{(m-d/2)\Gamma(d/2)N^m}} + \frac{2|\Phi(t)|_{\mathcal{H}^3} + |\Phi(t)|_{\mathcal{H}^1}}{N} \right],$$

where $|\cdot|_{\mathcal{H}^s}$ is the s -th Sobolev seminorm defined as $|\phi|_{\mathcal{H}^s}^2 := \sum_{\alpha \in \mathbb{N}^n: \|\alpha\|_1 = s} \langle \partial^\alpha \phi, \partial^\alpha \phi \rangle$. This result is obtained by bounding both the d -dimensional Fourier truncation and aliasing errors encountered by mapping to a finite grid. As done in Main Version [27, Theorem 4.9] this error bound can be used to show (10). Extending the results of [15], we derive (Main Version [27, Lemma C.9]) a bound on the growth of the s -th Sobolev seminorm over t in terms of those of the potential g :

$$|\Phi(T)|_{\mathcal{H}^s} \leq 2|\Phi(0)|_{\mathcal{H}^s} + \left(\frac{2 \sum_{k=1}^s \binom{s}{k} |g|_{\mathcal{H}^k} \|b\|_1}{s} \right)^s.$$

Under the assumption that f is G -Lipschitz, convex, and satisfies PBC, we (show Main Version [27, Lemma C.11]) that there exists a convex function very close to g , called g_σ , whose Sobolev seminorms are well-behaved:

$$\|g - g_\sigma\|_\infty = \mathcal{O}\left(\sqrt{d}GR\sigma\right), \quad |g_\sigma|_{\mathcal{H}^m} = \mathcal{O}\left(\|g\|_\infty \left(\frac{Cd}{\sigma}\right)^{3d}\right), \quad m \leq d + \mathcal{O}(1),$$

for some constant $C > 0$. This uses the notion of the mollification of a function. If g does not satisfy PBC, then we can use an additional, yet mild, assumption (Main Version [27, Definition 4.5]) to make this work. Due to the linearity of quantum simulation, the $\|g - g_\sigma\|_\infty$ term only causes a proportional blow-up in the overall error (Main Version [27, Lemma C.1]). With an appropriately chosen σ , this all suffices to ensure that the total number of qubits $d \log(N)$ remains $\mathcal{O}(d^2 \text{polylog}(1/\epsilon, \|a\|_1, \|b\|_1))$ for (10) to hold.

Our work is not the first to analyze quantum algorithms for real space simulation. Indeed, the pseudo-spectral framework of Section 4.2 was previously considered in [28], which made a significant step towards a rigorous characterization of real-space quantum Hamiltonian simulation. The framework and analysis has been subsequently used for various applications [78, 48, 50, 49, 8]. However, we argue that existing literature leaves out a critical piece of analysis that is essential to obtain a complete end-to-end analysis of quantum real-space algorithms is absent.

The output of the pseudo-spectral method is *neither a truncated Fourier series of the true solution nor a trigonometric interpolation*. However, [28] provides only a proof of the required grid spacing to ensure low Fourier truncation error, which does not by itself rigorously imply good pseudo-spectral method performance. For the truncation error bound, the results of [28] are given in terms of a quantity g' , which depends on properties of the solution wavefunction and are left unbounded. Note that no such quantity appears in our main simulation result (Theorem 5). As mentioned above, this is essential, since we demand a simulation theorem that is applicable to black-box potentials. The Fourier truncation result of [28] was only proven for one-dimension, which we generalized to multiple dimensions.

B Select Proofs of Results

Here we provide some selected proofs for results stated in the main body of this paper, chosen according to importance and simplicity. As usual, see Main Version [27] for all results and proofs.

Proof of Thm 14. As in [48], we proceed with a quantum Lyapunov argument. Let $p := -i\nabla$ and

$$\widehat{\mathcal{E}}_t := \frac{1}{2}(p/m_t + \lambda(x - x_\star))^2 + \omega_t^2(f(x) - f_\star). \quad (16)$$

Observe that $\widehat{\mathcal{E}}_t$ is the sum of two positive semi-definite operators (on the appropriate subdomain), hence positive. Define $\mathcal{E}_t := \langle \widehat{\mathcal{E}}_t \rangle_t$, where $\langle \cdot \rangle_t$ is shorthand for $\langle \cdot \rangle_{\Phi_t}$ hereon. Then $\mathcal{E}_t \in C^1$ is a non-negative function. We wish to prove that $\mathcal{E}_t$ is a *Lyapunov function*, meaning $\partial_t \mathcal{E}_t \leq 0$. This would imply

$$\omega_t^2 \langle f(x) - f_\star \rangle_t \leq \mathcal{E}_T \leq \mathcal{E}_0$$

which amounts to the theorem statement.

Toward this end, we note the well-known fact that

$$\partial_t \langle \widehat{\mathcal{E}}_t \rangle_t = i \langle [H_{\text{QHD}}(t), \widehat{\mathcal{E}}_t] \rangle_t + \langle \partial_t \widehat{\mathcal{E}}_t \rangle_t.$$

For the first term on the right, we evaluate the commutator as

$$i[H_{\text{QHD}}(t), \widehat{\mathcal{E}}_t] = \frac{i}{4} \frac{c_t}{m_t} [p^2, P_t^2] + \frac{i}{2} \frac{c_t \omega_t^2}{m_t} [p^2, f] + \frac{i}{2} c_t m_t \omega_t^2 [f, P_t^2]$$

where $P_t := p/m_t + \lambda(x - x_*)$. We handle each term on the right separately, making frequent use of the derivation property of the commutator: $[AB, C] = [A, C]B + A[B, C]$. We also use

$$[p, f] = -i\nabla f$$

where ∇f is understood as the gradient of f where it exists, and 0 elsewhere. By Rademacher's theorem, the "elsewhere" set is of measure zero [66]. The first term is

$$\frac{i}{4} \frac{c_t}{m_t} [p^2, P_t^2] = \lambda \frac{c_t}{m_t} \left(\frac{p^2}{m_t} + \lambda \Re(p \cdot \Delta x) \right)$$

where $\Re(A)$ denotes the Hermitian part and $\Delta x = x - x_*$. Next,

$$\frac{i}{2} \frac{c_t \omega_t^2}{m_t} [p^2, f] = \frac{1}{2} \frac{c_t \omega_t^2}{m_t} (p \cdot \nabla f + \nabla f \cdot p).$$

Finally,

$$\frac{i}{2} c_t m_t \omega_t^2 [f, P_t^2] = -\frac{1}{2} \frac{c_t \omega_t^2}{m_t} (p \cdot \nabla f + \nabla f \cdot p) - \lambda c_t \omega_t^2 \Delta x \cdot \nabla f.$$

The last two are easiest to combine, as there is a cancellation of one of the terms. Adding all the terms together one obtains

$$i \langle [H(t), \widehat{\mathcal{E}}_t] \rangle_t = \lambda c_t \frac{\langle p^2 \rangle_t}{m_t^2} + \lambda^2 c_t \frac{\langle \Re(\Delta x \cdot p) \rangle_t}{m_t} - \lambda c_t \omega_t^2 \langle \Delta x \cdot \nabla f \rangle_t.$$

Meanwhile, the partial derivative term is given by

$$\langle \partial_t \widehat{\mathcal{E}}_t \rangle_t = -\frac{\dot{m}_t}{m_t} \frac{\langle p^2 \rangle_t}{m_t^2} - \lambda \frac{\dot{m}_t}{m_t} \frac{\langle \Re(\Delta x \cdot p) \rangle_t}{m_t} + 2\dot{\omega}_t \omega_t \langle f - f_* \rangle_t.$$

Adding the previous two lines term by term, one can see that the $\langle p^2 \rangle_t$ and $\langle \Re(\Delta x \cdot p) \rangle_t$ terms cancel if c_t and m_t are chosen such that

$$\lambda c_t - \frac{\dot{m}_t}{m_t} = 0$$

which amounts to the first ideal scaling condition. Taking this from here on, we have

$$\begin{aligned} \partial_t \mathcal{E}_t &= 2\dot{\omega}_t \omega_t \langle f - f_* \rangle_t - \lambda c_t \omega_t^2 \langle \Delta x \cdot \nabla f \rangle_t \\ &= \omega_t^2 \left(2 \frac{\dot{\omega}_t}{\omega_t} \mathbb{E}_{x \sim |\Phi_t|^2} [f - f_*] - \lambda c_t \mathbb{E}_{x \sim |\Phi_t|^2} [\Delta x \cdot \nabla f] \right). \end{aligned} \quad (17)$$

Next, we make use of the (star-)convexity of the function f , which implies that

$$0 \leq f(x) - f_* \leq \Delta x \cdot \nabla f(x) \quad (18)$$

almost everywhere. Hence the expression holds unconditionally under the expectation value, leaving

$$\partial_t \mathcal{E}_t \leq \omega_t^2 \mathbb{E}_{x \sim |\Phi_t|^2} [\Delta x \cdot \nabla f] \left(2 \frac{\dot{\omega}_t}{\omega_t} - \lambda c_t \right). \quad (19)$$

By (18), the expectation value is nonnegative, and thus $\partial_t \mathcal{E}$ is non-positive provided that the second ideal scaling condition holds. $\blacktriangleleft$

Proof. Observe that $\langle (x - x_0)^2 \rangle_\phi = \sum_{i=1}^d \langle (x - x_0)_i^2 \rangle_\phi$ and $\langle p^2 \rangle_\phi = \sum_{i=1}^d \langle p_i^2 \rangle_\phi$. Using the Robertson form of the Heisenberg Uncertainty Principle in one dimension [43], we notice that $\langle p_i^2 \rangle_\phi \geq \varsigma / \langle (x - x_0)_i^2 \rangle_\phi$ where ς is an absolute constant.

We then note that,

$$\begin{aligned} \langle p^2 \rangle_\phi \cdot \langle (x - x_0)^2 \rangle_\phi &= \left(\sum_{i=1}^d \langle p_i^2 \rangle_\phi \right) \cdot \left(\sum_{i=1}^d \langle (x - x_0)_i^2 \rangle_\phi \right), \\ &\geq \varsigma \left(\sum_{i=1}^d \langle (x - x_0)_i^{-2} \rangle_\phi \right) \cdot \left(\sum_{i=1}^d \langle (x - x_0)_i^2 \rangle_\phi \right) \geq \varsigma d^2, \end{aligned}$$

where the last inequality arises from the well known AM-HM inequality. The result follows. $\blacktriangleleft$

C Query Complexity of the Li-Risteski Algorithm

As a comparison point, let's consider the classical algorithm from Li and Risteski [64] for approximately convex optimization. To our knowledge, this is the most competitive classical algorithm in our setting of noisy zeroth-order convex optimization. Although Li and Risteski only demonstrates polynomial scaling, here we supply the detailed scaling to obtain the first row of Table 1. See [64, Algorithm 1] for a description of the algorithm. Since we are interested in the case of unconstrained optimization, we will make this simplification while analyzing the query complexity. We will scale the input and output space such that $G = 1$ and $R = 1$, as a consequence we must perform the optimization up to error $\varepsilon := \epsilon / GR$.

The algorithm executes $T = \Theta(d^2/\varepsilon^4)$ iterations. The primary computational bottleneck in each step is estimating the expectation

$$f_t = \mathbb{E}_{w \in S^d} \left[\frac{d}{r} f(x_t + rw) w \right] = \mathbb{E}_{w \in S^d} \left[\frac{d}{r} (f(x_t + rw) - f(x_t)) w \right],$$

within accuracy $\Theta(\varepsilon)$ in ℓ_2 norm. Here x_t is a point in $B_2(x_0, R)$, $r = \Theta(\varepsilon)$ and w is uniformly sampled from the d -dimensional hypersphere S_d . In the worst case,

$$\Sigma = \text{Var}_{w \in S^d} \left[\frac{d}{r} (f(x_t + rw) - f(x_t)) w \right] \leq d^2 \cdot \text{Var}_{w \in S^d} [\|w\| w] = d^2 \text{Var}_{w \in S^d} [w] = d \mathbb{I}_d,$$

where the inequality from symmetry and the Lipschitz continuity of f . Using results from Hopkins [44] on optimal mean estimation in Euclidean norm, we observe that given some fixed desired success probability, the ℓ_2 norm accuracy of the estimated mean decays with the number of samples N as

$$\mathcal{O} \left(\sqrt{\text{Tr}(\Sigma)/N} + \sqrt{\|\Sigma\|/N} \right) = \mathcal{O}(d/\sqrt{N}).$$

In order to obtain $\mathcal{O}(\varepsilon)$ error in ℓ_2 norm, the number of samples taken per round must be $\sim d^2/\varepsilon^2$. Each sample requires a noisy evaluation of the function f , resulting in a final query complexity of $\mathcal{O}(d^4/\varepsilon^6)$.

Because their algorithm tolerates high evaluation noise, it is natural to wonder if refinements of the complexity are possible assuming lower noise levels. We essentially rule out this possibility in Appendix D. In summary, our algorithm appears to offer genuine speedups over known classical algorithms for stochastic convex optimization.

D Prospects for Reducing Classical Complexity

Here we discuss whether leading classical algorithms for noisy zeroth-order convex optimization can be improved if the noise tolerance is allowed to be lowered to that of the present quantum algorithm. It is not apparent to us that such an improvement is possible.

First, we note that the query complexity of the algorithm of Belloni et al [11] is based on the analysis of a classical hit-and-run walk. The bottleneck is the classical complexity of the algorithm and is unchanged even if the queries are chosen to be noiseless. The robustness of the algorithm arises from the natural robustness of walk based algorithms, and the lack of any gradient estimation. Thus reducing the noise tolerance does not lead to any improvements in oracle complexity.

Next, consider the algorithm of Ristetski and Li [64, Algorithm 1], which is based on a gradient estimator. It is evident by inspection of the algorithm that the query complexity is determined by a parameter r that is the radius of smoothing. As usual, we rescale the problem by GR , letting $\varepsilon = \epsilon/GR$ and $\varepsilon_f = \epsilon_f/GR$. A calculation borrowed from [38] shows that the choice for r must satisfy

$$f(x) - f_\star - 2r \geq \frac{\varepsilon_f \sqrt{d}}{r} \text{ whenever } f(x) - f_\star \geq \varepsilon.$$

Plugging in the 2nd inequality to the first, it suffices that r lies in the range

$$r = \frac{\varepsilon}{4} \left(1 \pm \sqrt{1 - 8\varepsilon_f \sqrt{d}/\varepsilon^2} \right). \quad (20)$$

Assuming a noise $\varepsilon_f = \mathcal{O}(\varepsilon^3/d)$, we choose the largest r consistent with these bounds, which minimizes sample complexity (see proof of Thm 3.2 [64]). This is $r = \Theta(\varepsilon)$, the same choice of r as the original work. Thus, following through the rest of the analysis (which is independent of the noise parameter) leads to no improvements.

Quantum Merlin-Arthur with an Internally Separable Proof

Roозbeh Bassirian ✉ 

University of Chicago, IL, USA

Bill Fefferman ✉ 

University of Chicago, IL, USA

Itai Leigh ✉ 

Tel Aviv University, Israel

Kunal Marwaha ✉ 

University of Chicago, IL, USA

Pei Wu ✉ 

Pennsylvania State University, University Park, PA, USA

Abstract

While the role of entanglement in quantum proof systems has been extensively studied, the computational power of unentanglement remains poorly understood. Since entanglement admits many inequivalent multipartite structures, it is natural to ask how more fine-grained structural promises affect computational power. In this work we investigate a mild promise: each proof is internally separable, meaning that after tracing out one register, a designated constant-size subsystem is separable from the rest – even though the overall proof may still be entangled across every bipartition. We prove a qualitative jump from one proof to two: with one internally separable proof, the resulting class is contained in EXP (even allowing an inverse-exponential completeness–soundness gap), whereas with two unentangled internally separable proofs, the class equals NEXP at constant gap. Notably, in the NEXP construction, the second proof is used solely to implement a SWAP-based purity test.

2012 ACM Subject Classification Theory of computation → Quantum complexity theory; Theory of computation → Interactive proof systems; Theory of computation → Complexity classes; Theory of computation → Problems, reductions and completeness; Theory of computation → Quantum complexity theory

Keywords and phrases entanglement structures, unentanglement, quantum complexity, QMA(2), NEXP

Digital Object Identifier 10.4230/LIPIcs.TQC.2026.5

Related Version *Full Version:* <https://arxiv.org/abs/2410.19152>

Funding This work was done in part while a subset of the authors were visiting the Simons Institute for the Theory of Computing, supported by NSF QLCI Grant No. 2016245. BF, RB, and KM acknowledge support from AFOSR (FA9550-21-1-0008). This material is based upon work partially supported by the National Science Foundation under Grant CCF-2044923 (CAREER), by the U.S. Department of Energy, Office of Science, National Quantum Information Science Research Centers (Q-NEXT) and by the DOE QuantISED grant DE-SC0020360.

Any views, opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation, European Union, or the European Research Council Executive Agency. Neither the European Union nor any granting authority can be held responsible for them.

Itai Leigh: co-funded by the European Union (ERC, EACTP, 101142020), the Israel Science Foundation (grant number 514/20) and the Len Blavatnik and the Blavatnik Family Foundation.

Kunal Marwaha: acknowledges support from the National Science Foundation Graduate Research Fellowship Program under Grant No. DGE-1746045.



© Roозbeh Bassirian, Bill Fefferman, Itai Leigh, Kunal Marwaha, and Pei Wu;
licensed under Creative Commons License CC-BY 4.0

21st Conference on the Theory of Quantum Computation, Communication and Cryptography (TQC 2026).

Editors: Rotem Arnon and Aram W. Harrow; Article No. 5; pp. 5:1–5:13

Leibniz International Proceedings in Informatics



LIPICs Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

Acknowledgements KM thanks Tyler Chen, Christopher Musco, and Apoorv Vikram Singh for helpful discussions on convex optimization. RB, KM, and IL thank the Institute for Advanced Study for organizing the summer school where they met and began collaborating. IL and PW thank Thomas Vidick for suggesting them to talk.

1 Overview

Entanglement is a fundamental property of quantum mechanics. One approach to understanding this phenomenon is through the lens of computational complexity, probing the power of quantum proof systems with and without entanglement. A canonical example is the relationship between QMA and QMA(2). Here QMA is the quantum analogue of NP (see [30] for a survey): a quantum polynomial-time verifier (Arthur) receives a quantum state (the proof) from an all-powerful prover (Merlin) and decides whether to accept. The class QMA(2) restricts Merlin’s proof to be a product across a fixed bipartition – equivalently, the two halves are sent by two unentangled Merlins. Understanding whether this restriction gives strictly more power is one of the central open problems in quantum complexity theory, unresolved for two decades (e.g. [49, 2, 38]). Despite substantial evidence that QMA(2) can outperform QMA in natural settings (e.g. [15, 2, 22, 58]), our unconditional knowledge in the standard constant-gap setting remains limited to the trivial inclusions

$$\text{QMA} \subseteq \text{QMA}(2) \subseteq \text{NEXP}$$

As posed in [2]: what is the power of unentanglement?

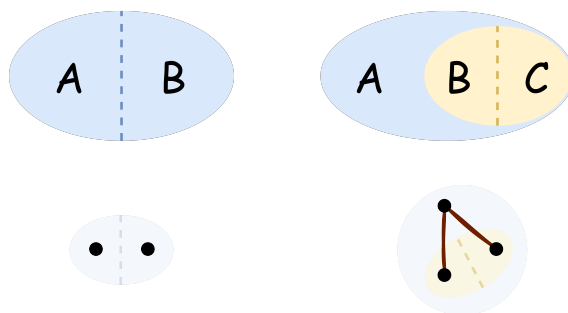
One of the unusual aspects of this landscape is that QMA(2) appears very sensitive to the structure of the proof system. Small changes to the model – extra interaction, very small gaps, or additional promises on the witness – can raise the power all the way to NEXP [58, 45, 10, 32]. This sensitivity complicates attempts to prove meaningful upper bounds for QMA(2). On the other hand, multipartite entanglement admits far richer structure than the binary distinction “entangled vs. unentangled,” motivating a finer-grained study of the witness’s entanglement structure. In this work, we take a first step exploring whether fine-grained entanglement promises can lead to a robust and analyzable refinement of QMA(2), or whether even very mild structural constraints on entanglement already trigger the same dramatic complexity jumps.

1.1 Internal separability: a fine-grained entanglement promise

We study the power of unentanglement through a natural multipartite promise that constrains only a *marginal* of the witness. We restrict Merlin to quantum states we call internally separable: a proof is a tripartite state ρ_{ABC} where C has only constant qubits and ρ_{BC} is separable. This is a mild constraint – ρ may remain entangled across every bipartition – but it fixes a separable marginal. Equivalently, these witnesses can be viewed as (partial) purifications of separable states. Importantly, internal separability is a constraint within a single proof. Our results show that even this very mild structural promise reveals unentanglement as powerful.

1.2 A qualitative one-versus-two jump, driven only by purity

Our main theorem establishes a sharp one-versus-two prover separation for internally-separable proofs. With one such proof, the resulting proof system is simulable in EXP. With two unentangled such proofs, the system becomes as powerful as NEXP. Thus, the “power of unentanglement” in this setting is as large as the gap between EXP and NEXP.



■ **Figure 1** Cartoons of a separable state and a state with a separable subsystem. The left side depicts a separable state, where parts A and B share no entanglement. The right side depicts a state where *after* tracing out part A , parts B and C share no entanglement; this state is in general entangled across every bipartition. The lower cartoons each display a graph where every vertex is a part (A , B , or C), and an edge between two parts X, Y allows bipartite entanglement in the reduced state ρ_{XY} . On the right side, part A is entangled with both part B and part C .

Beyond the separation itself, the proof reveals an unexpected mechanism. In our construction, the role of the two unentangled proofs is minimal: they are used solely to implement a purity test. This stands in contrast with the usual intuition around $\text{QMA}(2)$, where one typically relies on separability in a more robust way – most notably via the product test [38]. Conceptually, our results isolate a striking source of power: certifying that a state is pure (as opposed to mixed) can already suffice to unlock NEXP in the presence of even limited internal separability.

More broadly, our work highlights a qualitative distinction between pure and mixed witnesses in multi-prover quantum proof systems, and it points to a concrete route by which structural promises about internal entanglement might drive complexity-theoretic jumps. In particular, it resonates with recent efforts aimed at understanding whether $\text{QMA}(2)$ itself could reach NEXP [45, 10]. We hope this perspective – treating entanglement structure as a first-class computational resource – encourages further investigation into how subtle promises about witnesses can have outsized consequences for computational power.

2 Results

Consider the complexity class QMA_{IS} , a variant of QMA where the quantum proof must have a fixed, separable *subsystem*: that is, after tracing out all qubits beyond the subsystem, a predetermined set of $O(1)$ qubits are *not* entangled with the rest of the subsystem (see formal definition in Appendix A). Unlike in $\text{QMA}(2)$, this guarantee refers to *internal separability*: the two parts of the subsystem can be entangled, but only if the entanglement disappears when the subsystem is isolated. See Figure 1 for a visual comparison. We restrict one part of the subsystem to $O(1)$ qubits to zero in on the effect of *internal* entanglement structure; without the restriction, this class trivially contains $\text{QMA}(2)$.

We first show that QMA_{IS} , even with inverse exponentially small gap, is contained in EXP :

► **Theorem 1 (Upper bound).** *For any polynomial p and $1 > c > s > 0$ such that $c - s > \frac{1}{2^{p(n)}}$, we have $\text{QMA}_{\text{IS}}^{c,s} \subseteq \text{EXP}$, where c, s are the completeness and soundness parameters, respectively.¹*

¹ We formally define QMA_{IS} in Appendix A. In a $\text{QMA}_{\text{IS}}^{c,s}$ protocol, YES instances are accepted with probability at least c , and NO instances are accepted with probability at most s . QMA_{IS} includes every $\text{QMA}_{\text{IS}}^{c,s}$ with a constant gap $c - s = \Omega(1)$.

We also define $\text{QMA}_{\text{IS}}(2)$, where the verifier receives *two* unentangled proofs, each with the above guarantee. In contrast to Theorem 1, we find a constant-sized gap where $\text{QMA}_{\text{IS}}(2)$ equals NEXP.

► **Theorem 2** (Lower bound). *There exist constants $1 > c > s > 0$ where $\text{QMA}_{\text{IS}}^{c,s}(2) = \text{NEXP}$.*

We note two immediate consequences. First, there exists a quantum verifier with a specific guarantee on entanglement structure that is *unconditionally* (and *exponentially*) more powerful than a deterministic classical verifier. This follows as a consequence of the non-deterministic time hierarchy theorem:

► **Corollary 3.** $\text{NP} \subsetneq \text{QMA}_{\text{IS}}(2)$.

Second, QMA_{IS} and $\text{QMA}_{\text{IS}}(2)$ cannot be equal unless $\text{EXP} = \text{NEXP}$. In other words, the power of unentanglement in this setting is as strong as the difference between EXP and NEXP:

► **Corollary 4.** *If $\text{EXP} \neq \text{NEXP}$, then $\text{QMA}_{\text{IS}} \neq \text{QMA}_{\text{IS}}(2)$.*

We summarize these results in Table 1.

■ **Table 1** Categorizing quantum verification protocols by the guaranteed entanglement structures of the proof.

<i>internal structure</i> \ <i>external structure</i>	GENERAL	SEPARABLE BIPARTITION
GENERAL	QMA	QMA(2)
SEPARABLE SUBSYSTEM	$\text{QMA}_{\text{IS}} \subseteq \text{EXP}$	$\text{QMA}_{\text{IS}}(2) = \text{NEXP}$

3 Themes and implications

We highlight three conceptual takeaways: (i) internal entanglement promises are a natural complexity-theoretic probe of fine-grained separability structures; (ii) in our setting, unentanglement is used only to certify purity, exposing a sharp pure-vs-mixed distinction; (iii) this sheds some light on a gap-amplification route towards showing lower bounds of $\text{QMA}(2)$ that avoids known barriers for nearby models.

3.1 A complexity-theoretic probe of multipartite entanglement structure

Despite much attention given to $\text{QMA}(2)$ —the study of unentanglement, (un)entanglement can be much richer than simply across a bipartition. For example, the W and GHZ states, which are $\frac{1}{\sqrt{3}}(|001\rangle + |010\rangle + |100\rangle)$ and $\frac{1}{\sqrt{2}}(|000\rangle + |111\rangle)$ respectively, are entangled in *incomparable* ways: both states are entangled across every bipartition, yet the states are not equivalent under LOCC [27]. The same paper observes that tracing out a qubit from the W state gives an entangled state, but the same action on the GHZ state destroys all entanglement. These examples suggest that “entanglement vs. no entanglement across a cut” is only a coarse proxy for the structure that may matter in proof systems.

Our class QMA_{IS} enforces quantum proofs with a specific multipartite entanglement structure, which we call internal separability. To our knowledge, this notion first appeared in [19], where it was called *partial semi-separability*. It is natural to study the power of QMA with *other* types of multipartite entanglement structure. As a first step, we consider

the set of *multiseparable* states [60]: tripartite states where after tracing out *any* register, the reduced density matrix is separable. We show that a quantum proof system over pure, multiseparable states can also decide NEXP:

► **Corollary 5.** *There exist constants $1 > c > s > 0$ where $\text{QMA}^{c,s}$ restricted to pure, multiseparable quantum proofs can decide NEXP.*

A useful perspective is that many known NEXP-power phenomena in restricted-witness models arise because the promise enables the verifier to implement some form of *classicality/s-parsity detection* primitive – e.g., distinguishing structured “nearly classical” superpositions from genuinely diffuse ones. In our case, the internal separability promise plus purity plays exactly this role: it enables quasirigidity enforcement in the NEXP protocol. An interesting question is whether there is a robust way to characterize which multipartite entanglement promises yield NEXP-power in terms of the verification primitives they enable.

3.2 QMA(2) and purity testing

In our work, $\text{QMA}_{\text{IS}}(2)$ is no more powerful than QMA_{IS} with a purity test. This resonates with a broader theme in the previous works [54, 38, 9, 64] where they use the fact that a protocol over separable states can simulate a protocol over states with a purity constraint by the purity test. This raises the question: What is the relative power between purity and separability? For example, one QMA(2)-complete candidate was the *pure state marginal problem* [54]: decide if there exists a global pure state consistent with the input local density matrices. This problem is known to be in QMA(2), while the *mixed* variant of this problem is QMA-complete [52].

After our work was put online, [47] showed that the pure state marginal problem is in PSPACE, even with an inverse exponential promise gap. This suggests that the pure state marginal problem is in fact *not* QMA(2)-hard, as QMA(2) with an inverse-exponential completeness-soundness gap is equal to NEXP [58]. Moreover, they propose one systematic way to understand the power of purity by abstracting out the complexity class definition based on the pure version of the super verifier characterization of QMA [5]. Many of the questions remain unclear for their model.

The quantum proof systems are associated with their classical optimization counterparts. Therefore, we also consider the optimization problem over *purifications* of separable states that is enough to decide NP, and even hard to approximate within a constant error:

► **Corollary 6.** *Given $M \in \mathbb{R}^{n \times n}$, consider the following optimization problem over n -dimensional pure, internally separable states (denoted $\mathcal{W}_{\text{IS}}$):*

$$\max_{\rho \in \mathcal{W}_{\text{IS}}} \text{Tr}[M\rho] \tag{*}$$

Then there is a constant $0 < \alpha < 1$ for which it is NP-hard to compute an α -approximation of ().*

By contrast, optimizing over separable states (of $\text{poly}(n)$ dimension) is known to be NP-hard, but only with a gap inverse polynomial of the system size [35, 2, 29, 15, 58, 28]. If the gap in these problems could be made constant, then $\text{QMA}(2) = \text{NEXP}$.²

More generally, assuming mild conditions on a set of quantum states $\mathcal{W}$, we show how a QMA(2) protocol with both proofs restricted to $\mathcal{W}$ can implement a QMA protocol with a proof restricted to *pure states* in $\mathcal{W}$. We use this as a recipe to generate other proof systems

² One must also assume that the problems can be made succinct.

where one quantum proof is provably weaker than two *unentangled* quantum proofs, up to standard complexity-theoretic assumptions. We give two examples: the pure quantum proofs in QMA^+ , and the *proper states* of [2, 12]. Each set represents the pure states of some larger set $\mathcal{W}$. Then, $\text{QMA}_{\mathcal{W}}$ can be decided with a semidefinite program, but $\text{QMA}_{\mathcal{W}}(2)$ can restrict to the more capable pure states. This technique is likely to generate other examples that demonstrate the power of unentanglement.

3.3 A revived approach towards $\text{QMA}(2)$

Corollary 4 suggests a potentially cleaner gap-amplification route towards understanding the power of $\text{QMA}(2)$. For a constant r , denote by $\text{QMA}_{\text{IS}}^{c,s}[r](k)$ the class of problems in $\text{QMA}_{\text{IS}}^{c,s}(k)$ with at most r dimensions for the witnesses' bounded part of the separable subsystem. Since r is constant, we have that for any constant k , $\text{QMA}_{\text{IS}}[r](k)$ at large enough completeness-soundness gap is equal to $\text{QMA}(k)$:

▷ **Claim 7.** For any constants k and r , there exist constants $1 > c > s > 0$ so that $\text{QMA}_{\text{IS}}^{c,s}[r](k) = \text{QMA}(k)$.

By Claim 7 and Theorem 2,³ gap amplification of $\text{QMA}_{\text{IS}}[r](2)$ would imply $\text{QMA}(2) = \text{NEXP}$:

► **Corollary 8.** *If for some r $\text{QMA}_{\text{IS}}[r](2)$ exhibits gap amplification, then $\text{QMA}(2) = \text{NEXP}$.*

Corollary 8 is analogous to an approach suggested in [45]. That work introduced the class $\text{QMA}^+(k)$, and showed that $\text{QMA}^+(2)$ at some completeness-soundness gap is equal to NEXP , while at a larger gap, is equal to $\text{QMA}(2)$. Therefore, if $\text{QMA}^+(2)$ admits gap amplification, then $\text{QMA}(2) = \text{NEXP}$. However, [10] showed that the same is true for QMA^+ ; as a consequence, QMA^+ *cannot* have gap amplification unless $\text{QMA} = \text{NEXP}$. Conceptually, this limits the techniques available to prove gap amplification of $\text{QMA}^+(2)$ in order to show $\text{QMA}(2) = \text{NEXP}$. By contrast, gap amplification of $\text{QMA}_{\text{IS}}(2)$ can be plausibly found by studying gap amplification of QMA_{IS} , since the latter implies no such complexity collapse. This raises the possibility of using Corollary 8 to fully characterize $\text{QMA}(2)$. However, any such gap amplification must be sensitive to *purity*, as QMA_{IS} suffers the same issue as QMA^+ . We reinterpret the barrier of [10] by constructing a mixed-state variant of QMA^+ that also avoids the complexity collapse, as broadly explained in the previous subsection.

4 Techniques

4.1 Upper bound

To show Theorem 1, we rely on a well-known duality of membership testing of a convex set and optimizing a convex function over this set. We use this duality to reduce any QMA_{IS} problem to membership testing of the set of internally separable quantum states $\mathcal{W}_{\text{IS}}$. In order to use tools from convex optimization, we represent quantum states as vectors in a generalized Bloch sphere. We use a non-ellipsoidal reduction that permits an error analysis matching our setup [53, 29].

We then show how to test membership of $\mathcal{W}_{\text{IS}}$ in EXP . It is essential that only $O(1)$ qubits are unentangled from the rest of the subsystem. Our algorithm tests membership of two convex sets whose intersection forms $\mathcal{W}_{\text{IS}}$; note that this requires testers with inverse

³ The theorem also holds for $\text{QMA}_{\text{IS}}^{c,s}[r](2)$ for some fixed constant r .

exponential precision. The first set represents all Bloch vectors of quantum states, which can be tested using exponentially many explicit conditions (e.g. [48]). The second set represents all vectors whose reduced density matrix on the subsystem is separable. This can be tested by solving the Doherty-Parrilo-Spedalieri SDP [25, 26, 56] which decides if a state has a *symmetric extension*. A careful error analysis shows that separability of a bipartite state ρ with local dimensions M, N can be tested to error δ in time $\tilde{O}((2/\delta)^{9N} \text{poly}(M, N))$ [44]. When N is a fixed constant, this SDP runs in exponential time.

4.2 Lower bound

To prove Theorem 2, we consider $\text{QMA}_{|\mathcal{S}\rangle}$, a variant of $\text{QMA}_{\mathcal{S}}$ where the proof must also be a *pure state*. Note that unlike QMA and $\text{QMA}(2)$, restricting $\text{QMA}_{\mathcal{S}}$ to pure states can change the power of the complexity class, as an internally separable ensemble of pure states may not decompose into an *ensemble* of internally separable pure states.⁴ We then develop a $\text{QMA}_{|\mathcal{S}\rangle}$ protocol for an NEXP-complete problem. This protocol builds on the recent proof of $\text{QMA}^+ = \text{NEXP}$ [15, 45, 10], where QMA^+ is restricted to pure quantum proofs with real nonnegative amplitudes.

As in [45, 10], we consider a succinct constraint satisfaction problem (CSP) with constant promise gap, which is NEXP-hard by the PCP theorem [7, 6, 40]. In the QMA^+ protocol, one enforces a *rigidity* property of the quantum proof: the state must be close to $\frac{1}{\sqrt{R}} \sum_{j \in [R]} |j\rangle |a_j\rangle$. It is easy to ensure that all constraints $j \in [R]$ are included in the state (“density”), but the positive-amplitude guarantee is used to ensure there is at most one assignment a_j per constraint (“quasirigidity”). We find a way to instead enforce quasirigidity using both internal separability *and* purity.

► **Lemma 9** (informal). *There is a test A and a fixed quantum circuit C such that for any internally separable $|\psi\rangle$, if $A(|\psi\rangle) \geq 1 - \epsilon$, then $C|\psi\rangle$ has $1 - O(\epsilon)$ squared overlap with a quasirigid state.*

In our $\text{QMA}_{|\mathcal{S}\rangle}$ protocol, the verifier enforces tripartite proofs of the specific form $\sum_{j,a} z_{ja} |j\rangle |a, j\rangle |a\rangle$. This can be done by measuring all registers and accepting iff we see two copies of the same constraint j , and two copies of the same assignment a . We then observe that having multiple assignments per constraint adds entanglement to the second and third registers, even after tracing out the first register. Thus, if the state is *internally separable*, it has high overlap with a state with one assignment a per constraint j . Notably, this test is inspired by recasting the QMA^+ protocol in the language of “superposition detection” and non-collapsing measurement [11, 4], an unphysical operation related to a problem of Aaronson [1].

We then prove that $\text{QMA}_{|\mathcal{S}\rangle} \subseteq \text{QMA}_{\mathcal{S}}(2)$. Without loss of generality, the best proof Merlin can send in a $\text{QMA}_{\mathcal{S}}(2)$ protocol is a product state $\rho_1 \otimes \rho_2$, since separable states form the convex hull of the set of product states. The SWAP test accepts a product state with probability $\frac{1}{2}(1 + \text{Tr}[\rho_1 \rho_2])$: this is 1 only if ρ_1, ρ_2 are *pure*. The $\text{QMA}_{\mathcal{S}}(2)$ simulator thus applies a SWAP test with some probability, and otherwise applies the $\text{QMA}_{|\mathcal{S}\rangle}$ circuit.

To make this formal, we first generalize the above statement. Given a set of quantum states $\mathcal{W}$, let $\text{QMA}_{\mathcal{W}}$ be the variant of QMA where the proof must be in $\mathcal{W}$. Let $\text{rank1}(\mathcal{W}) \subseteq \mathcal{W}$ be the pure states contained in $\mathcal{W}$. Then provided $\mathcal{W}$ satisfies a mild *continuity condition*, we have that $\text{QMA}_{\text{rank1}(\mathcal{W})} \subseteq \text{QMA}_{\mathcal{W}}(2)$.

⁴ For example, the same is true in QMA^+ : this is the difference between doubly non-negative matrices and completely positive matrices.

► **Lemma 10** (informal). *Consider a protocol V in $\text{QMA}_{\text{rank1}(\mathcal{W})}$. If $\mathcal{W}$ has the continuity condition, then $\text{QMA}_{\mathcal{W}}(2)$ can simulate V .*

Intuitively, the continuity condition requires that every density operator in $\mathcal{W}$ that is “almost pure” is close to a pure state in $\mathcal{W}$; i.e. in $\text{rank1}(\mathcal{W})$. The condition ensures that when the SWAP test accepts with high probability, both ρ_1 and ρ_2 are close to a state in $\text{rank1}(\mathcal{W})$.

Ideally, we would show that $\mathcal{W}_{\text{IS}}$ satisfies the continuity condition, but a proof of this seems difficult. Nonetheless, we find a large subset $\mathcal{W}_{\text{IS}}^* \subseteq \mathcal{W}_{\text{IS}}$ that has the continuity condition, such that the NEXP protocol is still contained in $\text{QMA}_{\text{rank1}(\mathcal{W}_{\text{IS}}^*)}$. By the above lemma, NEXP is contained in $\text{QMA}_{\mathcal{W}_{\text{IS}}^*}(2)$. To complete the proof, we show how to simulate any protocol in $\text{QMA}_{\mathcal{W}_{\text{IS}}^*}(2)$ using a protocol in $\text{QMA}_{\mathcal{W}_{\text{IS}}}(2)$.

5 Related work

5.1 The complexity class $\text{QMA}(2)$

One sign that $\text{QMA}(2)$ is more powerful than QMA is that QMA with a $O(\log n)$ -qubit proof is equal to BQP [55]. By contrast, $\text{QMA}(2)$ with two unentangled $O(\log n)$ -qubit proofs can decide NP, although with inverse polynomially small completeness-soundness gap [2, 12, 15, 22, 23, 28, 58]. Improving the completeness-soundness gap to a constant would likely imply $\text{QMA}(2) = \text{NEXP}$; see [45, 10, 46] for a recent attempt using quantum proofs with non-negative amplitudes (QMA^+). In general, one only needs two unentangled proofs since $\text{QMA}(2) = \text{QMA}(k)$ [38]; the proof of this result uses an iterated SWAP test. There are few complete problems known for $\text{QMA}(2)$ [21, 41, 36]. One proposal is the *pure state marginal problem*, which is known to be QMA-hard and in $\text{QMA}(2)$ [52, 54, 53, 63, 65, 64]. Recently, [47] proved that this problem is in PSPACE, and gave evidence that it is *not* $\text{QMA}(2)$ -hard.

5.2 Convex optimization and quantum entanglement

Some quantum papers build a Turing reduction using the duality between convex optimization and membership testing of a convex set, for example NP-hardness of separability testing [35, 44, 29], and QMA-hardness of the mixed state marginal problem [52, 54, 53, 63]. This reduction can be implemented in several ways, including the ellipsoid method [66, 34], random walks [14], and a perceptron-like algorithm [53]. One can use a hierarchy of semidefinite programs to detect entanglement [25, 26, 56, 39] or optimize over separable states [16, 17, 9]. In some cases, carefully chosen ϵ -nets provide similar performance [59, 8, 18].

5.3 Measuring quantum entanglement

There are numerous ways to quantify entanglement; see the surveys [67, 37, 43, 42]. One approach considers the set of reachable states under transformations consisting only of local operations and classical communication (LOCC) [27, 13, 62]. Other works classify multipartite entanglement by whether a *subsystem* is separable [60, 19]. Permutation-symmetric approaches include the Positive Partial Transpose (PPT) criterion [57] and quantum de Finetti theorems [20, 24]. Mysteriously, both approaches identify entangled states that “look” separable under LOCC transformations [61, 16, 17, 50, 51]. There exist ensembles of low-entanglement quantum states that are computationally indistinguishable from maximally entangled quantum states [3].

6 Outlook

There are several ways to further the ideas within this work. In our view, the most promising future direction is to study other quantum proof systems with other multipartite entanglement structures. For example, what is the power of $\text{QMA}_{\mathcal{W}}$ when the states in $\mathcal{W}$ are the Absolutely Maximally Entangled states (see review in [31])?

Another direction is to probe the relationship of $\text{QMA}(2)$ and purity testing. The proof of Theorem 2 uses two quantum proofs only to implement a purity test. Is there a set of proofs $\mathcal{W}$ such that $\text{QMA}_{\mathcal{W}}(2)$ is strictly more powerful than $\text{QMA}_{\text{rank1}(\mathcal{W})}$? If so, there is a setting where *unentanglement* is useful for something beyond purity testing. A related technical question is whether the *continuity condition* is necessary to show $\text{QMA}_{\text{rank1}(\mathcal{W})} \subseteq \text{QMA}_{\mathcal{W}}(2)$. Is there a simple characterization of sets which do *not* satisfy the continuity condition?

This work also reignites the possibility of showing the power of $\text{QMA}(2)$ by proving a gap amplification result (Corollary 8). As a starting point, does the complexity class QMA_{IS} exhibit gap amplification? (Even a small amount of gap amplification implies the existence of new complete problems for QMA_{IS} .) One can ask similar questions about the additional quantum proof systems we discuss, see Section 3.1. Unlike with QMA^+ [10], each of these complexity classes demonstrates that unentanglement can be as powerful as the difference between EXP vs. NEXP.

References

- 1 Scott Aaronson. Quantum miscellany — shtetl-optimized, 2023. URL: <https://scottaaronson.blog/?p=7516>.
- 2 Scott Aaronson, Salman Beigi, Andrew Drucker, Bill Fefferman, and Peter Shor. The power of unentanglement. *Theory of Computing*, 5(1):1–42, 2009. doi:10.4086/toc.2009.v005a001.
- 3 Scott Aaronson, Adam Bouland, Bill Fefferman, Soumik Ghosh, Umesh Vazirani, Chenyi Zhang, and Zixin Zhou. Quantum Pseudoentanglement. In *15th Innovations in Theoretical Computer Science Conference (ITCS 2024)*, 2024. doi:10.4230/LIPIcs.ITCS.2024.2.
- 4 Scott Aaronson, Sabee Grewal, Vishnu Iyer, Simon C. Marshall, and Ronak Ramachandran. $\text{Pdqma} = \text{dqma} = \text{nexp}$: Qma with hidden variables and non-collapsing measurements, 2024. doi:10.48550/arXiv.2403.02543.
- 5 Dorit Aharonov and Oded Regev. A lattice problem in quantum np, 2003. arXiv:quant-ph/0307220.
- 6 Sanjeev Arora, Carsten Lund, Rajeev Motwani, Madhu Sudan, and Mario Szegedy. Proof verification and the hardness of approximation problems. *J. ACM*, 45(3):501–555, 1998. doi:10.1145/278298.278306.
- 7 Sanjeev Arora and Shmuel Safra. Probabilistic checking of proofs; A new characterization of NP. In *33rd Annual Symposium on Foundations of Computer Science*, 1992. doi:10.1109/SFCS.1992.267824.
- 8 Boaz Barak, Jonathan A. Kelner, and David Steurer. Rounding sum-of-squares relaxations. In *Proceedings of the Forty-Sixth Annual ACM Symposium on Theory of Computing*, 2014. doi:10.1145/2591796.2591886.
- 9 Boaz Barak, Pravesh K. Kothari, and David Steurer. Quantum entanglement, sum of squares, and the log rank conjecture. In *Proceedings of the 49th Annual ACM SIGACT Symposium on Theory of Computing*, 2017. doi:10.1145/3055399.3055488.
- 10 Roozbeh Bassirian, Bill Fefferman, and Kunal Marwaha. Quantum Merlin-Arthur and proofs without relative phase. In *15th Innovations in Theoretical Computer Science Conference (ITCS 2024)*, 2023. doi:10.4230/LIPIcs.ITCS.2024.9.
- 11 Roozbeh Bassirian and Kunal Marwaha. Superposition detection and qma with non-collapsing measurements, 2024. doi:10.48550/arXiv.2403.02532.

- 12 Salman Beigi. Np vs qma_{log(2)}, 2008. [arXiv:0810.5109](#).
- 13 Charles H. Bennett, Sandu Popescu, Daniel Rohrlich, John A. Smolin, and Ashish V. Thapliyal. Exact and asymptotic measures of multipartite pure-state entanglement. *Phys. Rev. A*, 63, December 2000. doi:10.1103/PhysRevA.63.012307.
- 14 Dimitris Bertsimas and Santosh Vempala. Solving convex programs by random walks. *J. ACM*, 51(4):540–556, June 2004. doi:10.1145/1008731.1008733.
- 15 Hugue Blier and Alain Tapp. A quantum characterization of np. *Comput. Complex.*, September 2012. doi:10.1007/s00037-011-0016-2.
- 16 Fernando G.S.L. Brandão, Matthias Christandl, and Jon Yard. A quasipolynomial-time algorithm for the quantum separability problem. In *Proceedings of the Forty-Third Annual ACM Symposium on Theory of Computing*, June 2011. doi:10.1145/1993636.1993683.
- 17 Fernando G.S.L. Brandão and Aram W. Harrow. Quantum de finetti theorems under local measurements with applications. In *Proceedings of the Forty-Fifth Annual ACM Symposium on Theory of Computing*, 2013. doi:10.1145/2488608.2488718.
- 18 Fernando G.S.L. Brandão and Aram W. Harrow. Estimating operator norms using covering nets, 2015. [arXiv:1509.05065](#).
- 19 Gilles Brassard and Tal Mor. Multi-particle entanglement via two-party entanglement. *Journal of Physics A: Mathematical and General*, 34(35), August 2001. doi:10.1088/0305-4470/34/35/306.
- 20 Carlton M. Caves, Christopher A. Fuchs, and Rüdiger Schack. Unknown quantum states: The quantum de finetti representation. *Journal of Mathematical Physics*, 43(9), September 2002. doi:10.1063/1.1494475.
- 21 Andre Chailloux and Or Sattath. The complexity of the separable hamiltonian problem. In *2012 IEEE 27th Conference on Computational Complexity*. IEEE, June 2012. doi:10.1109/coc.2012.42.
- 22 Jing Chen and Andrew Drucker. Short multi-prover quantum proofs for sat without entangled measurements, 2010. [arXiv:1011.0716](#).
- 23 Alessandro Chiesa and Michael A. Forbes. Improved soundness for qma with multiple provers. *Chicago Journal of Theoretical Computer Science*, January 2013. [arXiv:1108.2098](#).
- 24 Matthias Christandl, Robert König, Graeme Mitchison, and Renato Renner. One-and-a-half quantum de finetti theorems. *Communications in Mathematical Physics*, 273(2):473–498, March 2007. doi:10.1007/s00220-007-0189-3.
- 25 Andrew C. Doherty, Pablo A. Parrilo, and Federico M. Spedalieri. Distinguishing separable and entangled states. *Phys. Rev. Lett.*, 88, April 2002. doi:10.1103/PhysRevLett.88.187904.
- 26 Andrew C. Doherty, Pablo A. Parrilo, and Federico M. Spedalieri. Complete family of separability criteria. *Physical Review A*, 69(2), February 2004. doi:10.1103/physreva.69.022308.
- 27 W. Dür, G. Vidal, and J. I. Cirac. Three qubits can be entangled in two inequivalent ways. *Phys. Rev. A*, 62, November 2000. doi:10.1103/PhysRevA.62.062314.
- 28 François Le Gall, Shota Nakagawa, and Harumichi Nishimura. On QMA protocols with two short quantum proofs. *Quantum Inf. Comput.*, 12(7-8):589–600, 2012. doi:10.26421/QIC12.7-8-4.
- 29 Sevag Gharibian. Strong np-hardness of the quantum separability problem. *Quantum Info. Comput.*, 10(3):343–360, March 2010. doi:10.5555/2011350.2011361.
- 30 Sevag Gharibian. Guest column: The 7 faces of quantum NP. *SIGACT News*, 54(4):54–91, 2023. doi:10.1145/3639528.3639535.
- 31 Dardo Goyeneche, Daniel Alsina, José I Latorre, Arnau Riera, and Karol Życzkowski. Absolutely maximally entangled states, combinatorial designs, and multiunitary matrices. *Physical Review A*, 92(3), 2015. [arXiv:1506.08857](#).
- 32 Sabee Grewal and William Kretschmer. Unentanglement and post-measurement branching in quantum interactive proofs. *CoRR*, abs/2509.15319, 2025. doi:10.48550/arXiv.2509.15319.

- 33 Alex B. Grilo, Iordanis Kerenidis, and Jamie Sikora. Qma with subset state witnesses. In *International Symposium on Mathematical Foundations of Computer Science*, 2015. [arXiv:1410.2882](#).
- 34 Martin Grötschel, László Lovász, and Alexander Schrijver. *Geometric algorithms and combinatorial optimization*, volume 2. Springer Science & Business Media, 1988. [doi:10.1007/978-3-642-97881-4](#).
- 35 Leonid Gurvits. Classical deterministic complexity of edmonds' problem and quantum entanglement. In *Proceedings of the Thirty-Fifth Annual ACM Symposium on Theory of Computing*, 2003. [doi:10.1145/780542.780545](#).
- 36 Gus Gutoski, Patrick Hayden, Kevin Milner, and Mark M. Wilde. Quantum interactive proofs and the complexity of separability testing. *Theory of Computing*, 11(1):59–103, 2015. [doi:10.4086/toc.2015.v011a003](#).
- 37 Otfried Gühne and Géza Tóth. Entanglement detection. *Physics Reports*, 474(1–6):1–75, April 2009. [doi:10.1016/j.physrep.2009.02.004](#).
- 38 Aram W. Harrow and Ashley Montanaro. Testing product states, quantum merlin-arthur games and tensor optimization. *Journal of the ACM (JACM)*, 60(1):1–43, 2013. [doi:10.1145/2432622.2432625](#).
- 39 Aram W. Harrow, Anand Natarajan, and Xiaodi Wu. An improved semidefinite programming hierarchy for testing entanglement. *Communications in Mathematical Physics*, 352(3):881–904, March 2017. [doi:10.1007/s00220-017-2859-0](#).
- 40 Prahladh Harsha. *Robust PCPs of proximity and shorter PCPs*. PhD thesis, Massachusetts Institute of Technology, 2004. URL: <https://dspace.mit.edu/bitstream/handle/1721.1/26720/59552830-MIT.pdf>.
- 41 Patrick Hayden, Kevin Milner, and Mark M. Wilde. Two-message quantum interactive proofs and the quantum separability problem. In *2013 IEEE Conference on Computational Complexity*, June 2013. [doi:10.1109/cc.2013.24](#).
- 42 Paweł Horodecki, Lukasz Rudnicki, and Karol Życzkowski. Multipartite entanglement, 2024. [arXiv:2409.04566](#).
- 43 Ryszard Horodecki, Paweł Horodecki, Michał Horodecki, and Karol Horodecki. Quantum entanglement. *Rev. Mod. Phys.*, 81:865–942, June 2009. [doi:10.1103/RevModPhys.81.865](#).
- 44 Lawrence M. Ioannou. Computational complexity of the quantum separability problem. *Quantum Info. Comput.*, 7(4):335–370, May 2007. [doi:10.26421/QIC7.4-5](#).
- 45 Fernando Granha Jeronimo and Pei Wu. The power of unentangled quantum proofs with non-negative amplitudes. In *Proceedings of the 55th Annual ACM Symposium on Theory of Computing*, 2023. [doi:10.1145/3564246.3585248](#).
- 46 Fernando Granha Jeronimo and Pei Wu. Dimension Independent Disentangler from Unentanglement and Applications. In *39th Computational Complexity Conference (CCC 2024)*, 2024. [doi:10.4230/LIPIcs.CCC.2024.26](#).
- 47 Jonas Kamminga and Dorian Rudolph. The Pure-State Consistency of Local Density Matrices Problem: In PSPACE and Complete for a Class Between QMA and QMA(2). In *17th Innovations in Theoretical Computer Science Conference (ITCS 2026)*, 2026. [doi:10.4230/LIPIcs.ITCS.2026.83](#).
- 48 Gen Kimura. The bloch vector for n-level systems. *Physics Letters A*, 314(5–6):339–349, August 2003. [doi:10.1016/s0375-9601\(03\)00941-1](#).
- 49 Hirotada Kobayashi, Keiji Matsumoto, and Tomoyuki Yamakami. Quantum merlin-arthur proof systems: Are multiple merlins more helpful to arthur? In *Algorithms and Computation*, 2003. [doi:10.1007/978-3-540-24587-2_21](#).
- 50 Cécilia Lancien and Andreas Winter. Distinguishing multi-partite states by local measurements. *Communications in Mathematical Physics*, 323(2):555–573, August 2013. [doi:10.1007/s00220-013-1779-x](#).
- 51 Ke Li and Graeme Smith. Quantum de finetti theorem under fully-one-way adaptive measurements. *Physical Review Letters*, 114(16), April 2015. [doi:10.1103/physrevlett.114.160503](#).

- 52 Yi-Kai Liu. Consistency of local density matrices is qma-complete. In *APPROX and RANDOM*, 2006. [arXiv:quant-ph/0604166](#).
- 53 Yi-Kai Liu. *The Complexity of the Consistency and N-representability Problems for Quantum States*. PhD thesis, University of California, San Diego, 2007. [arXiv:0712.3041](#).
- 54 Yi-Kai Liu, Matthias Christandl, and F. Verstraete. N-representability is qma-complete. *Physical Review Letters*, 98(11), March 2007. [doi:10.1103/physrevlett.98.110503](#).
- 55 Chris Marriott and John Watrous. Quantum arthur-merlin games. *Comput. Complex.*, 2005. [doi:10.1007/s00037-005-0194-x](#).
- 56 Miguel Navascués, Masaki Owari, and Martin B. Plenio. Power of symmetric extensions for entanglement detection. *Physical Review A*, 80(5), November 2009. [doi:10.1103/physreva.80.052306](#).
- 57 Asher Peres. Separability criterion for density matrices. *Physical Review Letters*, 77(8):1413–1415, August 1996. [doi:10.1103/physrevlett.77.1413](#).
- 58 Attila Pereszlényi. Multi-prover quantum merlin-arthur proof systems with small gap, 2012. [arXiv:1205.2761](#).
- 59 Yaoyun Shi and Xiaodi Wu. Epsilon-net method for optimizations over separable states. *Theoretical Computer Science*, 598:51–63, 2015. [doi:10.1016/j.tcs.2015.03.031](#).
- 60 Ashish V. Thapliyal. Multipartite pure-state entanglement. *Physical Review A*, 59(5):3336–3342, May 1999. [doi:10.1103/physreva.59.3336](#).
- 61 Géza Tóth and Otfried Gühne. Entanglement and permutational symmetry. *Phys. Rev. Lett.*, 102, May 2009. [doi:10.1103/PhysRevLett.102.170503](#).
- 62 Péter Vrana and Matthias Christandl. Asymptotic entanglement transformation between W and GHZ states. *Journal of Mathematical Physics*, 56(2), February 2015. [arXiv:1310.3244](#).
- 63 Tzu-Chieh Wei, Michele Mosca, and Ashwin Nayak. Interacting boson problems can be qma hard. *Phys. Rev. Lett.*, 104, January 2010. [doi:10.1103/PhysRevLett.104.040501](#).
- 64 Xiao-Dong Yu, Timo Simnacher, H. Chau Nguyen, and Otfried Gühne. Quantum-inspired hierarchy for rank-constrained optimization. *PRX Quantum*, 3(1), March 2022. [doi:10.1103/prxquantum.3.010340](#).
- 65 Xiao-Dong Yu, Timo Simnacher, Nikolai Wyderka, H. Chau Nguyen, and Otfried Gühne. A complete hierarchy for the pure state marginal problem in quantum mechanics. *Nature Communications*, 12(1), February 2021. [doi:10.1038/s41467-020-20799-5](#).
- 66 D.B. Yudin and A.S. Nemirovskii. Informational complexity and efficient methods for the solution of convex extremal problems. *Ekonomika i Matematicheskie Metody*, 12:357–369, 1976. English translation: *Matekon* 13 (3) pp.25-45 (1977). URL: <https://books.google.com/books?id=0kEzAAAAIAAJ>.
- 67 Karol Życzkowski and Ingemar Bengtsson. *Geometry of Quantum States: An Introduction to Quantum Entanglement*. Cambridge University Press, 2 edition, 2017. The arXiv link is based on a book chapter. [doi:10.1017/9781139207010](#).

A Formal Definitions

Fix any set of quantum states $\mathcal{W}$. Let $\text{QMA}_{\mathcal{W}}(m)$ as $\text{QMA}(m)$ where Merlin *must* send m unentangled proofs from $\mathcal{W}$. Here we modify both completeness and soundness. Only modifying completeness sometimes has no effect, as seen in [33].

► **Definition 11** ($\text{QMA}_{\text{IS}}(m), \text{QMA}_{\text{IS}}^c$). Let $c, s : \mathbb{N} \rightarrow [0, 1]$ and $\hat{a}, \hat{b}, \hat{c} : \mathbb{N} \rightarrow \mathbb{N}$ be polynomial-time computable functions, and $m : \mathbb{N} \rightarrow \mathbb{N}$ be a function.

A promise problem $A = (A_{\text{yes}}, A_{\text{no}})$ is in $\text{QMA}_{\text{IS}}^{c,s}[\hat{a}, \hat{b}, \hat{c}](m)$ if and only if there exist a polynomial p and a polynomial-time uniform family of quantum circuits $\{Q_n\}$ that satisfies the following:

1. Q_n takes as input:
 - a string $x \in \{0, 1\}^n$;
 - m quantum states $\rho_1, \rho_2, \dots, \rho_m \in A \otimes B \otimes C$, each over three registers A, B, C of dimensions $\hat{a}, \hat{b}, \hat{c}$ respectively, and such that each is internally separable, meaning that $\text{Tr}_A(\rho_i)$ is separable;
 - and at most $p(n)$ scratch qubits in the state $|0\rangle$.
2. Q_n is complete and sound for A :
 - (completeness) If $x \in A_{\text{yes}}$, there is a set of m such quantum states $\rho_1, \dots, \rho_m$ such that Q_n accepts $|x\rangle\langle x| \otimes (\rho_1 \otimes \dots \otimes \rho_m) \otimes (|0\rangle\langle 0| \otimes \dots \otimes |0\rangle\langle 0|)$ with probability at least $c(n)$.
 - (soundness) If $x \in A_{\text{no}}$, then for any set of m such quantum states $\rho_1, \dots, \rho_m$, Q_n accepts $|x\rangle\langle x| \otimes (\rho_1 \otimes \dots \otimes \rho_m) \otimes (|0\rangle\langle 0| \otimes \dots \otimes |0\rangle\langle 0|)$ with probability at most $s(n)$.

Let

$$\text{QMA}_{|\text{IS}\rangle}^{c,s}(m) := \bigcup_{\substack{\hat{a}, \hat{b} = O(\text{poly}(n)), \\ \hat{c} = O(1)}} \text{QMA}_{|\text{IS}\rangle}^{c,s}[\hat{a}, \hat{b}, \hat{c}](m).$$

Define $\text{QMA}_{|\text{IS}\rangle}^{c,s}$ similarly, where each quantum state $\rho_i = |\psi_i\rangle\langle\psi_i|$ is always pure.

We also generalize the above definition for other sets of quantum states.

► **Definition 12** ($\text{QMA}_{\mathcal{W}}(m)$). Let $c, s : \mathbb{N} \rightarrow \mathbb{R}^+$ be polynomial-time computable functions, and m a polynomial. Fix an arbitrary set of quantum states $\mathcal{W}$.

A promise problem $A = (A_{\text{yes}}, A_{\text{no}})$ is in $\text{QMA}_{\mathcal{W}}^{c,s}(m)$ if and only if there exist polynomials p, q and a polynomial-time uniform family of quantum circuits $\{Q_n\}$, where Q_n takes as input a string $x \in \{0, 1\}^n$, m $p(n)$ -qubit quantum states in $\mathcal{W}$, and $q(n)$ scratch qubits in state $|0\rangle^{\otimes q(n)}$, such that:

- (completeness) If $x \in A_{\text{yes}}$, there exists a set of m $p(n)$ -qubit quantum states $\rho_1, \dots, \rho_m \in \mathcal{W}$ such that Q_n accepts $|x\rangle\langle x| \otimes (\rho_1 \otimes \dots \otimes \rho_m) \otimes |0\rangle\langle 0|^{\otimes q(n)}$ with probability at least $c(n)$.
- (soundness) If $x \in A_{\text{no}}$, then for any set of m $p(n)$ -qubit quantum states $\rho_1, \dots, \rho_m \in \mathcal{W}$, Q_n accepts $|x\rangle\langle x| \otimes (\rho_1 \otimes \dots \otimes \rho_m) \otimes |0\rangle\langle 0|^{\otimes q(n)}$ with probability at most $s(n)$.

► **Remark 13.** Following the convention in [45], we drop the c, s when the completeness-soundness gap is allowed to be any absolute constant: $\text{QMA}_{\mathcal{W}}(m) \stackrel{\text{def}}{=} \bigcup_{c=s=\Omega(1)} \text{QMA}_{\mathcal{W}}^{c,s}(m)$. However, note that $\text{QMA}_{\mathcal{W}}^{c,s}(m)$ is not in general equal to $\text{QMA}_{\mathcal{W}}^{c',s'}(m)$ [10].

We will only consider $\text{QMA}_{|\text{IS}\rangle}$ with one proof, as we prove $\text{QMA}_{|\text{IS}\rangle} = \text{NEXP}$. This implies $\text{QMA}_{|\text{IS}\rangle}(m) = \text{NEXP}$ for all $m \geq 1$.

We occasionally select another property P to choose a set of quantum states which we denote $\mathcal{W}_P$. When this occurs, we use the shorthand $\text{QMA}_P(m) \stackrel{\text{def}}{=} \text{QMA}_{\mathcal{W}_P}(m)$. We use the label $|P\rangle$ for the pure states in $\mathcal{W}_P$.

Limitations of Decoded Quantum Interferometry for MaxCut

Ojas Parekh 

Quantum Algorithms and Applications Collaboratory, Sandia National Laboratories,
Albuquerque, NM, USA

Abstract

Decoded Quantum Interferometry (DQI) is a framework for approximating special kinds of discrete optimization problems that relies on problem structure in a way that sets it apart from other classical or quantum approaches. We show that any family of instances of MaxCut on which DQI attains a nontrivial asymptotic approximation guarantee is also solvable exactly in classical polynomial time. We include a streamlined exposition of DQI tailored for MaxCut that relies on elementary graph theory instead of coding theory to motivate and explain the algorithm.

2012 ACM Subject Classification Theory of computation → Approximation algorithms analysis; Theory of computation → Quantum complexity theory

Keywords and phrases Decoded Quantum Interferometry, MaxCut, quantum approximation algorithms

Digital Object Identifier 10.4230/LIPIcs.TQC.2026.6

Related Version *arXiv Version*: <https://arxiv.org/abs/2509.19966>

Funding This work is supported by the U.S. Department of Energy (DOE), Office of Science, National Quantum Information Science Research Centers, Quantum Systems Accelerator (QSA). Additional support is acknowledged from DOE, Office of Science, Accelerated Research in Quantum Computing, Fundamental Algorithmic Research toward Quantum Utility (FAR-Qu). This article has been authored by an employee of National Technology & Engineering Solutions of Sandia, LLC under Contract No. DE-NA0003525 with the U.S. Department of Energy (DOE). The employee owns all right, title and interest in and to the article and is solely responsible for its contents. The United States Government retains and the publisher, by accepting the article for publication, acknowledges that the United States Government retains a non-exclusive, paid-up, irrevocable, world-wide license to publish or reproduce the published form of this article or allow others to do so, for United States Government purposes. The DOE will provide public access to these results of federally sponsored research in accordance with the DOE Public Access Plan <https://www.energy.gov/downloads/doe-public-access-plan>.

Acknowledgements I thank Stephen Jordan and Noah Shutty for helpful discussions on DQI and feedback on this work.

1 Introduction

MaxCut seeks to find a subset of vertices $S \subseteq V$ in a graph G maximizing the number of edges crossing S (i.e., edges with exactly one endpoint in S). It can also be equivalently viewed as a canonical constraint satisfaction problem in graphs by assigning boolean values to the vertices to maximize the number of edges with different values assigned to endpoints. MaxCut has sparked diverse avenues of research and has become a testbed problem for quantum-algorithmic approaches for solving or approximating discrete optimization problems, such as quantum annealing and the Quantum Approximate Optimization Algorithm (QAOA) [6]. Although approximation guarantees can be derived for the QAOA on MaxCut, rigorous approximation advantages remain elusive despite over a decade of extensive study. A quantum advantage for approximating general instances of MaxCut seems unlikely, since we can obtain



© Ojas Parekh;

licensed under Creative Commons License CC-BY 4.0

21st Conference on the Theory of Quantum Computation, Communication and Cryptography (TQC 2026).

Editors: Rotem Arnon and Aram W. Harrow; Article No. 6; pp. 6:1–6:12

Leibniz International Proceedings in Informatics



LIPICs Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

a 0.878...-approximation in classical polynomial time [9], and a $(0.878... + \epsilon)$ -approximation is NP-hard under the Unique Games Conjecture [14]. This suggests that it is hard for even efficient quantum algorithms to improve upon this classical threshold.

For “natural” discrete optimization problems, why would we *not* expect that there is an α such that an α -approximation is possible in classical polynomial time and an $\alpha + \epsilon$ -approximation is NP-hard? What kind of problems would admit a gap between classical approximability and NP-hardness of approximability, for a quantum advantage to possibly sneak in? In other words, why would a “natural” classical discrete optimization problem need a quantum algorithm to complete the story of its approximability?

A recent example of an unexpected provable exponential quantum approximation advantage is for the directed version of MaxCut [13, 12], which is arguably a natural problem. However, in this case the quantum advantage is enabled by considering space advantages in the streaming model instead of more traditional speedups. More broadly, perhaps certain kinds of restricted instances or regimes of natural discrete optimization problems are amenable to quantum approximation advantages.

Decoded Quantum Interferometry (DQI) potentially presents such an opportunity, since it relies heavily on special problem structure in a way that sets it apart from other classical or quantum approaches [11]. DQI is an application of Regev’s reduction [20] designed to work on constraint satisfaction problems for which an associated decoding problem is efficiently classically solvable. Moreover, it is based on Quantum Fourier Sampling [8], which does enable provable or conjectured quantum advantages in many settings. DQI is an exciting development, and there are some examples of problems for which DQI gives a better approximation guarantee than the best classical algorithm currently known. However, the demands that DQI puts on problem structure results in problem regimes unusual and understudied for classical algorithms. The most prominent candidate for a DQI advantage is for Optimal Polynomial Intersection, and there is essentially one nontrivial classical approximation algorithm, that is decades old, currently known for it.

Contributions

Understanding the power and limitations of DQI is thus an important challenge. We make some progress in establishing limitations for DQI, for the fundamental MaxCut problem. MaxCut is perhaps the simplest interesting special case of the Max-LINSAT problem considered by Jordan et al. [11]. Focusing on MaxCut allows for a streamlined presentation and analysis of DQI.

We show that DQI for MaxCut exhibits perhaps surprising behavior: on instance families where DQI attains *any* asymptotic improvement over the trivial random assignment approximation algorithm, it is not just that the approximation guarantee can be beaten classically – such instances can be solved *exactly* in classical polynomial time. To derive this result, we concretely demonstrate how the structure necessary for DQI can also be exploited classically. Our observations and classical algorithms are relatively simple, and MaxCut may be a useful guidepost for future DQI work.

DQI is generally not guaranteed to run in polynomial time and requires both feasibility of and an efficient algorithm for a related classical decoding problem. This limits both the problems and types of instances amenable to DQI. In contrast, DQI runs efficiently on any instance of MaxCut, since the decoding problem reduces to a perfect matching problem.

The key graph parameter for analysis of DQI for MaxCut is the *girth* of the unweighted input graph, which is the length of a shortest cycle. We assume the input comes from an infinite family of graphs with g , n , and m corresponding to girth, number of vertices, and number of edges. Based on the more general analysis of DQI in [11], we first observe:

► **Theorem 7.** *If the expected number of edges cut in a graph by DQI is $(\frac{1}{2} + \varepsilon)m$, for some constant $\varepsilon > 0$, the girth g must be linear in m .*

The trivial random assignment algorithm for MaxCut is expected to cut $\frac{1}{2}m$ edges, so the above theorem says that any improvement in the constant factor requires instances with very high girth. We show how this can be exploited classically:

► **Theorem 10.** *For a connected graph with girth g linear in m , the quantity $m - n + 1$ is constant.*

MaxCut can be solved exactly in classical polynomial time in this regime through a variety of approaches, and even the simple algorithm of cutting every edge in a spanning tree is guaranteed to cut $m - O(1)$ edges, since $m - n + 1$ is precisely the number of edges not in the spanning tree.

Related work

Patamawisut et al. describe a circuit implementation of DQI for MaxCut, using a different decoder based on Gauss-Jordan elimination instead of the aforementioned exact decoder based on perfect matching [19]. While Jordan et al. focus on more general cases of Max-LINSAT than MaxCut, they do observe a limitation on the performance of DQI for random instances of Max-2-XORSAT, which corresponds to $\{-1, +1\}$ -weighted MaxCut [11, Section 13.3]. In concurrent work with ours, Anschuetz, Gamarnik, and Lu identify obstructions for DQI on random instances of the more general Max- k -XORSAT problem, suggesting that a quantum advantage through DQI is not possible on typical random instances [1]. Another line of concurrent work, by Marwaha et al., gives evidence suggesting that the probability distribution from which DQI samples solutions may be hard to simulate classically [15]. However, even if DQI is doing something “inherently quantum” that is not efficiently simulable classically, this does not directly speak to whether efficient classical algorithms can outperform DQI. In fact MaxCut may be an example where DQI produces a distribution that is not efficiently producible classically, but as we show, independent polynomial-time classical algorithms that outperform it do exist.

Organization

The next section motivates and explains DQI for MaxCut without assuming any prior knowledge of DQI. Section 3 establishes Theorem 10 and demonstrates that MaxCut can be solved in classical polynomial time when the girth is linear. We close with a discussion in Section 4 on whether extensions of DQI might be able to provide approximation advantages for MaxCut.

2 Specializing Decoded Quantum Interferometry for MaxCut

It will be convenient to formulate MaxCut as a Boolean polynomial optimization problem. We will think of a cut as partitioning an unweighted graph, $G = (V, E)$ into two pieces by assigning either a $+1$ or -1 to each vertex, so that points $z \in \{-1, 1\}^V$ of the Boolean hypercube over V correspond to cuts. We take $n \stackrel{\text{def}}{=} |V|$, $m \stackrel{\text{def}}{=} |E|$, and g as the girth of G , which is the length of a shortest cycle. We implicitly assume that input instances G come

6:4 Limitations of Decoded Quantum Interferometry for MaxCut

from an infinite family $\mathcal{G}$ so that asymptotic quantities are well defined; in particular we assume that $\lim_{m \rightarrow \infty} g/m$ exists. This is a standard assumption and is also implicit in the asymptotic analysis in [11]. The number of edges cut by a z , called the *cut value* of z , is then

$$h(z) = \frac{1}{2} \sum_{ij \in E} 1 - z_i z_j.$$

MaxCut is equivalent to the Boolean polynomial optimization problem

$$\max_{z \in \{-1,1\}^V} h(z).$$

For the incidence vector $\gamma \in \{0,1\}^U$ of a set $S \subseteq U$, we take

$$x^\gamma \stackrel{\text{def}}{=} \prod_{i \in S} x_i^{\gamma_i}.$$

We can represent a polynomial over the Boolean hypercube,

$$p(z) = \sum_{\alpha \in \{0,1\}^V} p_\alpha z^\alpha$$

as a quantum state in two different ways:

- Encode the values of p over inputs as amplitudes

$$|p\rangle = \text{unit} \left(\sum_{z \in \{-1,1\}^V} p(z) |z\rangle \right),$$

where $|z\rangle \stackrel{\text{def}}{=} |\frac{1-z_1}{2} \frac{1-z_2}{2} \dots\rangle$ and $\text{unit}(\cdot)$ outputs a normalized state, or

- Encode the coefficients of p over monomials as amplitudes

$$|\hat{p}\rangle = \text{unit} \left(\sum_{\alpha \in \{0,1\}^V} p_\alpha |\alpha\rangle \right).$$

These two encodings are related by the Boolean Fourier transform, also known as the Hadamard transform (see [17] for Fourier analysis of Boolean functions). Hadamard gates applied on all qubits enact this transform on a quantum computer, and we have $H^{\otimes n} |p\rangle = |\hat{p}\rangle$ and $H^{\otimes n} |\hat{p}\rangle = |p\rangle$.

Quantum Fourier Sampling

The above observation gives a simple quantum algorithm, called Quantum Fourier Sampling (QFS) [8], for trying to find a cut z with relatively large value $h(z)$:

► **Algorithm 1** (Quantum Fourier Sampling).

1. *Prepare* $|\hat{h}\rangle$
2. *Produce* $|h\rangle = H^{\otimes n} |\hat{h}\rangle$
3. *Measure* $|h\rangle$ in the computational basis

This algorithm allows sampling a cut z with probability proportional to $h^2(z)$, so that the cut value dictates the probability of picking a cut, as we would want. We will discuss Step 1 in more detail below, and for now we note that it can be executed efficiently since h is an explicit degree-2 polynomial [8, Lemma 1].

What kind of approximation guarantee does this give for MaxCut? (Spoiler alert: it's not so great). The expected number of edges cut is $\frac{1}{2}m + O(1)$. This is just a slight improvement over the expectation of $\frac{1}{2}m$ edges achieved by the random assignment where each vertex is independently assigned to a side of the cut with probability $\frac{1}{2}$.

2.1 Decoded Quantum Interferometry

What if we could sample cuts from a distribution proportional to some higher power or maybe even a higher degree polynomial of the objective h ? This could amplify the probabilities of good cuts, and we might expect a better approximation guarantee.

This is where DQI steps in. Sampling from higher degree distributions is indeed possible, but the tradeoff is an increase in complexity of preparing the corresponding state in Step 1 of QFS. This leaves the questions: (i) how do we find and analyze a good polynomial of h , and (ii) can we prepare the corresponding state efficiently? DQI answers both of these questions.

For (i), the observation in [11] is that if we let $y_{ij} \stackrel{\text{def}}{=} z_i z_j$, then h is a symmetric polynomial in the variables y since it only depends on $g(y) \stackrel{\text{def}}{=} \sum_{ij \in E} y_{ij}$. The polynomials

$$g_k(y) \stackrel{\text{def}}{=} \sum_{\substack{\beta \in \{0,1\}^E \\ |\beta|=k}} y^\beta$$

are a basis for the symmetric polynomials. Hence, any degree- l polynomial of h , which is also a degree- l polynomial in the variables y , can be expressed as

$$q(y) = \sum_{k=0}^l \mu_k g_k(y) = \sum_{\beta \in \mathcal{E}_l} q_\beta y^\beta = \sum_{\alpha \in \{0,1\}^V} q'_\alpha z^\alpha, \quad (1)$$

where $\mathcal{E}_k \stackrel{\text{def}}{=} \{\beta \in \{0,1\}^E \mid |\beta| \leq k\}$. The idea is then to find a good polynomial q and run Algorithm 1 on it to sample a cut z with probability proportional to $q^2(z)$. The coefficients μ_k of a q that maximize the expected cut value, $\mathbb{E}_{z \sim q^2}[h(z)]$, can be found classically in polynomial time by solving a maximum-eigenvector problem for an $(l+1) \times (l+1)$ tridiagonal matrix (see Lemma 5). In light of this, we can think of the input to DQI as the unweighted graph G and the degree parameter l , where the approximation guarantee of DQI monotonically increases with l .

How large does l need to be? A distribution supported only on optimal cuts is proportional to

$$h^*(y) = \prod_{i=2}^s (h(y) - \lambda_i),$$

where $\lambda_1 > \lambda_2 > \dots > \lambda_s$ are the possible cut values $h(y)$ can attain. The symmetric polynomial h^* , which is proportional to $(h^*)^2$, has degree at most m since the cut value $h(y) \in [0, m]$ for all y . Hence, taking $l = m$ would solve MaxCut, and it suffices to choose $l \leq m$. This also suggests that DQI should not run in polynomial time when $l = m$, since MaxCut is NP-complete, and it is widely believed that polynomial-time quantum algorithms cannot solve NP-complete problems. DQI gives a better expected approximation guarantee than the best classical algorithm currently known for certain problems in the regime where $l = \Theta(n) = \Theta(m)$ (for more general constraint satisfaction problems, n and m are the number of variables and constraints).

Preparing the initial state for QFS

We need to be able to prepare $|\hat{q}\rangle$ in polynomial time. We can do so readily in a Hilbert space where we have m qubits corresponding to E . First note that g_k can be represented using a Dicke state:

$$|\hat{g}_k\rangle_E = \frac{1}{\sqrt{\binom{m}{k}}} \sum_{\substack{\beta \in \{0,1\}^E \\ |\beta|=k}} |\beta\rangle,$$

6:6 Limitations of Decoded Quantum Interferometry for MaxCut

where the subscript on the ket identifies the set that the qubits correspond to. Then we obtain, as a superposition of Dicke states

$$|\hat{q}\rangle_E = \text{unit} \left(\sum_{k=0}^l \mu_k \sqrt{\binom{m}{k}} |\hat{g}_k\rangle \right) = \text{unit} \left(\sum_{\beta \in \mathcal{E}_l} q_\beta |\beta\rangle \right),$$

which can be prepared efficiently as described in [11]. However, in order to apply Quantum Fourier Sampling, we instead need the state

$$|\hat{q}\rangle_V = \text{unit} \left(\sum_{\alpha \in \{0,1\}^V} q'_\alpha |\alpha\rangle \right).$$

How can we construct $|\hat{q}\rangle_V$ from $|\hat{q}\rangle_E$? Since $y_{ij} = z_i z_j$, we get a natural map f from $\beta \in \{0,1\}^E$ to $\alpha \in \{0,1\}^V$ as

$$y^\beta = \prod_{\substack{ij \in E: \\ \beta_{ij}=1}} z_i z_j = z^{f(\beta)},$$

where $z_i^2 = 1$ for all i is employed to reduce the monomial $z^{f(\beta)}$ so that $f(\beta) \in \{0,1\}^V$. We think of β as an incidence vector representing a subgraph (i.e., a subset of edges) and the *parity vector*, $f(\beta)$ as indicating the parity of the degree of vertex in V in the subgraph β (vertices may have no incident edges). We can see that f is not injective in general by letting G be a cycle on three vertices, for which $f(1,1,1) = f(0,0,0) = (0,0,0)$. In other words, all vertices in V have even degree in both G itself and the empty subgraph.

► **Lemma 1** (specialized for MaxCut from [11]). *Suppose that G is a graph where each parity vector $\alpha \in \{0,1\}^V$ arises from at most one subgraph $\beta \in \mathcal{E}_l$. Then f is injective (on the domain $\mathcal{E}_l$), and the following algorithm may be used to produce $|\hat{q}\rangle_V$.*

1. Prepare $|\hat{q}\rangle_E |0\rangle_V = \text{unit} \left(\sum_{\beta \in \mathcal{E}_l} q_\beta |\beta\rangle_E |0\rangle_V \right)$
2. Apply f coherently to get: $\text{unit} \left(\sum_{\beta \in \mathcal{E}_l} q_\beta |\beta\rangle_E |f(\beta)\rangle_V \right)$
3. Uncompute the E register using f^{-1} to yield: $|\psi\rangle \stackrel{\text{def}}{=} \text{unit} \left(\sum_{\beta \in \mathcal{E}_l} q_\beta |0\rangle_E |f(\beta)\rangle_V \right)$

Proof. We see from Equation (1) that

$$q'_\alpha = \sum_{\substack{\beta \in \mathcal{E}_l: \\ f(\beta)=\alpha}} q_\beta, \tag{2}$$

with $q'_\alpha = 0$ if there is no $\beta \in \mathcal{E}_l$ such that $f(\beta) = \alpha$. By Equation (2), discarding the E register from $|\psi\rangle$ produces the desired state:

$$\text{unit} \left(\sum_{\beta \in \mathcal{E}_l} q_\beta |f(\beta)\rangle_V \right) = \text{unit} \left(\sum_{\alpha \in \{0,1\}^V} q'_\alpha |\alpha\rangle \right). \quad \blacktriangleleft$$

DQI uses the algorithm of Lemma 1 and thus requires f to be injective. What kind of structural assumption does this impose on the input G, l ?

► **Lemma 2.** *Suppose we have a graph G and $l \in \mathbb{N}$. Then f is injective on $\mathcal{E}_l$ if and only if G has girth at least $2l + 1$.*

Proof. First we observe that f is linear when viewed as a function on $\mathbb{F}_2^E$. Let $B \in \mathbb{F}_2^{E \times V}$ be the edge-vertex incidence matrix of G , with $B_{e,v} = 1$ if and only if the edge e is incident to the vertex v . Then $f(\beta) = \alpha$ if and only if $\alpha = B^T \beta$, with α and β taken as vectors over $\mathbb{F}_2$.

We show that if f is not injective then the girth of G is at most $2l$. Suppose there are two distinct subgraphs $\beta, \beta' \in \mathcal{E}_l$ with $f(\beta) = f(\beta')$. Then, in $\mathbb{F}_2^E$, $\gamma \stackrel{\text{def}}{=} \beta + \beta' \neq \mathbf{0}$. The parity vector of γ is $\mathbf{0}$, since $f(\gamma) = f(\beta) + f(\beta') = \mathbf{0}$. This means that the nonempty subgraph corresponding to γ has even degree at all vertices and has at most $2l$ edges. This subgraph must therefore contain a cycle of size at most $2l$. To see this, one can start at a vertex and keep moving to other vertices using previously unused edges. Since all degrees are even, it will always be possible to leave a vertex that is visited. Since the graph is finite, one must eventually revisit a vertex, resulting in a cycle.

Now we show the opposite direction. Suppose there is a cycle with at most $2l$ edges. Break the cycle into two edge-disjoint paths of length at most l , both between the same pair of vertices i and j on the cycle. Represent these paths by incidence vectors $\beta, \beta' \in \mathcal{E}_l$. Then f is not injective since $f(\beta) = f(\beta') = \alpha$, where $\alpha_i = \alpha_j = 1$ and all other entries are 0. ◀

So for MaxCut there is a simple graph-theoretic characterization of the structure DQI requires. It makes sense to pick the largest l such that f remains injective; however, we also need an efficient classical algorithm to implement f^{-1} , which may limit the choice of l . For MaxCut, it turns out that f^{-1} can be computed in classical polynomial time for any value of l , as we show in the next section. In this case l can just be taken to be the largest integer so that G has girth at least $2l + 1$. We are now in position to present the full DQI algorithm for MaxCut.

► **Algorithm 2** (Decoded Quantum Interferometry for MaxCut).

Input: *unweighted graph* $G = (V, E)$

1. *Compute the girth* g *of* G , *and set* $l \stackrel{\text{def}}{=} \lfloor \frac{g-1}{2} \rfloor$
2. *Solve an* $(l+1) \times (l+1)$ *maximum eigenvector problem to find the coefficients of an optimal degree-* l *symmetric polynomial* q *in* y
3. *Run the algorithm of Lemma 1 to prepare* $|\hat{q}\rangle_V$
4. *Run QFS (Algorithm 1) on* $|\hat{q}\rangle_V$ *to sample a cut* z *with probability proportional to* $q^2(z)$

2.2 Solving the classical portion of DQI

Solving the following problem suffices to implement the uncompute step (Step 3) of the algorithm of Lemma 1.

► **Problem 1** (DQI uncomputation for MaxCut).

Input: *A parameter* $l \in \mathbb{N}$, *an unweighted graph* $G = (V, E)$, *and a vector* $\alpha \in \mathbb{F}_2^V$

Output: *A subgraph with at most* l *edges, represented as* $\beta \in \mathbb{F}_2^E$, *whose degree parities match* α *(i.e.,* $|\beta| \leq l$ *and* $f(\beta) \stackrel{\text{def}}{=} B^T \beta = \alpha$, *where* $B \in \mathbb{F}_2^{E \times V}$ *is the edge-vertex incidence matrix of* G *)*

Remark: *Additionally assuming that* G *has girth at least* $2l + 1$ *guarantees a unique solution for feasible instances by Lemma 2.*

► **Remark 3** (Classical decoding in DQI). DQI is originally framed in terms of a classical decoding problem for an error-correcting code. For MaxCut this perspective stems from the edge-incidence matrix B . The column space of B generates the *cut space* of G , which is the span over $\mathbb{F}_2$, of incidence vectors of cuts in G . Since any cycle crosses a cut an even number of times, the cut space is orthogonal to the *cycle space*, generated in the same fashion by incidence vectors of cycles. This space may be viewed as a cycle code space and is

characterized as incidence vectors of subgraphs of G which have even degree at every vertex. The parity-check matrix of this code is B^T , and the girth of G is the distance of the code. This code is usually called a graph code or cycle code.

In this picture, Problem 1 is about finding a short vector generating a given syndrome, and as observed in [11] it can also be solved as a more standard maximum-likelihood decoding problem. Finally, observe the following duality: for DQI to solve an optimization problem over cuts (MaxCut), it must solve a kind of decoding problem over the dual cycle space (Problem 1). See, e.g., Section 1.9 of [3] for more on the cut and cycle spaces of a graph.

We illustrate how Problem 1 can be cast as a T -join problem [4], which can be solved in polynomial time by a reduction to a perfect matching problem (see, e.g., [21, Theorem 29.1]).

► **Problem 2** (Minimum T -join).

Input: An unweighted graph $G = (V, E)$ and a subset of vertices $T \subseteq V$

Output: A subgraph H of G , with the minimum number of edges, such that set of odd-degree nodes in H is precisely T

► **Fact 1.** Problem 1 is polynomial-time reducible to Problem 2.

Proof. Given an instance of Problem 1, set $T \stackrel{\text{def}}{=} \{i \in V \mid \alpha_i = 1\}$ and solve Problem 2 on G, T . If this instance of Problem 2 is infeasible, or if H has more than l edges, then Problem 1 is infeasible. Otherwise, take β as the incidence vector of H . ◀

This is the last piece in establishing:

► **Theorem 4.** DQI for MaxCut (Algorithm 2) is correct and runs in polynomial time on any input graph G .

Proof. The girth of an unweighted undirected graph can be found by running n breadth-first searches, one rooted at each vertex [10], so Steps 1 and 2 run in classical polynomial time. Since $g \geq 2l + 1$, f is injective by Lemma 2, and Step 3 correctly produces $|\hat{q}\rangle_V$. This step also runs in polynomial time since Problem 1 runs in classical polynomial time by Fact 1. QFS (Step 4) runs in polynomial time and samples from q^2 as explained in [8]. The fact that Step 2 produces an optimal symmetric degree- l polynomial q comes from the proof of [11, Lemma 9.2], although this is not necessary to prove the approximation guarantee of Algorithm 2. ◀

We now move on to considering the approximation ratio that Algorithm 2 guarantees.

2.3 Approximation guarantee

The expected number of edges cut by DQI beyond the random-assignment guarantee of $\frac{1}{2}m$ is captured by the maximum eigenvalue of a tridiagonal matrix:

► **Lemma 5** (Specialization of Lemma 9.2 in [11] for MaxCut). *Let q be the polynomial obtained in Step 2 of Algorithm 2. Then the expected number of edges cut by Algorithm 2 is*

$$\mathbb{E}_{z \sim q^2}[h(z)] = \frac{1}{2}(m + \lambda_{\max}(A^{(m,l)})), \quad (3)$$

where

$$A^{(m,l)} \stackrel{\text{def}}{=} \begin{bmatrix} & a_1 & & & \\ a_1 & & a_2 & & \\ & a_2 & & \ddots & \\ & & \ddots & & a_l \\ & & & a_l & \end{bmatrix},$$

with $a_k = \sqrt{k(m-k+1)}$ for $1 \leq k \leq l$.

We show that DQI needs instances with $l = \Omega(m)$ in order to attain a asymptotically nontrivial approximation guarantee.

► **Lemma 6.** *For matrices $A^{(m,l)}$ with $l = o(m)$, $\lambda_{\max}(A^{(m,l)}) = o(m)$.*

Proof. Since $A^{(m,l)}$ is nonnegative, $\lambda_{\max}(A^{(m,l)}) \leq \sqrt{m} \lambda_{\max}(B^{(l)})$, where

$$B^{(l)} \stackrel{\text{def}}{=} \begin{bmatrix} & \sqrt{1} & & & \\ \sqrt{1} & & \sqrt{2} & & \\ & \sqrt{2} & & \ddots & \\ & & \ddots & & \sqrt{l} \\ & & & \sqrt{l} & \end{bmatrix}.$$

The matrix, $\frac{1}{\sqrt{2}}B^{(l)}$ is the $(l+1) \times (l+1)$ Jacobi matrix corresponding to the three-term recurrence relation for the normalized monic Hermite polynomials [16, 3.5(vi)]. The maximum eigenvalue of $\frac{1}{\sqrt{2}}B^{(l)}$ is consequently the largest zero of the degree- $(l+1)$ monic Hermite polynomial, which is $O(\sqrt{l})$ [16, 18.16(v)]. Thus $\lambda_{\max}(A^{(m,l)}) = O(\sqrt{ml})$. Since $l = o(m)$, we have $O(\sqrt{ml}) = o(m)$. ◀

From the two lemmas above, we have:

► **Theorem 7.** *If the expected number of edges cut by DQI in a graph $G \in \mathcal{G}$ is $(\frac{1}{2} + \varepsilon)m$, for some constant $\varepsilon > 0$, the girth g must be linear in m .*

Proof. If we do not have linear girth, $g = \Omega(m)$, then $g = o(m)$ since $\lim_{m \rightarrow \infty} g/m$ exists. In this case $l \leq \frac{g-1}{2} = o(m)$, and by the above lemmas, the expected number of edges cut by DQI is $\frac{1}{2}(1 + o(1))m$. ◀

► **Remark 8.** The approximation guarantee for DQI is expressed with respect to the trivial upper bound, m on the optimal cut since DQI does not leverage more refined upper bounds such as those arising from convex programs. However, this means that DQI provides a lower bound on the number of edges in a maximum cut and more generally on the number of satisfiable constraints in a constraint satisfaction problem, even when DQI is not guaranteed to run in polynomial time.

3 Classical solvability of high-girth instances

High girth has been previously exploited for solving MaxCut both classically and using the QAOA [5, 23, 7]; however, “high” in such contexts typically refers to constant or perhaps logarithmic girth, as expected in some kinds of random graphs. In contrast, in order for DQI to guarantee $(\frac{1}{2} + \varepsilon)m$ edges in expectation, we know that we must have girth $g = \Theta(n) = \Theta(m)$. We show that in this regime, MaxCut can be solved exactly in polynomial time. In fact, the approaches we describe can also solve weighted instances, while it is nontrivial to extend DQI to handle weighted instances.

► **Lemma 9.** *Suppose $G \in \mathcal{G}$ is a connected graph with girth $g = \Omega(m)$. Then the vertices of G can be partitioned into vertex-disjoint trees $T_1, \dots, T_t$ where: (i) $t = O(1)$, and (ii) there is at most one edge in G between any two T_i and T_j .*

Proof. Set $d \stackrel{\text{def}}{=} \lfloor \frac{g-3}{4} \rfloor$, and find a spanning tree T of G . Then run the following algorithm.

While T has depth at least d :

1. Find the deepest leaf u in T , and let S_i be the subtree rooted at the ancestor of u at distance d from u

2. Let T_i be the subgraph of G induced by the vertices of S_i
3. Remove S_i from T and set $i = i + 1$

When the loop has terminated, set T_i to be subgraph induced by T itself. Let $t \stackrel{\text{def}}{=} i$ at this point.

We are guaranteed that each tree S_i , except possibly S_t , has depth exactly d , since we picked u to be the deepest leaf in T , and therefore in S_i , in each iteration. The last tree S_t has depth at most d .

To prove (i), since the trees $S_1, \dots, S_{t-1}$ have depth d , these disjoint trees each have at least d edges. This gives us $t = O(1)$ since $g = \Omega(m)$ implies $d = \Omega(m)$ (the hidden constants for t and d only depend on those for g).

Next observe that each T_i can contain no edge of $G - T$ and is consequently a tree. Since each T_i is connected by edges in T and has depth at most d , if one contained some edge in $G - T$ there would be a cycle with at most $2d + 1 < g$ edges, contradicting that the girth is g . The proof of (ii) is similar to this. Two edges between distinct T_i and T_j would create a cycle of length at most $4d + 2 < g$. ◀

► **Theorem 10.** *For a connected graph $G \in \mathcal{G}$ with girth $g = \Omega(m)$, the quantity $\mu \stackrel{\text{def}}{=} m - n + 1$ is constant.*

Proof. Let T be a spanning tree, and construct $T_1, \dots, T_t$ from T as specified in Lemma 9. Since each T_i is a subgraph of T , each edge in $G - T$ must go between distinct T_i and T_j . Conditions (i) and (ii) then imply that there is only a constant number of such edges. In other words, μ is constant. ◀

► **Corollary 11.** *MaxCut can be solved in polynomial time on a graph $G \in \mathcal{G}$ with girth $g = \Omega(m)$.*

Proof. Each component of G must also have girth $\Omega(m)$, so we can apply Theorem 10 to each component individually. MaxCut can be solved in time $O(m + n) \min\{2^{m/5}, 2^{(m-n)/2}\}$ on a connected graph [22, Theorem 2]. ◀

Other approaches for high-girth MaxCut

Theorem 10 enables other approaches for solving MaxCut in polynomial time. The fundamental cycles of a spanning tree T are the set of the μ unique cycles created by adding each of the μ non-tree edges to T . These cycles form a basis for all cycles over $\mathbb{F}_2^E$ (see Remark 3). Since μ is constant for graphs with linear girth, so are the total number of cycles, which is at most 2^μ . This allows solving positive-weighted MaxCut as a binary integer program with a constant number of constraints:

$$\begin{aligned} \max \quad & \sum_{ij \in E} w_{ij} x_{ij} \\ \text{subject to} \quad & \sum_{ij \in C} x_{ij} \leq |C| - 1, \forall \text{ odd cycles } C \\ & x_{ij} \in \{0, 1\}, \forall ij \in E, \end{aligned}$$

which can be done in polynomial time [18]. The constraints ensure that a bipartite subgraph is selected by preventing picking all the edges of any odd cycle.

Perhaps the simplest way to solve high-girth instances of MaxCut is via a dynamic program. Pick a spanning tree T . Cutting all the edges of the tree already gives a cut of size $m - O(1)$, since μ is constant. Enumerate all ways of assigning -1 or $+1$ to the endpoints of

the μ non-tree edges. For each assignment, one can solve a dynamic program that works bottom up from the leaves to find the best cut of the edges of T assuming that some nodes have been fixed by an assignment. This works for weighted MaxCut as well.

4 Discussion

The limitation presented here applies when the decoding portion of DQI is always required to succeed. A natural question is whether some extension of DQI, including quantum decoding, might be able to get around the barriers we presented. One can consider probabilistic decoding approaches for DQI that need not always succeed [11, 2]. Such approaches can conceivably tolerate some cycles of smaller girth, possibly extending the kinds of instances of MaxCut that DQI can address. However, some of the classical approaches from Section 3 work even when $\mu = \Theta(\log n)$ rather than $\mu = O(1)$. This, for instance, implies that MaxCut on any graph with a logarithmic number of cycles (of any size) can be solved efficiently. Some instances with a polynomial number of cycles may also be solvable, since it is possible to have a polynomial cycles with a fundamental cycle basis of size $\mu = \Theta(\log n)$. Thus classical approaches seem fairly robust in terms of handling cycles, and it is not clear if DQI would be able to do better.

Is MaxCut an isolated case, or are there more general ways for classical algorithms to leverage the same structure that DQI does? The particular property we leverage for MaxCut is special and seems difficult to generalize to other types of Max-LINSAT problems. The prime candidates for a DQI advantage, such as Optimal Polynomial Intersection, should be further vetted by attempts at designing better classical approximation algorithms in the regimes where running DQI is feasible and efficient.

References

- 1 Eric R. Anschuetz, David Gamarnik, and Jonathan Z. Lu. Decoded quantum interferometry requires structure. *arXiv*, 2025. [arXiv:2509.14509](#).
- 2 André Chailloux and Jean-Pierre Tillich. Quantum advantage from soft decoders. In *Proceedings of the 57th Annual ACM Symposium on Theory of Computing (STOC 2025)*. ACM, 2025. Extended version available as [arXiv:2411.12553](#). [doi:10.1145/3717823.3718319](#).
- 3 Reinhard Diestel. *Graph theory*, volume 173 of *Graduate Texts in Mathematics*. Springer, 6 edition, 2025. [doi:10.1007/978-3-662-70107-2](#).
- 4 Jack Edmonds and Ellis L. Johnson. Matching, Euler tours and the Chinese postman. *Math. Program.*, 5:88–124, 1973. [doi:10.1007/BF01580113](#).
- 5 Ahmed El Alaoui, Andrea Montanari, and Mark Sellke. Local algorithms for MaxCut and minimum bisection on locally treelike regular graphs of large degree. *Random Struct. Algorithms*, 63(3):580–645, 2023. [doi:10.1002/rsa.21110](#).
- 6 Edward Farhi, Jeffrey Goldstone, and Sam Gutmann. A quantum approximate optimization algorithm. *arXiv*, 2014. [arXiv:1411.4028](#).
- 7 Edward Farhi, Sam Gutmann, Daniel Ranard, and Benjamin Villalonga. Lower bounding the MaxCut of high girth 3-regular graphs using the QAOA. *arXiv*, 2025. [arXiv:2503.12789](#).
- 8 Bill Fefferman and Christopher Umans. On the power of quantum fourier sampling. In *Proceedings of the 11th Conference on the Theory of Quantum Computation, Communication and Cryptography (TQC 2016)*, pages 1–19. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2016. [doi:10.4230/LIPIcs.TQC.2016.1](#).
- 9 Michel X. Goemans and David P. Williamson. Improved approximation algorithms for MaxCut and satisfiability problems using semidefinite programming. *J. ACM*, 42(6):1115–1145, 1995. [doi:10.1145/227683.227684](#).

- 10 Alon Itai and Michael Rodeh. Finding a minimum circuit in a graph. *SIAM J. Comput.*, 7(4):413–423, 1978. doi:10.1137/0207033.
- 11 Stephen P. Jordan, Noah Shutty, Mary Wootters, Adam Zalcman, Alexander Schmidhuber, Robbie King, Sergei V. Isakov, Tanuj Khattar, and Ryan Babbush. Optimization by decoded quantum interferometry. *arXiv*, 2024. arXiv:2408.08292.
- 12 John Kallaugher, Ojas Parekh, and Nadezhda Voronova. Exponential quantum space advantage for approximating maximum directed cut in the streaming model. In *Proceedings of the 56th Annual ACM Symposium on Theory of Computing (STOC 2024)*, pages 1805–1815. ACM, 2024. doi:10.1145/3618260.3649656.
- 13 John Kallaugher, Ojas Parekh, and Nadezhda Voronova. How to design a quantum streaming algorithm without knowing anything about quantum computing. In *Proceedings of the 2025 Symposium on Simplicity in Algorithms (SOSA 2025)*, pages 9–45. Society for Industrial and Applied Mathematics, 2025. doi:10.1137/1.9781611977998.2.
- 14 Subhash Khot, Guy Kindler, Elchanan Mossel, and Ryan O'Donnell. Optimal inapproximability results for MaxCut and other 2-variable CSPs? *SIAM J. Comput.*, 37(1):319–357, 2007. doi:10.1137/S0097539705447372.
- 15 Kunal Marwaha, Bill Fefferman, Alexandru Gheorghiu, and Vojtech Havlicek. On the complexity of decoded quantum interferometry. *arXiv*, 2025. doi:10.48550/arXiv.2509.14443.
- 16 Nist digital library of mathematical functions, 2025. Release 1.2.3 of 2025-09-15, F. W. J. Olver, A. B. Olde Daalhuis, D. W. Lozier, B. I. Schneider, R. F. Boisvert, C. W. Clark, B. R. Miller, B. V. Saunders, H. S. Cohl, and M. A. McClain, eds. URL: <https://dlmf.nist.gov/>.
- 17 Ryan O'Donnell. Analysis of Boolean functions. *arXiv*, 2021. arXiv:2105.10386.
- 18 Christos H. Papadimitriou. On the complexity of Integer Programming. *J. ACM*, 28(4):765–768, 1981. doi:10.1145/322276.322287.
- 19 Natchapol Patamawisut, Naphan Benchasattabuse, Michal Hajdušek, and Rodney Van Meter. Quantum circuit design for decoded quantum interferometry. *arXiv*, 2025. arXiv:2504.18334.
- 20 Oded Regev. On lattices, learning with errors, random linear codes, and cryptography. In *Proceedings of the 37th Annual ACM Symposium on Theory of Computing (STOC 2005)*, pages 84–93. ACM, 2005. doi:10.1145/1060590.1060603.
- 21 Alexander Schrijver. *Combinatorial optimization: polyhedra and efficiency*, volume 24 of *Algorithms and Combinatorics*. Springer, 2003. doi:10.1007/978-3-540-68279-0.
- 22 Alexander D. Scott and Gregory B. Sorkin. Faster algorithms for MaxCut and MaxCSP, with polynomial expected time for sparse instances. In *Proceedings of the International Workshop on Randomization and Approximation Techniques in Computer Science (RANDOM-APPROX 2003)*, pages 382–395. Springer, 2003. doi:10.1007/978-3-540-45198-1_29.
- 23 Jessica K. Thompson, Ojas Parekh, and Kunal Marwaha. An explicit vector algorithm for high-girth MaxCut. In *Proceedings of the SIAM Symposium on Simplicity in Algorithms (SOSA 2022)*, pages 148–159. Society for Industrial and Applied Mathematics, 2022. doi:10.1137/1.9781611977066.17.

On the Complexity of the Circuit Width Problem

Zhengfeng Ji 

Department of Computer Science and Technology, Tsinghua University, Beijing, China

Yinchen Liu 

Institute for Interdisciplinary Information Sciences, Tsinghua University, Beijing, China

Zhe'ou Zhou 

Institute for Interdisciplinary Information Sciences, Tsinghua University, Beijing, China

Abstract

Montanaro associated to every quantum circuit over the gate set $\{H, Z, CZ, CCZ\}$ a degree-three polynomial over $\mathbb{F}_2$, and showed that the strong simulation cost of the circuit is governed by the minimum number of qubit wires needed to realize that polynomial. We study this minimum-width problem. We prove that deciding whether a degree-three polynomial has width at most k is NP-complete. The reduction is parsimonious enough to imply that, for every $\alpha < 49/48$, the corresponding gap version is NP-hard. We also prove the same inapproximability threshold for degree-two polynomials by a twin-copy replacement of the cubic constraints. On the positive side, we give a nondeterministic search algorithm using only $O(k \log(en/k))$ nondeterministic bits, and a deterministic fixed-parameter algorithm running in $k^{6k+o(k)}n + O(m)$ time on an input with n symbols and m monomials.

2012 ACM Subject Classification Theory of computation $\rightarrow$ Quantum complexity theory; Theory of computation $\rightarrow$ Problems, reductions and completeness; Theory of computation $\rightarrow$ Parameterized complexity and exact algorithms

Keywords and phrases IQP circuits, circuit width, NP-completeness, approximation hardness, parameterized algorithms

Digital Object Identifier 10.4230/LIPIcs.TQC.2026.7

Related Version *Full Version:* <https://arxiv.org/abs/2606.18201> [18]

Funding *Zhengfeng Ji:* National Natural Science Foundation of China (Grant No. 12347104), National Key Research and Development Program of China (Grant No. 2023YFA1009403), Beijing Science and Technology Planning Project (Grant No. Z25110100810000), and Beijing Natural Science Foundation (Grant No. Z220002).

Yinchen Liu: Beijing Natural Science Foundation (Grant No. QY26408).

Acknowledgements We thank the anonymous reviewers for their helpful comments and suggestions, especially on the importance of better characterizing the degree-2 instances used in our hardness reduction and of clarifying the role of potentially tractable special cases.

1 Introduction

Quantum computation is widely believed to outperform classical computation on tasks such as simulating quantum systems [11] and factoring large integers [25]. Since Yao's equivalence theorem [27], the quantum circuit model has been the standard formalism for studying this power. A central theme is to relate circuit amplitudes to classical counting problems: quantum computation can be simulated using counting oracles, giving $BQP \subseteq PP \subseteq P^{\#P}$, and is closely connected to counting-based classes such as AWPP [3, 12]. Similar counting viewpoints appear in linear optics, Jones-polynomial approximation, Ising partition functions, IQP circuits, and boson sampling [1, 21, 13, 5, 2, 8].



© Zhengfeng Ji, Yinchen Liu, and Zhe'ou Zhou;
licensed under Creative Commons License CC-BY 4.0

21st Conference on the Theory of Quantum Computation, Communication and Cryptography (TQC 2026).

Editors: Rotem Arnon and Aram W. Harrow; Article No. 7; pp. 7:1–7:22

Leibniz International Proceedings in Informatics



LIPICs Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

For circuit amplitudes, this connection can be made explicit through an algebraic path-integral representation. Dawson et al. [9] showed that amplitudes of circuits built from Toffoli and Hadamard gates can be expressed as normalized counts of solutions to polynomial equations over $\mathbb{F}_2$. Montanaro [23] refined this framework for the gate set $\mathcal{Z} = \{H, Z, CZ, CCZ\}$. Given a circuit $\mathcal{C}$ on w qubits over $\mathcal{Z}$, one obtains a degree-3 polynomial $f_{\mathcal{C}}$ over $\mathbb{F}_2$, with no constant term, such that the amplitude $\langle 0^w | \mathcal{C} | 0^w \rangle$ is proportional to $\text{gap}(f_{\mathcal{C}})$, the difference between the number of zeros and ones of $f_{\mathcal{C}}$. Thus a BQP-complete quantity is encoded by a classical gap-counting problem whose cancellations reflect quantum interference.

This representation naturally leads to a width-minimization problem. Montanaro defined the width $w(f)$ of a polynomial f to be the minimum number of qubits in any circuit $\mathcal{C}$ over $\mathcal{Z}$ such that $f_{\mathcal{C}} = f$. If f has n variables and width $w(f)$, then a quantum computer can approximate $\text{gap}(f)$ to absolute error $2^{(n+w(f))/2}/3$. Hence smaller width gives better approximate counting precision. Efficiently computing, or even approximating, $w(f)$ would therefore provide a clean combinatorial route toward approximate counting for quantum amplitudes, and was posed by Montanaro as a natural open problem [23].

In this paper, we undertake a systematic study of the quantum circuit width problem, formally denoted as $\text{CIRCUITWIDTH}(f, k)$, which asks whether the width $w(f) \leq k$. We show that $\text{CIRCUITWIDTH}(f, k)$ is NP-complete, and that even approximating the circuit width within a small constant factor remains NP-hard. Furthermore, we extend our hardness results to degree-2 polynomials, demonstrating that both the circuit width problem and its approximate version remain NP-hard. This extension has broad implications because many quantum gate sets yield degree-2 polynomials: for instance, circuits using $\{H, \text{CS}\}$ or Clifford+ T gates produce polynomials with at most quadratic terms modulo 4 or 8 depending on the gate set [26]. Thus, our approximation hardness result establishes that the same computational intractability is universal across all these diverse gate sets. Moreover, since degree-2 polynomials without linear terms correspond to graphs, circuit width in this setting can be interpreted as a graph parameter related to treewidth or pathwidth, but carrying a quantum information-theoretic interpretation concerning the arrangement of Hadamard gates and diagonal two-qubit gates. Consequently, our results rule out efficient width optimization across all these practically relevant quantum computing scenarios on classical or quantum computers unless unlikely complexity-theoretic collapses occur. We further complement these results with algorithmic upper bounds, including a nondeterministic search algorithm with succinct witnesses and an efficient fixed-parameter tractable algorithm parameterized by the circuit width.

Our results have several consequences. First, they answer Montanaro's open question by showing that circuit-width minimization is a genuine computational bottleneck in the approximate-counting approach to quantum amplitudes. Second, circuit width provides a natural framework for minimizing qubit usage in quantum computation, with potential relevance to practical implementations and compilation of quantum algorithms. Our hardness results indicate that, in the worst case, even quantum computers cannot solve this problem efficiently or even approximately. Any further progress in this direction must therefore rely on heuristic methods and exploit the structure of circuits for specific algorithms. Finally, our hardness and fixed-parameter results suggest that circuit width behaves more like a structural parameter, analogous to treewidth for tensor-network simulation [22] or linear rank-width for decision-diagram simulation [6], than a direct measure of intrinsic quantum computational power.

1.1 Our results

We summarize our main results on the circuit width problem. The results are twofold: hardness results and algorithmic results.

For the hardness of circuit width, we rule out the possibility of efficiently computing the width of a degree-3 polynomial f with no constant term by proving that the problem of deciding $w(f) \leq k$ (denoted by $\text{CIRCUITWIDTH}(f, k)$) is NP-complete.

► **Theorem 1.1** (See also Corollary 4.10). *CIRCUITWIDTH is NP-complete.*

In fact, we are able to prove an inapproximability result: CIRCUITWIDTH is even NP-hard to approximate within a multiplicative factor of about 1.02 (even restricted to instances with $k = \Theta(n)$), and hence it is impossible to approximate the width efficiently (in BQP power) within additive $O(\log n)$ error unless $\text{NP} \subseteq \text{BQP}$.

► **Theorem 1.2** (Informal, see also Theorem 5.3). *For any constant $0 < \epsilon < 1/48$, it is NP-hard to distinguish degree-3 polynomials f with no constant term and $w(f) \leq k$ from those with $w(f) \geq (\frac{49}{48} - \epsilon)k$.*

The previous hardness results crucially rely on cubic terms (CCZ gates) to enforce complex geometric constraints. However, many quantum computational models naturally correspond to degree-2 polynomials. A natural question thus arises: does the approximation hardness persist when we restrict to *only* degree-2 polynomials? We answer this affirmatively by showing that approximation hardness persists with the same factor:

► **Theorem 1.3** (Informal, see Theorem 5.5). *The approximation hardness of CIRCUITWIDTH holds for degree-2 polynomials: for any constant $0 < \epsilon < 1/48$, it is NP-hard to distinguish quadratic f with $w(f) \leq k$ from those with $w(f) \geq (\frac{49}{48} - \epsilon)k$.*

Finally, from the viewpoint of conditional lower bounds, our reduction from 3-SAT also yields an exponential lower bound under the standard Exponential Time Hypothesis (ETH) [16]:

► **Theorem 1.4** (See also Theorem 4.11). *Assuming the Exponential Time Hypothesis (ETH), no algorithm can solve $\text{CIRCUITWIDTH}(f, k)$ in time $2^{o(n)}$.*

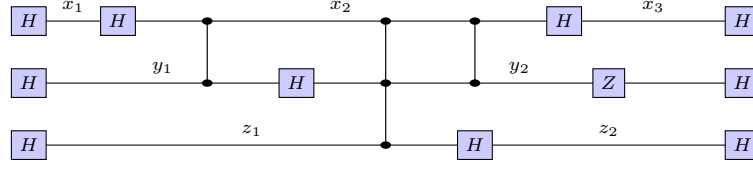
On the positive side, we provide a nondeterministic polynomial-time algorithm for the search version of $\text{CIRCUITWIDTH}(f, k)$ with improved witness complexity:

► **Theorem 1.5** (Informal, see Theorem 6.3). *There exists a nondeterministic polynomial-time algorithm that, given a witness of size $2 \log_2 \binom{n}{k} = O(k \log(en/k))$ bits, either constructs a quantum circuit on at most k qubits with associated polynomial f , or correctly concludes that no such circuit exists. Enumerating all endpoint witnesses gives a deterministic $\binom{n}{k}^2 \cdot \text{poly}(n)$ -time algorithm. When $k = \Theta(n)$, this is $2^{O(n)}$, matching the ETH lower bound in Theorem 1.4 up to constant factors in the exponent.*

Taking k as a constant, this yields an XP algorithm with runtime $n^{2k+O(1)}$. Moreover, from the perspective of parameterized complexity [10, 7], the nondeterministic algorithm can be easily transformed into a placement of CIRCUITWIDTH in the parameterized complexity class $\text{W}[1]$. A natural question then arises: is the problem $\text{W}[1]$ -complete, or is it fixed-parameter tractable (FPT) with respect to parameter k ? We answer the latter affirmatively:

► **Theorem 1.6** (Informal, see also Theorem 6.5). *$\text{CIRCUITWIDTH}(f, k)$ is fixed-parameter tractable with respect to parameter k . There exists an algorithm with runtime $k^{O(k)} \cdot n$ that solves the search version of $\text{CIRCUITWIDTH}(f, k)$.*

The concrete runtime is $k^{6k+o(k)} \cdot n$, which is faster than the XP algorithm when $k \leq n^{1/4-o(1)}$, making it more efficient and practical for polynomially small k .



■ **Figure 1** Circuit-to-polynomial map. The displayed circuit contributes $y_2 + x_1x_2 + x_2x_3 + y_1y_2 + z_1z_2 + x_2y_1 + x_2y_2 + x_2y_2z_1$.

1.2 Organization

Section 2 gives the main ideas of the reductions and algorithms. Section 3 records the circuit-polynomial correspondence, the interval model, and the complexity notation. Section 4 gives the 3-SAT reduction, including the main gadget definitions and proof sketches for completeness and soundness. Section 5 derives approximation hardness and the degree-two extension. Section 6 gives the structural characterization, the short nondeterministic witness, and the high-level fixed-parameter algorithm.

2 Technical overview

We first recall the circuit-polynomial correspondence. Normalize a circuit by placing an H gate at the beginning and end of every qubit wire. Each open wire segment between consecutive H gates receives a symbol. A Z gate on symbol u contributes u ; a CZ gate between two simultaneously active symbols u, v contributes uv ; a CCZ gate contributes uvw ; and an internal H gate between adjacent symbols u, v on the same wire also contributes uv . Thus the polynomial has the form

$$f_C(x) = \sum_{u \in \mathcal{S}} \alpha_u u + \sum_{\{u, v\} \subseteq \mathcal{S}} \beta_{uv} uv + \sum_{\{u, v, w\} \subseteq \mathcal{S}} \gamma_{uvw} uvw,$$

with coefficients in $\mathbb{F}_2$.

Hardness overview

For hardness, the important observation is the relation between the number of symbols, the number of internal H gates, and the number of wires. If a circuit has k wires and h internal Hadamard gates, then it has exactly $k + h$ symbols. A quadratic monomial can be supplied either by a CZ gate or by an internal H adjacency. In our reduction the number of symbols and the number of available quadratic monomials are arranged so that achieving the budget $k = 1 + 6m$ is possible only if all relevant quadratic monomials are used as Hadamard adjacencies. Once this happens, the polynomial no longer merely asks for certain overlaps: it forces a nearly rigid partition of the symbols into wires.

We distinguish variables from literals as follows. Symbols x, y, z name Boolean variables, while ℓ_x, ℓ_y, ℓ_z denote the corresponding literals, each either positive or negated. For an occurrence based on variable x , the local wire always uses $x_c, \widehat{x}_c, \bar{x}_c$; the literal ℓ_x is represented by x_c if it is positive and by $\bar{x}_c$ if it is negated. The same convention applies to y and z .

The reduction uses five kinds of objects. There is one base symbol b , whose support can be viewed as spanning the full time interval. For each clause c , a truth-marker wire contains three symbols $t_c, \widehat{t}_c, \widetilde{t}_c$, with t_c serving as the local truth reference. For each literal occurrence, a variable wire contains $x_c, \widehat{x}_c, \bar{x}_c$; the two possible orders of the end symbols

encode the truth value. Synchronization terms force all occurrences of the same Boolean variable to make the same choice. Finally, each clause is checked by two binary OR gadgets: p_c is the output of the first disjunction $\ell_x \vee \ell_y$, and o_c is the final output $p_c \vee \ell_z$ of the whole clause. The cubic term $bt_c o_c$ forces this final output to be true.

The proof of soundness has two layers. The first layer is a counting and isolation argument: in a width- $(1 + 6m)$ realization, the base symbol is isolated on one wire, and each clause contributes exactly six three-segment wires: one truth-marker wire, three variable-occurrence wires, and two OR wires. The second layer is geometric. Cubic monomials involving b force the other two symbols to overlap at a common time. Since b can be taken to be active for the whole interval, these terms become vertical overlap constraints. The OR wire can overlap the marker t_c exactly when at least one input literal does, and therefore every clause must be satisfied.

Approximation hardness is obtained by the same construction applied to MAX-3-SAT. If a formula leaves Δ clauses unsatisfied, then each such clause must spend one additional wire. Hence the optimum width is $1 + 6m + \Delta$. Håstad's 7/8 inapproximability theorem [14] gives $\Delta \geq (1/8 - o(1))m$, which turns into the threshold $(1 + 6m + m/8)/(1 + 6m) \rightarrow 49/48$.

The degree-two result replaces each cubic constraint by many quadratic constraints between twin copies. The key combinatorial fact is a survival lemma: in any near-optimal realization, for each symbol family, many copies are not used as exceptional connectors. Among surviving copies, the complete pairwise quadratic constraints simulate the pairwise intersections that a cubic term would have enforced. Helly's theorem for intervals then recovers a common intersection, so the degree-two instance preserves the original soundness argument.

Algorithmic overview

The algorithms use a structural characterization. Guessing a circuit is the same as partitioning the symbols into k ordered lists, one per wire, and then asking whether all overlaps and Hadamard adjacencies can be scheduled in a common time order. This can be checked by building a directed constraint graph on interval endpoints and testing acyclicity. The nondeterministic algorithm therefore only guesses the sets of first and last symbols. Given the right endpoints, a greedy procedure repeatedly extends an earliest-ending current head; a geometric argument shows that such a head has at most one possible unused neighbor. For the deterministic FPT algorithm, a width- k realization gives a path decomposition of the connection graph of width at most k . We run dynamic programming over a tree decomposition, storing only boundary wire order, coarse information about forgotten subpaths, and a boundary topological order. The transition verification is handled by the boundary invariant described in Section 6.4; the important point for the overview is that the state count is $k^{6k+o(k)}$ per bag.

3 Preliminaries

3.1 Associated polynomials and width

We use the circuit-to-polynomial representation of Montanaro [23]. A circuit over $\{H, Z, CZ, CCZ\}$ is first put into a normalized form by adding an H gate at the beginning and end of every qubit wire. A *symbol* is a maximal open wire segment between two consecutive Hadamard gates. We write $\mathcal{S}$ for the set of symbols and $\text{supp}(u) = (L_u, R_u)$ for the open time interval during which u is active.

The associated polynomial $f_C \in \mathbb{F}_2[\mathcal{S}]$ is obtained by adding one monomial for each gate interaction. A Z gate on u contributes u . A CZ gate on overlapping symbols u, v contributes uv . A CCZ gate on symbols u, v, w with a common time intersection contributes uvw . Finally, an internal Hadamard gate between two consecutive symbols u, v on the same wire also contributes the quadratic term uv . All sums are over $\mathbb{F}_2$, so identical monomials cancel in pairs.

► **Definition 3.1** (Support and overlap). *The support of a symbol u in a realizing circuit is the interval $\text{supp}(u) = (L_u, R_u)$. Two symbols u, v overlap if $L_u < R_v$ and $L_v < R_u$. A set of symbols has a common overlap if the intersection of their supports is nonempty.*

A cubic term requires a common overlap of three intervals. We repeatedly use the following special case of Helly's theorem.

► **Theorem 3.2** (Helly's theorem for intervals [15]). *For intervals on the line, if every two intervals in a finite family intersect, then all intervals in the family have a common intersection.*

In particular, for three symbols, pairwise overlap is equivalent to common overlap. This observation is what later allows us to replace cubic constraints by quadratic constraints between twin copies.

► **Definition 3.3** (Width). *For a degree-three polynomial $f \in \mathbb{F}_2[\mathcal{S}]$ with no constant term, the width $w(f)$ is the minimum number of qubit wires in a normalized circuit C over $\{H, Z, CZ, CCZ\}$ such that $f_C = f$.*

► **Definition 3.4** (Decision and search versions). *The decision problem $\text{CIRCUITWIDTH}(f, k)$ asks whether $w(f) \leq k$. The search version asks either to output a width- k circuit realizing f , or to certify that no such circuit exists.*

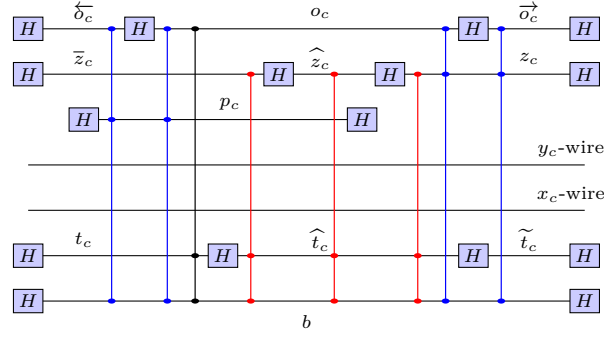
The input polynomial is specified by its nonzero monomials. We write $n = |\mathcal{S}|$ for the number of symbols and m for the number of monomials. In the search version, an output circuit may be represented combinatorially by ordered symbol lists, one list per qubit wire, together with the choice of which quadratic monomials are realized by internal Hadamard gates rather than CZ gates.

3.2 Complexity and algorithmic preliminaries

We use 3-SAT in the standard form where each clause contains three literals. For a formula Φ , let m be the number of clauses and let $\text{OPT}(\Phi)$ be the maximum number of clauses simultaneously satisfiable. The problem MAX-3-SAT asks for this optimum.

Håstad's theorem states that, for every $\varepsilon > 0$, it is NP-hard to distinguish satisfiable 3-CNF formulas from formulas in which at most a $(7/8 + \varepsilon)$ -fraction of clauses are satisfiable [14]. We also use the Exponential Time Hypothesis and the sparsification lemma in the standard consequence that linear-size 3-SAT cannot be solved in $2^{o(m)}$ time unless ETH fails [16, 17].

A parameterized algorithm is fixed-parameter tractable with respect to k if its running time is $g(k)n^{O(1)}$ for some computable function g [10, 7]. We use standard tree-decomposition terminology [24, 4, 7]. A tree decomposition of a graph G is a tree whose nodes carry bags of vertices so that every vertex appears in a connected set of bags and every edge is contained in some bag. The width is one less than the maximum bag size. A path decomposition is the special case in which the decomposition tree is a path. The treewidth $\text{tw}(G)$ is the minimum width of a tree decomposition of G , and the pathwidth $\text{pw}(G)$ is the minimum width of a path decomposition of G .



■ **Figure 2** Clause gadget for $c = (\ell_x \vee \ell_y \vee z)$. The figure shows the base and marker wires, the compressed first OR output p_c , the z wire, and the final OR wire. The columns through p_c and o_c are OR-gadget CCZ constraints; the columns through the z wire are marker–literal constraints; and the remaining column through t_c and o_c is the pinning term $bt_c o_c$.

4 The SAT reduction

Let Φ be a 3-CNF formula with clause set C , $|C| = m$. We construct a polynomial f_Φ and set the target width to $k = 1 + 6m$. The target should be read as one global wire plus six local wires per clause. The reduction is designed so that this budget is tight: any attempt to realize a quadratic term by a CZ gate instead of by an internal Hadamard adjacency would consume too much room and force an extra wire. Thus the width bound turns algebraic monomials into a rigid interval geometry.

We use the variable/literal convention from the overview: x, y, z are Boolean variables, while ℓ_x, ℓ_y, ℓ_z are literals. If $\ell_x = x$, its input symbol is x_c ; if $\ell_x = \bar{x}$, its input symbol is $\bar{x}_c$, and similarly for y, z .

There is one global base symbol b . Geometrically, b is the anchor that allows a cubic monomial buv to behave like a vertical overlap constraint between u and v : since the base wire may span the entire construction, requiring b, u, v to meet at a common time is equivalent to requiring the two local intervals u and v to meet. This is the mechanism that lets the reduction implement Boolean tests using interval geometry.

For each clause $c = (\ell_x \vee \ell_y \vee \ell_z)$, introduce the eighteen symbols

$$t_c, \hat{t}_c, \tilde{t}_c, \quad x_c, \hat{x}_c, \bar{x}_c, y_c, \hat{y}_c, \bar{y}_c, z_c, \hat{z}_c, \bar{z}_c, \quad \overleftarrow{p}_c, p_c, \overrightarrow{p}_c, \quad \overleftarrow{o}_c, o_c, \overrightarrow{o}_c.$$

Let $\mathcal{S}_c$ denote this set of eighteen symbols. The intended six wires for c are the truth-marker wire, the three variable-occurrence wires, and two OR wires. Each of these wires contains three symbols. The middle symbol fixes the local role of the wire, while the two end symbols are the parts whose position relative to the truth marker encodes a Boolean value or the input-output behavior of an OR gate.

4.1 The gadgets

The truth-marker chain is

$$G_{\text{TM}_c} = \hat{t}_c t_c + \tilde{t}_c \tilde{t}_c.$$

Global truth-marker synchronization is enforced by

$$G_{\text{TM}} = b \sum_{\{c,d\} \in \binom{C}{2}} (t_c t_d + \hat{t}_c \hat{t}_d + \tilde{t}_c \tilde{t}_d).$$

7:8 On the Complexity of the Circuit Width Problem

For every literal occurrence of a Boolean variable x in clause c , the local variable gadget is

$$G_{x_c} = b\hat{t}_c(x_c + \hat{x}_c + \bar{x}_c) + \hat{x}_c(x_c + \bar{x}_c).$$

For each Boolean variable x , all occurrences are synchronized by

$$G_x = b \sum_{\{(\ell_1, c_1), (\ell_2, c_2)\} \in \binom{\text{Occ}(x)}{2}} (x_{c_1}x_{c_2} + \hat{x}_{c_1}\hat{x}_{c_2} + \bar{x}_{c_1}\bar{x}_{c_2}),$$

where $\text{Occ}(x)$ is the multiset of literal occurrences whose underlying Boolean variable is x , and the sum ranges over unordered pairs of distinct occurrences.

The binary OR gadget has input symbols u, v and output symbol q , with two endpoint decorations $\overleftarrow{q}, \overrightarrow{q}$. It is

$$G_{q=u \vee v} = b\overleftarrow{q}u + bqu + bq v + b\overrightarrow{q}v + \overleftarrow{q}q + \overrightarrow{q}q.$$

For a clause $c = (\ell_x \vee \ell_y \vee \ell_z)$, we set

$$G_c = G_{p_c = \ell_x \vee \ell_y} + G_{o_c = p_c \vee \ell_z} + b\hat{t}_c o_c,$$

where p_c is the intermediate output for $\ell_x \vee \ell_y$, and o_c is the output Boolean value of the whole clause. A positive literal x is represented by x_c , and a negative literal $\bar{x}$ by $\bar{x}_c$. The full polynomial is

$$f_\Phi = G_{\text{TM}} + \sum_x G_x + \sum_{c \in C} (G_{\text{TM}_c} + G_{x_c} + G_{y_c} + G_{z_c} + G_c).$$

The construction uses $1 + 18m$ symbols and has size polynomial in $|\Phi|$. If a clause contains repeated literals, distinct literal occurrences should be read as distinct algebraic symbols; we suppress the corresponding occurrence indices to keep the notation light. Only the local clause gadgets are needed to understand completeness. The global terms G_{TM} and G_x are used in soundness: they prevent a realization from choosing different coordinate systems for different clauses, and they force all appearances of a Boolean variable to agree on which endpoint represents the true literal.

4.2 Gadget behavior

► **Lemma 4.1** (Base and clause isolation). *In every realization of f_Φ , the base symbol b lies on a wire with no internal Hadamard gate. Moreover, no wire contains symbols from two distinct sets $\mathcal{S}_c$ and $\mathcal{S}_d$, $c \neq d$. Thus any realization on at most $1 + 6m$ wires must use one isolated base wire and exactly six local wires for each clause.*

► **Lemma 4.2** (Wire optimality). *If f_Φ has a realization on at most $k = 1 + 6m$ wires, then the realization uses exactly k wires, has exactly $12m$ internal Hadamard gates, and every quadratic monomial of f_Φ is realized by a Hadamard adjacency rather than by a CZ gate.*

► **Lemma 4.3** (Good clause structure). *In any realization of f_Φ on at most $1 + 6m$ wires, for every clause c , the eighteen symbols $\mathcal{S}_c$ occupy exactly six wires. Up to reversal of individual three-symbol chains, these wires are precisely*

1. $t_c, \hat{t}_c, \bar{t}_c$ with $\hat{t}_c$ in the middle;
2. $x_c, \hat{x}_c, \bar{x}_c$, with $\hat{x}_c$ between x_c and $\bar{x}_c$, and similarly for y, z ;
3. each OR triple $\overleftarrow{q}, q, \overrightarrow{q}$ lies on one wire, with $q \in \{p_c, o_c\}$ in the middle.

The global truth-marker terms force all truth-marker wires to have the same orientation, up to reversing the entire time axis.

► **Definition 4.4** (Encoding). For a clause $c = (\ell_x \vee \ell_y \vee \ell_z)$ satisfying Lemma 4.3, a symbol

$$s \in \{x_c, \bar{x}_c, y_c, \bar{y}_c, z_c, \bar{z}_c, p_c, o_c\}$$

encodes TRUE if s overlaps the truth marker t_c , and encodes FALSE otherwise. The literal ℓ_x is represented by x_c when $\ell_x = x$ and by $\bar{x}_c$ when $\ell_x = \bar{x}$, and similarly for y, z .

► **Lemma 4.5** (Encoding well-definedness). In every realization satisfying Lemma 4.3, for each variable occurrence in a clause c , exactly one of x_c and $\bar{x}_c$ overlaps t_c . The same statement holds for the y - and z -occurrence wires.

► **Lemma 4.6** (Variable-value realizability). For every Boolean variable x :

- (a) In any realization on at most $1 + 6m$ wires, all occurrences whose underlying variable is x induce the same value: x_c overlaps t_c for every such occurrence c if and only if this common value is TRUE.
- (b) Given any truth assignment, the truth-marker and variable-occurrence wires can be placed so that all marker-occurrence and synchronization cubic monomials are realized and each occurrence encodes the assigned value.

► **Lemma 4.7** (OR correctness). For a binary OR gadget $G_{q=u \vee v}$ in a good clause c :

- (a) In any realization on at most $1 + 6m$ wires, if q overlaps t_c , then at least one of u, v overlaps t_c .
- (b) Fix the input wires carrying u and v . There is a placement of the $\overleftarrow{q}, q, \overrightarrow{q}$ wire that realizes exactly the quadratic and cubic monomials in $G_{q=u \vee v}$; moreover, if at least one of u, v overlaps t_c , the placement can be chosen with q overlapping t_c .

► **Lemma 4.8** (Clause correctness). For any good clause $c = (\ell_x \vee \ell_y \vee \ell_z)$:

- (a) If o_c overlaps t_c , then at least one of the three literal symbols representing ℓ_x, ℓ_y, ℓ_z overlaps t_c .
- (b) Given fixed literal-occurrence wires, if at least one represented literal overlaps t_c , then the p_c - and o_c -wires can be placed so that all monomials in G_c are realized and o_c overlaps t_c .

Proof sketches for Lemmas 4.1–4.3 and 4.5–4.8 are deferred to Appendix A.1.

► **Theorem 4.9** (Completeness and soundness). For the polynomial f_Φ constructed above,

$$\Phi \text{ is satisfiable} \iff w(f_\Phi) \leq 1 + 6m.$$

Proof sketch. Completeness follows by placing the variable-occurrence wires according to a satisfying assignment and then applying Lemmas 4.6 and 4.8. For soundness, Lemmas 4.1–4.3 force the intended six-wire structure in every clause; Lemmas 4.5 and 4.6 decode a global assignment; and Lemma 4.8 plus the pinning term $bt_c o_c$ show that each clause is satisfied. ◀

► **Corollary 4.10.** CIRCUITWIDTH is NP-complete.

Proof sketch. Membership in NP follows because a certificate consists of a partition of symbols into at most k ordered wires and a specification of which quadratic monomials are Hadamard adjacencies; the interval constraints can be checked by a topological-sort test as in Lemma 6.1. Hardness is Theorem 4.9. ◀

► **Theorem 4.11.** Unless ETH fails, CIRCUITWIDTH has no $2^{o(n)}$ -time algorithm, where n is the number of symbols, even on instances with $k = \Theta(n)$.

Proof sketch. By the sparsification lemma [17], it is enough to consider 3-SAT formulas with N variables and $m = O(N)$ clauses. The reduction has $1 + 18m$ symbols and $k = 1 + 6m = \Theta(n)$. A $2^{o(n)}$ -time algorithm for CIRCUITWIDTH would therefore solve these 3-SAT instances in $2^{o(m)}$ time, contradicting ETH. ◀

5 Approximation hardness and degree-two inputs

5.1 Gap amplification from MAX-3-SAT

► **Definition 5.1** (Approximation version). For $\alpha > 1$, CIRCUITWIDTH- α -APPROX asks us to distinguish instances with $w(f) \leq k$ from instances with $w(f) \geq \alpha k$, promised that one case holds.

► **Theorem 5.2** (Max-3-SAT reduction). Let Φ be a 3-CNF formula with m clauses, and let $\text{OPT}(\Phi)$ be the maximum number of satisfiable clauses. The construction above can be realized with $1 + 6m + (m - \text{OPT}(\Phi))$ wires, and no realization can use fewer wires.

Proof sketch. The extra-wire accounting is deferred to Appendix A.1. Each unsatisfied clause costs one additional wire, and each additional wire can repair at most one clause obstruction. ◀

► **Theorem 5.3.** For every constant $\alpha < 49/48$, CIRCUITWIDTH- α -APPROX is NP-hard.

Proof sketch. By Håstad's theorem [14], it is NP-hard to distinguish satisfiable formulas from formulas for which at least $(1/8 - o(1))m$ clauses remain unsatisfied. The corresponding widths are at most $1 + 6m$ in the first case and at least $1 + 6m + (1/8 - o(1))m$ in the second. The ratio tends to $(6 + 1/8)/6 = 49/48$. ◀

5.2 Removing cubic terms

We now describe the degree-two transformation as an explicit recipe. The input is the cubic polynomial f_Φ from Section 4; every cubic monomial in that polynomial has the form buv , with the global base symbol b and two non-base symbols u, v . Fix a constant $N > 60$ and a sufficiently large polynomially bounded integer M . The degree-two polynomial f'_Φ is built as follows.

- (i) Replace b by a base family $B = \{b^1, \dots, b^{N^m}\}$, and replace every other symbol u by a twin family $U = \{u^1, \dots, u^M\}$.
- (ii) Add all quadratic terms inside each family. These clique constraints are not meant to simulate gates of the original circuit; they make it expensive for many copies of the same symbol to be used as exceptional connectors.
- (iii) For each original quadratic monomial uv , add only the matching terms $u^i v^i$, $1 \leq i \leq M$. Thus the i -th non-base copies reproduce the original Hadamard-adjacency structure independently for each index i .
- (iv) For each original cubic monomial buv , add the three complete bipartite graphs between the corresponding families:

$$\sum_{r,i} b^r u^i + \sum_{r,j} b^r v^j + \sum_{i,j} u^i v^j.$$

This is the only place where a cubic constraint is removed: the required common intersection is replaced by all three pairwise-overlap requirements among copies of b, u, v .

The yes-case realization uses M parallel non-base copies of the original six-wires-per-clause construction, together with the remaining base copies on separate base wires, for baseline width $(6M + N)m$.

The clique and bipartite terms make the replacement robust: near-optimal realizations have many surviving copies, and every surviving copy triple for a replaced term buv satisfies the original common-intersection condition by Helly's theorem. The accounting is deferred to Appendix A.2.

► **Lemma 5.4** (Twin survival). *In any near-optimal realization of $f^{(2)}$, all but a bounded number of copies in each family are surviving: they are not used as flexible connectors for symbols outside their intended family interactions. For surviving copies, the quadratic replacement of a cubic term buv enforces a common intersection in the same sense as the original cubic term.*

Proof sketch. See Appendix A.2. ◀

► **Theorem 5.5.** *For every constant $\alpha < 49/48$, CIRCUITWIDTH- α -APPROX remains NP-hard when the input polynomial is restricted to degree at most two.*

Proof sketch. Apply the twin transformation to the gap instances of Theorem 5.2. The surviving copies simulate the original cubic constraints by Lemma 5.4, so a near-optimal degree-two realization induces a near-optimal realization of the original cubic instance, up to the predetermined lower-order slack. Taking M sufficiently large makes this slack negligible compared with the $m/8$ gap from Håstad's theorem [14], preserving every approximation factor below $49/48$. ◀

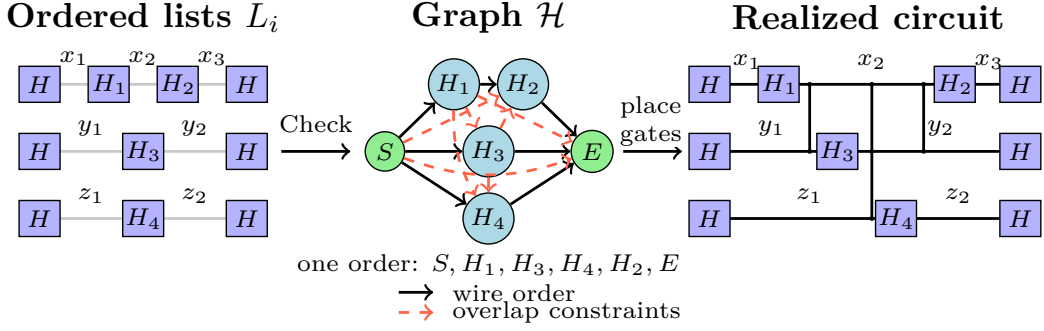
6 Algorithms

We now turn to the algorithmic side of the width problem. The guiding point is that, once the symbols have been assigned to ordered wire lists, the remaining question is purely one-dimensional: can the active intervals of the symbols be scheduled so that all monomials of the polynomial are realized by the permitted H , Z , CZ , and CCZ gates? This scheduling problem is exactly an acyclicity test for a directed constraint graph. Thus the algorithmic task splits into two parts. First, find or certify a plausible partition of the symbols into k ordered lists. Second, run a topological-sort check on the endpoint constraints induced by these lists.

Let G_f denote the *connection graph* of f . Its vertex set is the symbol set $\mathcal{S}$, and two symbols are adjacent whenever they occur together in a quadratic monomial or as part of a cubic monomial. We use the following harmless cleanup throughout this section. If a quadratic term uv appears while a cubic term uvw also appears for some w , then temporarily delete the quadratic copy. Any realization of uvw already places u, v, w on three distinct wires with a common overlap, and hence the deleted term can be restored at the end by placing a CZ gate inside that overlap. After this cleanup, a quadratic term that is used as a direct adjacency on a wire is never simultaneously forced as a subterm of a CCZ constraint.

6.1 Checking fixed ordered lists

Suppose first that we are given ordered lists $L_1, \dots, L_k$, one for each wire. The lists include only the real symbols; conceptually, each list is also bracketed by a local source and a local sink, which are identified with global endpoint vertices S and E . Consecutive real symbols on a list are the only pairs that may be separated by an internal H gate. Therefore every



■ **Figure 3** The CHECK routine for fixed ordered lists. Left: the proposed wire lists. Middle: the endpoint digraph $\mathcal{H}$, where black edges encode wire order and dashed red edges encode overlap constraints from monomials. Right: any topological order of $\mathcal{H}$ gives gate positions and hence a realizing circuit with H , CZ , and CCZ gates placed consistently.

consecutive real pair (u, v) must correspond to a quadratic monomial uv of f . Quadratic monomials not used in this direct way are interpreted as CZ -overlap constraints, and cubic monomials are interpreted as CCZ -overlap constraints.

For fixed lists we build a directed graph $\mathcal{H}$. Its vertices are endpoint or link vertices, representing the positions of H gates. For a symbol u , write $L(u)$ and $R(u)$ for the link immediately before and immediately after u on its list. If u and v are consecutive on a wire, then $R(u) = L(v)$. The wire order gives the directed path from S through the links on each list to E . A remaining quadratic monomial uv adds the two precedence constraints

$$L(u) \rightarrow R(v), \quad L(v) \rightarrow R(u),$$

which are exactly the inequalities saying that the intervals of u and v overlap. A cubic monomial uvw adds the same two constraints for each of the three pairs. The construction is illustrated in Figure 3.

► **Lemma 6.1** (Structural characterization). *A polynomial f has a width- k realization with prescribed ordered lists $L_1, \dots, L_k$ if and only if every adjacent real-symbol pair in the lists is a quadratic monomial of f , and the directed graph $\mathcal{H}$ constructed above is acyclic. Given the lists, the check and the corresponding circuit construction take $O(m + n)$ time.*

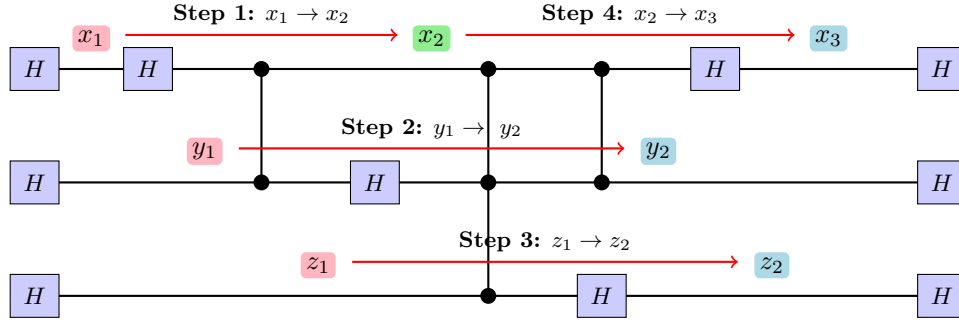
Proof sketch. A realizing circuit gives real times to all H gates, so every wire-order edge and every monomial-induced precedence edge in $\mathcal{H}$ points forward in time. Thus $\mathcal{H}$ is acyclic. Conversely, suppose $\mathcal{H}$ is acyclic and choose a topological order. This order assigns times to all endpoints, so each symbol u occupies the interval $(L(u), R(u))$. The two directed inequalities for a quadratic monomial uv are equivalent to the intersection of these two intervals, allowing a CZ gate unless uv was already used as a direct H -adjacency. For a cubic monomial uvw , the six directed inequalities force the three intervals to be pairwise intersecting; by Helly's theorem for intervals (Theorem 3.2), they have a common intersection, where we place a CCZ gate. Linear terms are realized by Z gates inside their symbol intervals. The graph has $O(n)$ vertices and $O(m + n)$ edges, so Kahn's topological-sort algorithm [19] gives the stated running time and also constructs the schedule. ◀

We denote this linear-time subroutine by $\text{CHECK}(f, L_1, \dots, L_k)$. The rest of the section explains how to find suitable ordered lists. The nondeterministic algorithm guesses only the endpoints of the lists and reconstructs the interior greedily; the deterministic algorithm stores the same information locally in a tree-decomposition dynamic program.

6.2 Endpoint witnesses and greedy reconstruction

The witness only names the two endpoints of every wire. Namely, it gives a set S_{head} of the k leftmost symbols and a set S_{tail} of the k rightmost symbols, using $2 \log_2 \binom{n}{k} = O(k \log(en/k))$ nondeterministic bits. The rest of the wire order is then forced greedily: start one partial list at each head, keep **CurrentHeads** as the current right endpoints of unfinished lists, and let U be the symbols not yet placed.

In each round, first close any list whose current endpoint already lies in S_{tail} . Then look for an open endpoint v that has exactly one neighbor u in U in the connection graph G_f . Such a neighbor is the only possible next symbol on that wire, so append u , replace v by u in **CurrentHeads**, and remove u from U . If unplaced symbols remain but no endpoint has a unique unplaced neighbor, the guessed endpoints are rejected. After all symbols are placed, **CHECK** verifies the resulting ordered lists. Figure 4 illustrates this greedy reconstruction on the example from Figure 3.



■ **Figure 4** Greedy reconstruction from endpoint sets. For $f_0 = x_1x_2 + x_2x_3 + y_1y_2 + z_1z_2 + x_2y_1 + x_2y_2 + x_2y_2z_1$, the witness says that the three wires start at x_1, y_1, z_1 and end at x_3, y_2, z_2 . The algorithm repeatedly extends a current endpoint by its unique unplaced neighbor in G_f , producing $L_1 = (x_1, x_2, x_3)$, $L_2 = (y_1, y_2)$, and $L_3 = (z_1, z_2)$, and then calls **CHECK** to verify the lists.

The unique-neighbor property is the reason the endpoint witness is enough.

► **Lemma 6.2** (Unique neighbor and greedy extension). *If S_{head} and S_{tail} are exactly the symbols at the start and end of each wire in some valid width- k circuit realizing f , then after the procedure closes the current heads in S_{tail} , at every nonterminal step there exists an open current head $v \in \text{CurrentHeads}$ with exactly one neighbor among the untouched symbols in G_f . This neighbor is the true next symbol on the same wire. Consequently, the greedy procedure with the correct witness reconstructs ordered lists compatible with a valid realization.*

Proof sketch. Maintain the invariant that each partial list built by the algorithm is a prefix of a true wire list. The closing step is valid because a true tail has no successor on its own wire. After the closing step, every current head is an open head. Choose one whose active interval ends earliest in the realizing circuit. Its true successor is an untouched neighbor because an internal H gate between consecutive symbols contributes the corresponding quadratic monomial. No untouched symbol on another wire can also be adjacent to the earliest-ending head: such an adjacency would require overlap, but every untouched symbol starts to the right of the current frontier on its own wire, whose endpoint is no later than the start of that symbol. Thus the only untouched neighbor is the true successor. Induction proves the lemma. ◀

► **Theorem 6.3** (Endpoint witness). *For the search version of $\text{CIRCUITWIDTH}(f, k)$, there is a nondeterministic polynomial-time algorithm using $2 \log_2 \binom{n}{k} = O(k \log(en/k))$ non-deterministic bits. Enumerating all endpoint witnesses gives a deterministic $\binom{n}{k}^2 \text{poly}(n)$ -time algorithm.*

Proof sketch. The witness is the pair $(S_{\text{head}}, S_{\text{tail}})$. With the correct pair, Lemma 6.2 reconstructs compatible lists, and Lemma 6.1 then constructs a realizing circuit. With an incorrect pair, the algorithm accepts only if CHECK verifies an actual width- k realization, so every accepting branch is sound. Deterministically trying all two endpoint sets gives the claimed running time. ◀

6.3 From greedy reconstruction to bounded pathwidth

The same greedy view also gives the structural observation used by the fixed-parameter algorithm: if the circuit exists, then G_f has a path decomposition with bags of size at most $k + 1$. Intuitively, the moving set of current heads is a separator between the symbols already reconstructed and the symbols not yet touched.

► **Lemma 6.4** (Path decomposition from a realization). *If $w(f) \leq k$, then $\text{pw}(G_f) \leq k$, and hence $\text{tw}(G_f) \leq k$.*

Proof sketch. Run the greedy procedure using the true endpoint sets. Keep an auxiliary set S_t of size k , retaining the tail symbol of a wire even after that wire is closed. Let S_0 be the head set. Whenever the greedy process replaces a head v by its successor u , set $S_t = (S_{t-1} \setminus \{v\}) \cup \{u\}$. Use bags $B_0 = S_0$ and $B_t = S_{t-1} \cup S_t$. Each bag has size at most $k + 1$, so the width is at most k . Each symbol appears in a contiguous subsequence of bags. A direct H -adjacency is covered by the transition bag where the successor is introduced. An overlap edge (coming from CZ or CCZ gates) cannot jump over the frontier; otherwise an earliest-ending current head would have an untouched neighbor other than its true successor, contradicting Lemma 6.2. Therefore every edge of G_f is covered, and the bags form a path decomposition. ◀

Given the polynomial, we first build G_f in $O(m)$ time. We then run, for example, Korhonen's single-exponential 2-approximation algorithm for treewidth [20]. If it certifies that $\text{tw}(G_f) > k$, then we reject by Lemma 6.4. Otherwise we obtain a tree decomposition of width $O(k)$ and convert it into a standard nice tree decomposition with leaf, introduce, forget, and join nodes, as recalled in the full version [18].

6.4 Dynamic programming on a tree decomposition

We now describe the explicit dynamic program. Let x be a node of the nice tree decomposition, let B_x be its tree bag, and let V_x be the set of symbols appearing in the subtree rooted at x . The table $\mathcal{D}_x$ stores *realizable states*. Each state encodes a partial wire assignment and a boundary topological order for the symbols in B_x . A state is realizable if it can be extended to a valid circuit configuration using all symbols in V_x .

Intuitively, a valid circuit configuration for V_x consists of:

- **wire assignments:** a partition of symbols in V_x into k ordered lists $L_1, \dots, L_k$;
- **edge types:** an assignment ρ describing, for each pair of consecutive symbols on the same list, whether they are adjacent by a direct internal H gate or are not connected inside the current partial solution;

such that the induced partial endpoint graph $\mathcal{H}[V_x(\rho)]$ is acyclic. This captures exactly the structure needed to realize the symbols in V_x : the lists determine the wire assignment, the edge types encode local connectivity, and acyclicity of $\mathcal{H}[V_x(\rho)]$ ensures that the relevant H gates can be consistently ordered in time. At the root, where $V_x = \mathcal{S}$, such a configuration is precisely a complete circuit layout.

A state in $\mathcal{D}_x$ has two components $(\mathcal{W}_x, \mathcal{O}_x)$:

1. $\mathcal{W}_x = (W_1, \dots, W_k, \tau)$. This is a partition of the boundary symbols in B_x into k ordered lists W_i , each with endpoint sentinels s, e , together with an edge type τ on consecutive entries of each W_i :
 - $\tau(u, v) = 1$: real symbols u, v are adjacent on the wire, so they are connected by a direct internal H gate;
 - $\tau(u, v) = 2$: the two visible entries are connected through forgotten real symbols in $V_x \setminus B_x$, and each consecutive real-symbol pair on that hidden path is connected by a direct internal H gate;
 - $\tau(u, v) = 0$: the two visible entries are not connected inside the current subtree.
2. $\mathcal{O}_x$. This is a topological ordering of the boundary endpoint vertices

$$\{S, E\} \cup \{L(u), R(u) : u \in B_x\},$$

after the identifications induced by type-1 pairs, with S first and E last.

The formal realizability condition is the following. A state $(\mathcal{W}_x, \mathcal{O}_x)$ is stored in $\mathcal{D}_x$ if and only if there exist ordered lists $L_1, \dots, L_k$ partitioning V_x and edge types $\rho \in \{0, 1\}$ on consecutive entries in each L_i such that:

- (i) each boundary list W_i is a subsequence of L_i with order preserved;
- (ii) τ extends to ρ , meaning that type-0 and type-1 boundary claims remain unchanged, while a type-2 boundary relation is realized by a nonempty path of type-1 direct H -adjacencies through forgotten real symbols;
- (iii) the partial constraint graph $\mathcal{H}[V_x(\rho)]$ is acyclic, and $\mathcal{O}_x$ is the restriction of one of its topological orderings to the boundary vertices $\{S, E\} \cup \{L(u), R(u) : u \in B_x\}$.

Following the standard bottom-up dynamic-programming paradigm on nice tree decompositions [7], the four nice-node transitions are local. At a leaf node, the unique state has empty boundary lists and the order $S < E$. At an introduce node, a new symbol is inserted into one visible type-0 gap of one boundary wire; the transition chooses the two new boundary types, inserts $L(u), R(u)$ into the boundary order, and checks all monomial constraints that have become visible. At a forget node, the symbol is removed from the boundary list after all constraints that would become internal have been checked; the two exposed boundary gaps are merged, recording type 2 if a hidden direct- H path has been completed. At a join node, two child states are compatible only if they agree on the visible wire order and on the boundary topological order; the type information is then merged. The common boundary order certifies acyclicity of the union, because any directed cycle would either already lie in one child or would force two boundary vertices to appear in opposite orders. For details, see the full version [18].

At the root r , the bag is empty. The algorithm accepts exactly when $\mathcal{D}_r$ is nonempty. In that case, backpointers recover the ordered lists and the direct- H adjacencies, and a final call to CHECK from Lemma 6.1 constructs the actual circuit. If $\mathcal{D}_r$ is empty, then no valid width- k circuit exists.

► **Theorem 6.5** (Fixed-parameter algorithm). *Given a degree-three polynomial f on n symbols with m monomials and an integer k , the search version of $\text{CIRCUITWIDTH}(f, k)$ can be solved in $k^{6k+o(k)}n + O(m)$ time.*

Proof sketch. By Lemma 6.4, the treewidth preprocessing is sound: if the connection graph has treewidth greater than k , no width- k realization exists; otherwise we obtain a nice decomposition of width $O(k)$. For each bag, the number of possible ordered boundary wire structures, three-valued type assignments, and boundary endpoint orders is $k^{6k+o(k)}$. The introduce, forget, and join transitions are bounded by the same factor, and the number of nice bags is linear in n . The DP invariant above proves by induction that a state is stored exactly when it has a compatible extension over the processed subtree. Hence a nonempty root table yields a circuit, recovered by backpointers and Lemma 6.1, while an empty root table certifies that none exists. Building G_f and performing the cleanup take $O(m)$ time, so the total running time is $k^{6k+o(k)}n + O(m)$. ◀

7 Conclusion

We proved that the width of the polynomial representation of IQP-type circuits is computationally hard to determine and hard to approximate below $49/48$, with the same approximation hardness persisting for degree-two inputs. At the same time, the problem has a strong structural side: width- k realizations force low pathwidth in the connection graph, which leads to compact nondeterministic witnesses and to a deterministic FPT algorithm. The main open directions are to close the approximability gap, improve the dependence on k in the dynamic program, and identify important circuit families for which width can be optimized efficiently.

References

- 1 Scott Aaronson. A linear-optical proof that the permanent is #p-hard. *arXiv preprint*, 2011. [arXiv:1109.1674](https://arxiv.org/abs/1109.1674).
- 2 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, pages 333–342, New York, NY, USA, 2011. Association for Computing Machinery. doi:10.1145/1993636.1993682.
- 3 Leonard M. Adleman, Jonathan DeMarrais, and Ming-Deh A. Huang. Quantum computability. *SIAM Journal on Computing*, 26(5):1524–1540, 1997. doi:10.1137/S0097539795293639.
- 4 Hans L Bodlaender. A tourist guide through treewidth. *Acta cybernetica*, 11(1-2):1–21, 1993. URL: <https://cyber.bibl.u-szeged.hu/index.php/actcybern/article/view/3417>.
- 5 Michael J. Bremner, Richard Jozsa, and Dan J. Shepherd. Classical simulation of commuting quantum computations implies collapse of the polynomial hierarchy. *Proc. R. Soc. A*, 467(2126):459–472, 2011.
- 6 Bin Cheng, Ziyuan Wang, Ruixuan Deng, Jianxin Chen, and Zhengfeng Ji. Breaking the treewidth barrier in quantum circuit simulation with decision diagrams. *arXiv:2510.06775*, 2025.
- 7 Marek Cygan, Fedor V. Fomin, Lukasz Kowalik, Daniel Lokshtanov, Dániel Marx, Marcin Pilipczuk, Michal Pilipczuk, and Saket Saurabh. *Parameterized Algorithms*. Springer, 2015. doi:10.1007/978-3-319-21275-3.
- 8 Alexander M. Dalzell, Aram W. Harrow, Dax Enshan Koh, and Rolando L. La Placa. How many qubits are needed for quantum computational supremacy? *Quantum*, 4:264, May 2020. doi:10.22331/q-2020-05-11-264.

- 9 Christopher M. Dawson, Andrew P. Hines, Duncan Mortimer, Henry L. Haselgrove, Michael A. Nielsen, and Tobias J. Osborne. Quantum computing and polynomial equations over the finite field $\mathbb{Z}_2$. *Quantum Info. Comput.*, 5(2):102–112, March 2005. doi:10.26421/QIC5.2-2.
- 10 Rodney G Downey and Michael R Fellows. *Fundamentals of Parameterized Complexity*, volume 4. Springer, 2013. doi:10.1007/978-1-4471-5559-1.
- 11 R. P. Feynman. Simulating physics with computers. *International Journal of Theoretical Physics*, 21(6–7):467–488, 1982. doi:10.1007/BF02650179.
- 12 Lance Fortnow and John Rogers. Complexity limitations on quantum computation. *Journal of Computer and System Sciences*, 59(2):240–252, 1999. doi:10.1006/jcss.1999.1651.
- 13 Keisuke Fujii and Tomoyuki Morimae. Commuting quantum circuits and complexity of ising partition functions. *New Journal of Physics*, 19(3):033003, March 2017. doi:10.1088/1367-2630/aa5fdb.
- 14 Johan Håstad. Some optimal inapproximability results. *J. ACM*, 48(4):798–859, July 2001. doi:10.1145/502090.502098.
- 15 Ed. Helly. Über mengen konvexer körper mit gemeinschaftlichen punkte. *Jahresbericht der Deutschen Mathematiker-Vereinigung*, 32:175–176, 1923. URL: <http://eudml.org/doc/145659>.
- 16 Russell Impagliazzo and Ramamohan Paturi. On the complexity of k-sat. *Journal of Computer and System Sciences*, 62(2):367–375, 2001. doi:10.1006/jcss.2000.1727.
- 17 Russell Impagliazzo, Ramamohan Paturi, and Francis Zane. Which problems have strongly exponential complexity? *Journal of Computer and System Sciences*, 63(4):512–530, 2001. doi:10.1006/jcss.2001.1774.
- 18 Zhengfeng Ji, Yincheng Liu, and Zhe’ou Zhou. On the complexity of the circuit width problem, 2026. arXiv:2606.18201.
- 19 A. B. Kahn. Topological sorting of large networks. *Communications of the ACM*, 5(11):558–562, 1962. doi:10.1145/368996.369025.
- 20 Tuukka Korhonen. A single-exponential time 2-approximation algorithm for treewidth. In *2021 IEEE 62nd Annual Symposium on Foundations of Computer Science (FOCS)*, pages 184–192. IEEE, 2021. doi:10.1109/FOCS52979.2021.00026.
- 21 Greg Kuperberg. How hard is it to approximate the jones polynomial? *Theory of Computing*, 11(6):183–219, 2015. doi:10.4086/toc.2015.v011a006.
- 22 Igor L. Markov and Yaoyun Shi. Simulating quantum computation by contracting tensor networks. *SIAM Journal on Computing*, 38(3):963–981, 2008. doi:10.1137/050644756.
- 23 Ashley Montanaro. Quantum circuits and low-degree polynomials over $\mathbb{F}_2$. *Journal of Physics A: Mathematical and Theoretical*, 50(8):084002, January 2017. doi:10.1088/1751-8121/aa565f.
- 24 Neil Robertson and Paul D Seymour. Graph minors. i. excluding a forest. *Journal of Combinatorial Theory, Series B*, 35(1):39–61, 1983. doi:10.1016/0095-8956(83)90079-5.
- 25 Peter W. Shor. Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. *SIAM Journal on Computing*, 26(5):1484–1509, 1997. doi:10.1137/S0097539795293172.
- 26 Ziyuan Wang, Bin Cheng, Longxiang Yuan, and Zhengfeng Ji. FeynmanDD: Quantum Circuit Analysis with Classical Decision Diagrams. In *Proceedings of the 37th International Conference on Computer Aided Verification*, Zagreb, Croatia, July 2025.
- 27 Andrew Chi-Chih Yao. Quantum circuit complexity. In *Proceedings of 1993 IEEE 34th Annual Foundations of Computer Science*, pages 352–361, 1993. doi:10.1109/SFCS.1993.366852.

A

 Technical details deferred from the main text

A.1 Deferred SAT-gadget checks

Base isolation

The symbol b appears only in cubic monomials. If an internal Hadamard gate were adjacent to b , it would contribute a quadratic monomial bs . Since the two adjacent supports are disjoint, the same monomial cannot also be produced by a CZ gate and canceled over $\mathbb{F}_2$. This contradicts the definition of f_Φ . The same argument applies to a Hadamard transition between symbols from two different clauses, because no cross-clause quadratic monomial is present in f_Φ . Hence clause symbol sets cannot share wires, which proves Lemma 4.1.

Wire optimality and good clauses

The construction has $1 + 18m$ symbols. A circuit with w wires and h internal Hadamard gates has $w + h$ symbols, so $w \leq 1 + 6m$ implies $h \geq 12m$. On the other hand, f_Φ has exactly $12m$ quadratic monomials: two for each of the six local chains in every clause. Each internal Hadamard gate contributes one quadratic monomial, and an adjacent pair of symbols has disjoint supports, so this monomial cannot be canceled by a CZ gate. Hence $h \leq 12m$, and equality proves Lemma 4.2.

Together with Lemma 4.1, this forces the two quadratic monomials in each intended chain to be Hadamard adjacencies. A wire containing four local symbols would create a length-three Hadamard chain and hence a quadratic monomial not present in f_Φ . Thus each clause needs at least six local wires, and the global budget leaves exactly six. The listed quadratic pairs then uniquely force the six three-symbol chains. The terms in G_{TM} align the roles $t, \hat{t}, \tilde{t}$ across clauses, up to one global time reversal, proving Lemma 4.3.

Encoding and synchronization

The marker–occurrence cubic monomials

$$b\hat{t}_c x_c, \quad b\hat{t}_c \hat{x}_c, \quad b\hat{t}_c \bar{x}_c$$

force $\hat{t}_c$ to overlap all three symbols $x_c, \hat{x}_c, \bar{x}_c$. In the good-clause structure, x_c and $\bar{x}_c$ lie on opposite sides of $\hat{x}_c$, while $\hat{t}_c$ lies between t_c and $\tilde{t}_c$. Hence exactly one of $x_c, \bar{x}_c$ overlaps t_c , proving Lemma 4.5.

The truth-marker synchronization terms force every pair of truth-marker wires to have the same role order, up to the same global reversal. For two occurrences of x , the terms

$$bx_{c_1}x_{c_2}, \quad b\hat{x}_{c_1}\hat{x}_{c_2}, \quad b\bar{x}_{c_1}\bar{x}_{c_2}$$

force the two occurrence wires to have the same orientation relative to their truth-marker wires. Thus the overlap pattern of x_c and $\bar{x}_c$ with t_c is independent of the clause. Conversely, given an assignment, choose the common truth-marker orientation and orient each occurrence wire according to the assigned value. The same overlaps realize the required marker–occurrence and synchronization terms, proving Lemma 4.6.

OR and clauses

In $G_{q=u \vee v}$, the quadratic terms $\overleftarrow{q}q$ and $\overrightarrow{q}q$ place q between its two decorated endpoints. The cubic terms involving u force the junction between $\overleftarrow{q}$ and q to lie in $\text{supp}(u)$, and the terms involving v force the junction between q and $\overrightarrow{q}$ to lie in $\text{supp}(v)$. Thus $\text{supp}(q)$ is

contained in the interval spanned by $\text{supp}(u) \cup \text{supp}(v)$. If neither input overlaps t_c , this interval cannot cross t_c , so q cannot overlap t_c . Conversely, if one input meets t_c , put the corresponding junction point in that intersection and the other junction point in the other input support. This realizes the four cubic terms and makes q overlap t_c , proving Lemma 4.7.

The clause gadget is the cascade $p_c = \ell_x \vee \ell_y$ and $o_c = p_c \vee \ell_z$. Applying Lemma 4.7 to the two OR gadgets gives both directions of Lemma 4.8; the final monomial $bt_c o_c$ is realizable exactly when o_c overlaps t_c , since b is active for the whole interval.

Gap accounting

Let $\Delta = m - \text{OPT}(\Phi)$. An assignment satisfying $m - \Delta$ clauses is realized by the intended six wires on every satisfied clause. For each unsatisfied clause, one extra wire realizes the final output constraint locally without making o_c overlap the truth marker, giving $1 + 6m + \Delta$ wires. Conversely, a realization using $1 + 6m + r$ wires has the width-optimal six-wire form on all but at most r clauses. On those good clauses, the OR constraints and $bt_c o_c$ imply satisfaction under the decoded assignment. Hence each extra wire can repair at most one clause obstruction, proving Theorem 5.2.

A.2 Degree-two twin replacement

The key point is that a clique of quadratic constraints leaves almost all copies geometrically usable. If three clique copies lie on one wire in order $a - b - c$, then ac cannot be realized; if two wires each contain two clique copies, the earlier internal H -adjacency makes a crossed pair disjoint. Thus at most two copies in such a clique can share a wire with another clique copy. For a replaced cubic term buv , the family cliques and the three complete bipartite graphs form a clique on the copies of b, u, v . Hence at least $Nm - 2$ base copies and $M - 2$ copies of each non-base symbol survive this term. Union-bounding over all cubic-term families gives at least $M - 20$ globally surviving copies of every non-base symbol and at least $Nm - 60m$ globally surviving copies of b : the worst non-base losses are 20 for truth markers, 8 for variable-occurrence symbols, and 10 for OR symbols, while the base losses are $6 + 18m + 6n + 16m + 2m \leq 60m$, using $n \leq 3m$. Fix a clause c . The sharper local count loses at most $26 + 48 + 22 = 96$ indices among its 18 local symbols, so at least $M - 96$ indices j make all copies s^j , $s \in \mathcal{S}_c$, globally surviving. Since $N > 60$, choose one globally surviving base copy b^r . For every original cubic term buv , the quadratic replacement forces b^r, u^j, v^j to pairwise intersect; by Theorem 3.2, they have a common intersection, exactly as the original CCZ constraint required. Finally, distinct surviving base copies and distinct surviving clause copies cannot share wires: a consecutive shared-wire pair would either have no matching quadratic term or contradict survival for a tripartite replacement. Thus each surviving clause copy uses six wires if it is good and at least seven otherwise. If g is the number of good surviving clause copies, then

$$w(f'_\Phi) \geq Nm - 60m + 7 \sum_c |J_c| - g.$$

A circuit with at most $(7M + N)m - tM - 750m$ wires would force $g > tM$, so more than t clauses have a good copy. The decoding lemmas from the cubic construction then give an assignment satisfying more than t clauses. This proves Lemma 5.4 and the degree-two gap.

A.3 Checking a fixed wire order

We also record the constructive content of Lemma 6.1. Suppose the symbols have already been partitioned into ordered lists $L_1, \dots, L_k$. Consecutive symbols on a list are the only candidates for internal Hadamard adjacencies. Therefore the first check is purely algebraic: if two consecutive genuine symbols u, v are declared adjacent, then the quadratic monomial uv must appear in f . Otherwise the wire order would create a quadratic term that is not in the target polynomial.

After this check, all remaining monomials are interpreted as overlap requirements. For every symbol u introduce two endpoint vertices $L(u), R(u)$ and the edge $L(u) \rightarrow R(u)$. If u, v are consecutive on a wire and are realized by an internal Hadamard adjacency, identify $R(u) = L(v)$. A remaining quadratic monomial uv requires the two intervals to intersect, and this is equivalent to the two precedence constraints $L(u) \rightarrow R(v)$ and $L(v) \rightarrow R(u)$. A cubic monomial uvw requires a common intersection of three intervals; for intervals, by Helly's theorem, this is equivalent to the three pairwise intersections, so we add the same two edges for each of the three pairs.

If the resulting directed graph has a cycle, then no assignment of real endpoint times can satisfy all precedence constraints. Conversely, if it is acyclic, choose any topological order and place endpoints according to that order, with a small perturbation to separate equalities except those intentionally identified by Hadamard adjacencies. The added edges guarantee that every required pair of intervals overlaps. For every cubic monomial, the three pairwise overlaps imply a common overlap, so a CCZ gate can be placed at a common time. Quadratic monomials not used as Hadamard adjacencies are realized by CZ gates at their overlap times, and linear monomials are realized by Z gates. This produces a circuit realizing exactly f .

The construction is linear in the input size once the monomials are stored as incidence lists. The endpoint graph has $O(n)$ vertices and $O(m + n)$ edges, because each monomial contributes only constantly many precedence edges. A single topological sort either returns a schedule or certifies impossibility. This is the subroutine used by both the nondeterministic reconstruction and the final reconstruction step of the FPT algorithm.

A.4 Greedy reconstruction from endpoints

We expand the proof of Lemma 6.2. Assume that the guessed heads and tails are the true sets of first and last symbols of the wires in some valid realization. During the greedy procedure, maintain the invariant that every partial list already constructed is a prefix of the corresponding true wire. Equivalently, the current heads form a cut through the realizing circuit: all assigned symbols lie weakly to the left of this cut on their wires, and all unassigned symbols lie to the right.

At the beginning of each iteration, the algorithm closes every current head that belongs to S_{tail} . This is safe for the reconstruction step: such a symbol has no successor on its own wire, while any remaining overlap edges incident to it will be verified later by the independent check. If unassigned symbols remain, then some unclosed wire still has an unassigned successor.

Choose, among the remaining current heads, a head v whose right endpoint is earliest in the true schedule, and let u be its true successor. The pair v, u is an internal Hadamard adjacency, hence an edge of the connection graph, so u is an unassigned neighbor of v . We claim that u is the only such neighbor. If there were another unassigned neighbor z , then some monomial would force the interval of z to overlap the interval of v . But z lies to the

right of the current cut on its own wire. Its wire therefore has a current head whose right endpoint must occur no later than the left endpoint of z . Since z overlaps v , this would force that current head to end before or at the same time as v , contradicting the choice of v as the earliest-ending available head unless z is exactly the successor on the same wire.

Thus the selected open head has exactly one unassigned neighbor, and the greedy rule is forced to append its true successor. The prefix invariant is preserved, and the number of unassigned symbols decreases after each extension. Induction proves that with the correct endpoint witness the greedy procedure never gets stuck and reconstructs wire lists compatible with the original realization.

For soundness, no trust is placed in the witness. The greedy procedure may construct lists from an incorrect pair of endpoint sets, but the algorithm accepts only after the independent check of Lemma 6.1 succeeds. Thus any accepting branch outputs an actual width- k realization. The witness size is exactly the information needed to specify the two endpoint sets, namely $2\log_2 \binom{n}{k}$ bits.

A.5 Dynamic-programming states and transitions

We use the notation from Section 6.4. For a bag B_x , a state $(\mathcal{W}_x, \mathcal{O}_x)$ has the following formal meaning. The wire part $\mathcal{W}_x$ places the symbols of B_x into k ordered lists, each with sentinels s, e . For every consecutive visible pair (u, v) it stores a type $\tau(u, v) \in \{0, 1, 2\}$. Type 1 means that u, v are already a direct internal-Hadamard pair, and therefore uv must be a quadratic monomial of f . Type 2 means that u and v are connected by a nonempty path of forgotten direct-Hadamard pairs. Type 0 means that the processed part has not connected them by a Hadamard chain.

The order part $\mathcal{O}_x$ is a topological order of the visible endpoint vertices $S, E, L(u), R(u)$ for $u \in B_x$, after the identifications $R(u) = L(v)$ forced by type-1 pairs. It must respect the interval edges $L(u) \rightarrow R(u)$, the visible wire-order edges, and every currently visible overlap edge induced by a monomial whose relevant symbols are in the processed subtree. A state is stored exactly when it has an extension to all symbols in V_x whose full partial endpoint graph is acyclic and restricts to $\mathcal{O}_x$ on the boundary.

For a leaf bag, the unique state consists of k empty lists (s, e) and the order $S < E$. At an introduce node for a symbol u , we insert u into one visible type-0 gap of one wire. Inserting into a type-1 or type-2 gap is forbidden: type 1 is already a direct Hadamard adjacency, and type 2 hides a completed forgotten path. The two new neighboring pairs are marked type 0 or legal type 1, and the order inserts $L(u), R(u)$ in all possible positions consistent with the interval, wire, and newly visible overlap edges.

At a forget node for u , we remove u from the visible wire list. If the visible neighbors of u are a and b , the new type of (a, b) records whether the removed portion has become a hidden Hadamard path: two incident pieces that contain a type-1 or type-2 segment are merged into type 2, while two empty type-0 pieces merge into type 0. Before deleting $L(u), R(u)$, the transition checks all monomial constraints involving u whose other symbols have already appeared in the processed subtree; after the forget step, such constraints will no longer be visible from the boundary.

At a join node, the two child states are compatible only if they induce the same visible wire order and the same boundary topological order. Their type labels are then merged. A direct-adjacency claim cannot be made independently on both sides unless it is the same visible claim; hidden paths are merged as hidden information. This rule reflects the separator property of the tree decomposition: the two processed subinstances can interact only through the bag.

A.6 Correctness and size of the dynamic program

Correctness follows by induction over the nice tree decomposition. The leaf case is immediate. For an introduce node, any realizing extension of the parent state restricts to a realizing child state after deleting the newly introduced symbol. Conversely, if a child extension exists and the introduce checks pass, then the only new constraints involving u are boundary-visible; adding the checked endpoint vertices to the child's acyclic endpoint graph preserves acyclicity. The bag-separator property ensures that u has no unrecorded interaction with vertices outside the current subtree.

The forget step is the same invariant in reverse. A realizable child state can forget u precisely when all constraints that become internal after deleting u have been checked. The type of the merged boundary pair contains exactly the information an outside continuation may need: either the gap is still empty, or it already hides a completed Hadamard path and therefore cannot be split by a future insertion. No absolute timing information is needed because all timing constraints are represented by the acyclicity of the endpoint graph.

For a join node, let H_1 and H_2 be the two partial endpoint graphs. They intersect only on boundary endpoint vertices. If the two child states agree on the boundary topological order, any directed cycle in $H_1 \cup H_2$ either lies inside one H_i or crosses the boundary at least twice. In the latter case, the cycle would force two boundary vertices to appear in opposite orders in the common boundary order, a contradiction. Therefore the union is acyclic. The converse is obtained by restricting any parent realization to the two child subtrees.

Backpointers stored with every accepted transition reconstruct the ordered wire lists and the direct-Hadamard pairs. The final call to CHECK performs a topological sort of the full endpoint graph and places the remaining CZ and CCZ gates at the corresponding overlap times. Thus the DP finds a realizing width- k circuit exactly when one exists.

For a bag of size $O(k)$, assigning visible symbols to k ordered boundary lists with sentinels, choosing the three-valued pair types, and choosing a boundary endpoint order yields $k^{6k+o(k)}$ states after the standard Stirling-type simplification. The number of nice bags is linear in n , and transition enumeration is bounded by the same state factor. The treewidth preprocessing costs $2^{O(k)}n$, and building the connection graph costs $O(m)$, so the total running time is $k^{6k+o(k)}n + O(m)$.

Quantum Speedups for Sampling and Non-Convex Optimization with Stochastic Oracles

Guneykan Ozgul ✉ 

Department of Computer Science and Engineering, Pennsylvania State University,
University Park, PA, USA

Global Technology Applied Research, JPMorgan Chase, New York, NY, USA

Xiantao Li ✉ 

Department of Mathematics, Pennsylvania State University, University Park, PA, USA

Mehrdad Mahdavi ✉

Department of Computer Science and Engineering, Pennsylvania State University,
University Park, PA, USA

Chunhao Wang ✉ 

Department of Computer Science and Engineering, Pennsylvania State University,
University Park, PA, USA

Abstract

We present quantum speedups for sampling from distributions of the form $\pi \propto e^{-f}$ on $\mathbb{R}^d$. We consider two stochastic oracle models: a stochastic gradient oracle, where $f = \frac{1}{n} \sum_{i=1}^n f_i$ and component gradients $\{\nabla f_i\}_{i \in [n]}$ are available, and a stochastic evaluation oracle, where only noisy values of f are available. Our framework accelerates classical stochastic Langevin Monte Carlo (LMC) and Hamiltonian Monte Carlo (HMC) algorithms by replacing stochastic gradient estimators with variance-controlled quantum mean estimation and gradient estimation subroutines. Unlike quantum walk based approaches, our algorithms do not require reversibility or exact gradients, and they preserve the structure of the underlying Markov chain. In the finite-sum setting, quantum mean estimation combined with classical variance-reduction techniques improves the stochastic gradient-query complexity for the approximate sampling task. In the stochastic zeroth-order setting, we develop gradient estimators robust to noisy function evaluations, yielding improved evaluation complexity for LMC and HMC. These results apply to strongly log-concave and/or non-log-concave distributions satisfying a log-Sobolev inequality, with convergence guarantees in Wasserstein distance and Kullback–Leibler divergence. We also show that faster sampling methods lead to quantum speedups for optimization, including for non-smooth and approximately convex objectives.

2012 ACM Subject Classification Theory of computation → Quantum computation theory; Mathematics of computing → Markov-chain Monte Carlo methods; Mathematics of computing → Stochastic processes

Keywords and phrases Quantum algorithms, sampling, Langevin Monte Carlo, Hamiltonian Monte Carlo, stochastic oracles, non-convex optimization

Digital Object Identifier 10.4230/LIPIcs.TQC.2026.8

Related Version *Full Version*: <https://arxiv.org/abs/2504.03626> [37]

Funding *Guneykan Ozgul*: National Science Foundation grant CCF-2238766 (CAREER).

Xiantao Li: National Science Foundation grants DMS-2111221 and CCF-2312456.

Chunhao Wang: National Science Foundation grant CCF-2238766 (CAREER).



© Guneykan Ozgul, Xiantao Li, Mehrdad Mahdavi, and Chunhao Wang;
licensed under Creative Commons License CC-BY 4.0

21st Conference on the Theory of Quantum Computation, Communication and Cryptography (TQC 2026).

Editors: Rotem Arnon and Aram W. Harrow; Article No. 8; pp. 8:1–8:25

Leibniz International Proceedings in Informatics



LIPICs Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

1 Introduction

1.1 Motivation

Efficient sampling from complex distributions is a fundamental problem in many scientific and engineering disciplines and it is becoming increasingly important as modern applications deal with high-dimensional data and complex probabilistic models. In machine learning, sampling plays an important role in Bayesian inference, as it facilitates posterior estimation and uncertainty quantification [47, 46, 22, 39]. In non-convex optimization, sampling allows for the exploration of complex energy landscapes and helps avoid local minima, facilitating progress in tasks such as resource allocation, scheduling, and hyperparameter tuning [51, 13]. It is also widely used in other domains of science, such as statistical mechanics [10, 23], and convex geometry [31, 18].

Given a potential function $f : \mathbb{R}^d \rightarrow \mathbb{R}$, we consider the problem of approximately sampling from *Gibbs-Boltzmann distribution* $\pi \in L^1(\mathbb{R}^d, d\mathbf{x})$ of the form

$$\pi(\mathbf{x}) = \frac{e^{-f(\mathbf{x})}}{\int e^{-f(\mathbf{x})} d\mathbf{x}}. \quad (1)$$

Solving this problem is in general difficult as it encodes NP-hard problems, and efficient sampling is only possible when f is highly structured. Given the challenging nature of the problem, it is natural to ask whether quantum computers can provide any speedup for the sampling task. The state-of-the-art classical algorithms are based on Markov chains, and this is a domain where quantum algorithms have been studied extensively [43, 48, 1]. In the continuous setting, Childs et al. [16] initiated the study of quantum algorithms for sampling from log-concave distributions, assuming that f is strongly convex and that ∇f can be accessed to very high precision. Their construction uses the quantum-walk framework of [43] together with fixed-point amplitude amplification similar to [48]. Although this approach improves the best classical dependence on certain parameters, the speedup does not directly apply in the stochastic setting because the relevant Markov chains are not *reversible*. Ozgul et al. [36] later addressed this limitation by analyzing a reversible proxy chain; as a result, the speedup is obtained relative to a suboptimal version of the best classical sampler.

Moreover, existing quantum results in this line primarily guarantee convergence in total variation distance. Obtaining quantum speedups with guarantees in Wasserstein distance or Kullback-Leibler divergence, metrics that are standard in classical sampling theory, remains largely open. Motivated by these challenges, we ask the following question:

► **Question 1.** *Can quantum computers improve the oracle complexity of sampling algorithms such that*

1. *the algorithm can use stochastic gradient and/or noisy evaluation oracles,*
2. *it can be based on fast (potentially nonreversible) Markov chains rather than a reversibilized surrogate, and*
3. *it provides guarantees in Wasserstein distance and Kullback–Leibler divergence?*

We answer Question 1 affirmatively by introducing a framework that replaces stochastic gradient computation with variance-controlled quantum subroutines, yielding provable oracle-complexity improvements while keeping the underlying nonreversible Markov chain unchanged. Conceptually, our contribution is not a new sampler, but a principled way to accelerate existing stochastic samplers by exploiting variance structure (via quantum mean/gradient estimation) rather than relying on restrictive quantum-walk techniques.

Another motivation for this work is to investigate whether quantum speedups for certain optimization problems can be obtained via our sampling algorithms. Sampling and optimization are tightly connected: sampling from a Gibbs distribution with potential βf at large inverse temperature β concentrates around minimizers of f . Although using sampling as a generic optimization method is not always optimal, it can be effective in regimes where gradient information is noisy or the objective is non-smooth/non-convex. To this end, we ask the following question:

► **Question 2.** *Can we use the quantum sampling algorithms to recover the existing quantum speedups for certain optimization problems and/or identify new classes of optimization problems that can be accelerated through sampling?*

We also answer Question 2 affirmatively by designing optimization algorithms based on our quantum samplers. In the finite-sum setting, we recover the quantum speedup of [50]; in the zeroth-order setting, we improve some parameter dependencies for non-smooth optimization problems studied in [29, 9]. While most of the resulting improvements are sub-quadratic, the point of our work is not to focus on the degree of the speedup for these particular problems. More broadly, we highlight that viewing optimization through the lens of sampling can be a useful organizing principle for identifying future quantum speedups. The key point is that sampling becomes competitive with standard optimization methods in regimes where (i) gradients are unavailable or prohibitively expensive, (ii) the objective is non-smooth (the gradients are not Lipschitz continuous), or (iii) the landscape is non-convex and escaping shallow traps is important. In these regimes, a low-temperature Gibbs distribution can serve as a powerful tool to approach these problems.

This perspective is also useful from a quantum-algorithmic standpoint for two related reasons. First, many standard first-order optimization routines (e.g., gradient descent) do not admit meaningful generic quantum speedups in the usual black-box models. Second, the sampling problem offers a more natural ground for quantum algorithms compared to deterministic iterative procedures. Moreover, in non-convex settings, which are a primary target for quantum acceleration, sampling-based methods are shown to be particularly effective [32].

We use Langevin Monte Carlo (LMC) and Hamiltonian Monte Carlo (HMC) algorithms as base samplers. LMC is the Euler-Maruyama discretization of the Langevin diffusion

$$d\mathbf{x}_t = -\nabla f(\mathbf{x}_t) dt + \sqrt{2} d\mathbf{B}_t.$$

With a stochastic gradient $\mathbf{g}_k$, the update at each iteration is

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \eta \mathbf{g}_k + \sqrt{2\eta} \epsilon_k, \quad (2)$$

where $\epsilon_k \sim \mathcal{N}(0, I_d)$ is distributed normally. HMC augments the state with a momentum $\mathbf{p}$ and simulates the Hamiltonian $H(\mathbf{x}, \mathbf{p}) = f(\mathbf{x}) + \|\mathbf{p}\|^2/2$. We give more detailed overview on these sampling algorithms in Appendix A. While they are effective, both algorithms are sensitive to gradient noise: smaller variance allows larger steps or fewer iterations, but producing a lower-variance gradient estimator costs more oracle queries.

In the finite-sum setting, a stochastic gradient can be obtained by sampling an index $i \in [n]$ and using $\nabla f_i(\mathbf{x})$ as an approximation to $\nabla f(\mathbf{x})$. This estimator is unbiased, but because of the large variance, the error due to stochasticity may be too large for sampling unless the step size is made very small. In zeroth-order settings, stochastic gradients are often built from finite differences or Gaussian smoothing. Such estimators have an additional dimension-dependent cost because a classical method must probe enough directions to

■ **Algorithm 1** Stochastic-gradient LMC and HMC templates.

input Current point $\mathbf{x}_0$, step size η , number of leapfrog steps S (for HMC only), stochastic gradient routine $\mathbf{g}(\cdot)$.

output One step LMC update or HMC proposal.

1: **LMC step:** sample $\epsilon_k \sim \mathcal{N}(0, I_d)$ and set

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \eta \mathbf{g}(\mathbf{x}_k) + \sqrt{2\eta} \epsilon_k.$$

2: **HMC proposal:** set $\mathbf{q}_0 = \mathbf{x}_0$ and sample $\mathbf{p}_0 \sim \mathcal{N}(0, I_d)$.

3: **for** $s = 0, \dots, S-1$ **do**

4: Perform the leapfrog updates

$$\mathbf{q}_{s+1} = \mathbf{q}_s + \eta \mathbf{p}_s - \frac{\eta^2}{2} \mathbf{g}(\mathbf{q}_s), \quad \mathbf{p}_{s+1} = \mathbf{p}_s - \frac{\eta}{2} \mathbf{g}(\mathbf{q}_s) - \frac{\eta}{2} \mathbf{g}(\mathbf{q}_{s+1}).$$

5: **end for**

6: **Return** the LMC point $\mathbf{x}_{k+1}$ or the HMC proposal $\mathbf{q}_S$.

recover a d -dimensional gradient. The quantum algorithms in this paper improve these two bottlenecks: the finite-sum algorithms reduce the cost of estimating $\nabla f = \frac{1}{n} \sum_{i=1}^n \nabla f_i$ using queries to $\{\nabla f_i\}_{i \in [n]}$, while the zeroth-order algorithms reduce the cost of estimating ∇f using queries to an oracle that returns f with some noise.

1.2 Quantum Oracle Model

Here, we describe our quantum input model and assumptions.

- The *stochastic gradient oracle*. In this setting, we assume $f(\mathbf{x}) = \frac{1}{n} \sum_{i=1}^n f_i(\mathbf{x})$, and we assume access to the gradients of each component, i.e., $\nabla f_i(x)$ for any $i \in [n]$. This is a relevant setting in statistics and machine learning problems, where access to the function f is provided only through sample data [47]. For each $i \in [n]$, we consider the quantum oracle access of the form

$$O_{\nabla f} |\mathbf{x}\rangle |i\rangle |0\rangle \mapsto |\mathbf{x}\rangle |i\rangle |\nabla f_i(\mathbf{x})\rangle. \quad (3)$$

- The *stochastic evaluation oracle*. In this setting, we assume that we have access only to noisy function values $f(\mathbf{x}; \xi)$, where ξ is the random seed that characterizes the noise. This setting is more relevant in contexts where the function value f is replaced by a statistical estimate, perhaps obtained through a numerical simulation [39]. The function is accessed through the binary oracle below.

$$O_f |\mathbf{x}\rangle |\xi\rangle |0\rangle \mapsto |\mathbf{x}\rangle |\xi\rangle |f(\mathbf{x}; \xi)\rangle. \quad (4)$$

Since this model only uses function values, we use the terms “zeroth-order oracle” and “evaluation oracle” interchangeably. We note that the evaluation oracle requires coherent access to the random seed ξ and that the same seed can be queried at multiple points. Although this model is stronger than a standard classical noisy oracle, we show that it naturally captures non-smooth optimization settings. This assumption is also explicit in the classical baseline work [39].

Both oracles can be implemented by making the corresponding classical oracles reversible with additional registers, so their query cost is comparable to the classical setting. As our study spans multiple settings under various assumptions, we state the precise noise assumptions separately for each setting.

1.3 Main Contributions

We summarize our technical contributions as follows.

Speedups for sampling in the stochastic gradient oracle setting. Using quantum mean estimation together with classical variance-reduction techniques such as stochastic variance-reduced gradients (SVRG) and control variates (CV), we obtain improved query complexities for sampling from both convex and non-convex potentials in Section 2. The main challenge is that these algorithms periodically compute the full gradient to control the variance of the stochastic gradients, and this full-gradient cost can dominate the improvement obtained from quantum mean estimation. To address this, we analyze the variance of the stochastic gradients and show that one can choose the step size and target variance online so that full-gradient computations are needed less frequently when quantum variance reduction is used. We also use these samplers to construct optimization algorithms for strongly convex functions.

Quantum speedups for gradient estimation via stochastic evaluation oracle. The challenge in this setting is that existing gradient estimation algorithms assume that the evaluation oracle is almost exact. However, in the setting we consider, the error between the estimated function value and the true function value can be unbounded. To address this, we develop quantum gradient estimation algorithms under several smoothness assumptions in Section 3. When the potential is smooth, our estimator reduces the number of evaluation queries from $\tilde{O}(d^2\sigma^2/\epsilon^2)$ to $\tilde{O}(d\sigma/\epsilon)$ for computing an ϵ -accurate gradient estimate (Theorem 3.4), where σ^2 is the noise variance in Assumption 3.1. When the stochastic functions are also smooth with high probability, we improve the dimension dependence by an additional factor of $d^{1/2}$ (Theorem 3.7). Our gradient estimation algorithms could be useful as independent tools, especially in zeroth-order stochastic optimization.

Speedups for stochastic zeroth-order sampling. In Section 4, we combine our quantum gradient estimation algorithm with HMC and LMC and analyze the convergence of the resulting samplers. We prove that, under the same assumptions, these hybrid algorithms use fewer queries to the evaluation oracle than the best-known classical samplers (Theorems 4.1 and 4.2).

Application to non-smooth non-convex optimization. In Section 5, we extend our quantum sampling methods to optimize non-convex functions with specific structural properties without the smoothness assumption. In particular, we show that Lipschitz continuous and approximately convex functions can be optimized with fewer stochastic-evaluation queries than the best-known classical algorithms in terms of dimension dependence (Theorem 5.4).

To help the readers navigate, we summarize the problems, oracles, assumptions, and results in Table 1.

1.4 Related Work

Non-asymptotic convergence rates for stochastic LMC and HMC have been analyzed extensively by [38, 49, 54, 19] and [12, 52], respectively. In the finite-sum setting, more sophisticated variance-reduction techniques, such as SVRG, SAGA, SARAH, and control variates (CV) [26, 20, 35, 3], have been used to reduce stochastic-gradient variance by using gradient information from previous iterations. Although these methods were originally

■ **Table 1** Overview of the main results. The table separates assumptions on the oracle/noise model from assumptions on the target distribution or optimization objective.

Problem	Oracle	Oracle/noise assumptions	Target or objective assumptions	Results
Sampling	Stochastic gradient	Lipschitz stochastic gradients	Strong convexity and Smoothness	Theorems 2.6 and 2.7
Sampling	Stochastic gradient	Lipschitz stochastic gradients	Log-Sobolev inequality and Smoothness	Theorem 2.3
Sampling	Stochastic evaluation	Bounded noise variance	Strong convexity and Smoothness	Theorem 4.1
Sampling	Stochastic evaluation	Bounded noise variance	Log-Sobolev inequality and Smoothness	Theorem 4.2
Optimization	Stochastic evaluation	Uniform error bound	Approximate convexity and Lipschitz continuity	Theorem 5.4

introduced in the context of optimization, successive works have applied these methods to improve sampling efficiency via LMC [21, 11, 3, 28] and HMC [53, 52]. In particular, [52] has incorporated various variance reduction techniques to SG-HMC and analyzed convergence in Wasserstein distance for smooth and strongly convex potentials. In the non-log-concave setting, [28] has analyzed the convergence of SVRG-LMC and SARAH-LMC for target distributions that satisfy the log-Sobolev inequality and applied their results to optimize structured non-convex objectives.

In quantum sampling, most existing results use quantum walks, which can speed up certain Markov Chain Monte Carlo (MCMC) methods by improving the mixing time of the underlying Markov chain [43, 41, 42, 48, 6]. These methods have been used in several domains to reduce the computational time of sampling-related tasks [33, 1, 16, 6, 29, 8]. For continuous distributions, [16, 36] used the quantum-walk framework to reduce the number of gradient queries needed to approximately sample from strongly log-concave and non-log-concave distributions.

In the context of optimization, quantum algorithms such as multi-dimensional quantum mean estimation [17] and quantum gradient estimation [27, 24] have shown promise in reducing the computational cost associated with gradient-based methods [44, 7, 40, 50, 30]. These techniques are particularly well-suited for addressing challenges in large-scale and noisy settings, as they can provide more accurate gradient estimates with fewer queries. However, these methods have not been considered for sampling tasks, and this paper focuses on integrating these quantum techniques to enhance the efficiency of stochastic gradient-based samplers and alleviate the computational burden inherent in classical methods.

1.5 Notation and Glossary

Bold symbols, such as $\mathbf{x}$ and $\mathbf{y}$, are used to represent vectors, with $\|\cdot\|$ indicating the Euclidean or operator norm depending on the context. Given two scalars a and b , we use $a \wedge b$ to denote $\min\{a, b\}$ and use $a \vee b$ to denote $\max\{a, b\}$. We use $\mathcal{B}_d(c, r)$ to denote the d -dimensional ball centered at c with radius r and $G_d^l(c)$ to denote the d -dimensional grid centered at point c with side length l . We occasionally use G_d^l when the center of the grid is clear from the context. The notation $\tilde{O}$ is used to suppress the polylogarithmic dependencies on d, ϵ, L, μ and α that will be defined later in the text.

We use several metrics to compare probability distributions over a state space $\mathcal{X}$. Let π and μ be two probability distributions on $\mathcal{X}$. The p -Wasserstein distance between π and μ is defined as $W_p(\pi, \mu) = (\inf_{\gamma \in \Gamma(\pi, \mu)} \mathbb{E}_{(\mathbf{x}, \mathbf{y}) \sim \gamma} \|\mathbf{x} - \mathbf{y}\|^p)^{1/p}$ where $\Gamma(\pi, \mu)$ is the set of all joint distributions $\gamma(\mathbf{x}, \mathbf{y})$ whose marginals are π and μ . The KL divergence of π with respect to μ is defined as $\text{KL}(\pi \| \mu) = \int_{\mathcal{X}} d\mathbf{x} \pi(\mathbf{x}) \log \left(\frac{\pi(\mathbf{x})}{\mu(\mathbf{x})} \right)$ and the relative Fisher information is $\text{FI}(\pi \| \mu) = \int_{\mathcal{X}} d\mathbf{x} \pi(\mathbf{x}) \left\| \nabla \log \left(\frac{\pi(\mathbf{x})}{\mu(\mathbf{x})} \right) \right\|^2$. The total variation distance is defined as $\text{TV}(\pi, \mu) = \sup_{A \subseteq \mathcal{X}} |\pi(A) - \mu(A)| = \frac{1}{2} \int_{\mathcal{X}} d\mathbf{x} |\pi(\mathbf{x}) - \mu(\mathbf{x})|$.

1.6 Organization of the paper

In Section 2, we present our quantum sampling algorithms for stochastic gradient oracles. In Section 3, we introduce the stochastic zeroth-order gradient estimation algorithm that serves as the main technical tool for the sampling results in Section 4. Finally, in Section 5, we show how these sampling results can be used to speed up non-convex optimization.

2 Quantum Speedups for Finite-Sum Sampling and Optimization via Gradient Oracle

Throughout this section, we consider $f = \frac{1}{n} \sum_{i=1}^n f_i$, and use the quantum stochastic-gradient oracle in Equation (3). Our goal is to sample approximately from π using as few oracle queries as possible. We first introduce stochastic-gradient estimators that combine unbiased quantum mean estimation with classical variance-reduction frameworks, yielding improved query complexity for both LMC and HMC. Classically, a stochastic gradient at $\mathbf{x}_t$ is often formed by subsampling a mini-batch $B \subseteq [n]$ and setting $\mathbf{g}_t = |B|^{-1} \sum_{i \in B} \nabla f_i(\mathbf{x}_t)$. Then $\mathbb{E}_B[\mathbf{g}_t] = \nabla f(\mathbf{x}_t)$, so $\mathbf{g}_t$ is an unbiased estimator. If LMC or HMC takes T iterations to converge, the resulting query complexity is $\mathcal{O}(|B|T)$, and there is a tradeoff between the batch size $|B|$ and the number of iterations T .

To decouple these quantities, variance-reduction methods such as SVRG periodically compute a full gradient $\nabla f(\tilde{\mathbf{x}})$ and use it to correct later stochastic gradients. Control variates (CV) use a similar correction based on the initial gradient $\nabla f(\mathbf{x}_0)$. Our quantum LMC algorithms preserve this structure, but replace mini-batch averaging with unbiased quantum mean estimation; see Algorithms 2 and 3.

For QSVRG, the epoch length m determines how often the full gradient is computed. The parameter b is the quantum analogue of the batch size and is chosen analytically. In QCV, the stochastic gradient is corrected using the initial iterate, as in classical control variates. We also highlight that quantum mean estimation requires a target variance level as input, unlike classical subsampling. Although the optimal target variance changes along the trajectory and depends on quantities that are difficult to compute, QSVRG uses $L^2 \|\mathbf{x}_k - \tilde{\mathbf{x}}\|^2 / b^2$, where $\tilde{\mathbf{x}}$ is the most recent point at which the full gradient was computed (Algorithm 2). Similarly, QCV uses $L^2 \|\mathbf{x}_k - \mathbf{x}_0\|^2 / b^2$, where $\mathbf{x}_0$ is the initial point (Algorithm 3). Note that we only use QCV for HMC but (Algorithm 3) uses LMC update for ease of exposition. Changing the update rule with HMC proposal in Algorithm 1 implements QCV-HMC.

Although the algorithm is simple, proving the speedup is more delicate because the cost of full-gradient computations must also be reduced. We show that quantum mean estimation enables less frequent full-gradient computations through improved variance reduction. Our analysis quantifies stochastic-gradient variance along the algorithm trajectory and balances the frequency of full-gradient computations with the target variance used in quantum mean

Algorithm 2 QSVRG-LMC.

input Stochastic gradient oracle $O_{\nabla f}$, initial point $\mathbf{x}_0$, number of iterations K , step size η , smoothness constant L , variance scale factor b , epoch length m .

output Final iterate $\mathbf{x}_K$.

- 1: Set $\tilde{\mathbf{x}} = \mathbf{x}_0$ and $\tilde{\mathbf{g}} = \nabla f(\tilde{\mathbf{x}})$.
- 2: **for** $k = 0, \dots, K - 1$ **do**
- 3: **if** $k \bmod m = 0$ **then**
- 4: Set $\tilde{\mathbf{x}} = \mathbf{x}_k$ and $\tilde{\mathbf{g}} = \nabla f(\tilde{\mathbf{x}})$.
- 5: Set $\mathbf{g}_k = \tilde{\mathbf{g}}$.
- 6: **else**
- 7: Define the quantum sampling oracle $O_{\text{SVRG}}^{\mathbf{x}_k, \tilde{\mathbf{x}}}$ by

$$|0\rangle |0\rangle \mapsto \frac{1}{\sqrt{n}} \sum_{i=1}^n |\nabla f_i(\mathbf{x}_k) - \nabla f_i(\tilde{\mathbf{x}}) + \tilde{\mathbf{g}}\rangle |i\rangle.$$

- 8: Set $\hat{\sigma}_k^2 = L^2 \|\mathbf{x}_k - \tilde{\mathbf{x}}\|^2 / b^2$.
 - 9: Set $\mathbf{g}_k = \text{QuantumMeanEstimation}(O_{\text{SVRG}}^{\mathbf{x}_k, \tilde{\mathbf{x}}}, \hat{\sigma}_k^2)$.
 - 10: **end if**
 - 11: Sample $\epsilon_k \sim \mathcal{N}(0, I_d)$.
 - 12: Set $\mathbf{x}_{k+1} = \mathbf{x}_k - \eta \mathbf{g}_k + \sqrt{2\eta} \epsilon_k$.
 - 13: **end for**
 - 14: **Return** $\mathbf{x}_K$.
-

estimation. This yields improved complexity bounds in both strongly convex and non-convex settings. We first present the LMC result without assuming convexity; later, we consider HMC with similar stochastic-gradient estimators to further improve some parameter dependencies.

► **Assumption 2.1** (Lipschitz Stochastic Gradients). There exists a positive constant L such that for all $\mathbf{x}, \mathbf{y} \in \mathbb{R}^d$ and all functions f_i , $i = 1, \dots, n$, it holds that

$$\|\nabla f_i(\mathbf{x}) - \nabla f_i(\mathbf{y})\| \leq L \|\mathbf{x} - \mathbf{y}\|. \quad (5)$$

This is a standard assumption in the optimization and sampling literature [45, 32].

2.1 Sampling under Log-Sobolev Inequality via Langevin Monte Carlo

We use the SVRG-LMC algorithm of [28] as the base sampler and replace its stochastic-gradient estimator with the quantum estimator in QSVRG-LMC; see Algorithm 2. In this section, we assume that the target distribution satisfies a log-Sobolev inequality.

► **Assumption 2.2** (Log-Sobolev Inequality). We say that π satisfies the log-Sobolev inequality with constant α if for all ρ , it holds that

$$\text{KL}(\rho \| \pi) \leq \frac{1}{2\alpha} \text{FI}(\rho \| \pi). \quad (6)$$

The log-Sobolev inequality is a sampling analog of the Polyak-Łojasiewicz (PL) condition used in optimization [15]. It is standard in the non-log-concave sampling literature [45, 32, 14, 28]. Moreover, if f is μ -strongly convex, then π satisfies an LSI with constant μ . We also note that this assumption is weaker than the dissipative gradient condition [38, 53] which is commonly used in non-log-concave sampling.

Algorithm 3 QCV-LMC.

input Stochastic gradient oracle $O_{\nabla f}$, initial point $\mathbf{x}_0$, number of iterations K , step size η , smoothness constant L , variance scale factor b .

output Final iterate $\mathbf{x}_K$.

- 1: Set $\mathbf{g}^{(0)} = \nabla f(\mathbf{x}_0)$.
- 2: **for** $k = 0, \dots, K-1$ **do**
- 3: Define the quantum sampling oracle $O_{\text{CV}}^{\mathbf{x}_k, \mathbf{x}_0}$ by

$$|0\rangle |0\rangle \mapsto \frac{1}{\sqrt{n}} \sum_{i=1}^n |\nabla f_i(\mathbf{x}_k) - \nabla f_i(\mathbf{x}_0) + \mathbf{g}^{(0)}\rangle |i\rangle.$$

- 4: Set $\hat{\sigma}_k^2 = L^2 \|\mathbf{x}_k - \mathbf{x}_0\|^2 / b^2$.
 - 5: Set $\mathbf{g}_k = \text{QuantumMeanEstimation}(O_{\text{CV}}^{\mathbf{x}_k, \mathbf{x}_0}, \hat{\sigma}_k^2)$.
 - 6: Sample $\boldsymbol{\epsilon}_k \sim \mathcal{N}(0, I_d)$.
 - 7: Set $\mathbf{x}_{k+1} = \mathbf{x}_k - \eta \mathbf{g}_k + \sqrt{2\eta} \boldsymbol{\epsilon}_k$.
 - 8: **end for**
 - 9: **Return** $\mathbf{x}_K$.
-

► **Theorem 2.3** (Main Theorem for QSVRG-LMC). *Let μ_k be the distribution of $\mathbf{x}_k$ in QSVRG-LMC algorithm. Suppose that f satisfies Assumptions 2.1 and 2.2. Then for $\eta = \mathcal{O}\left(\frac{\epsilon\alpha}{dL^2} \wedge \frac{\alpha}{L^2m}\right)$, $K = \tilde{\mathcal{O}}\left(\frac{L^2 \log(\text{KL}(\mu_0|\pi))}{\alpha^2} (n^{2/3} + \frac{d}{\epsilon})\right)$, $b = \tilde{\mathcal{O}}(n^{1/3})$, and $m = \tilde{\mathcal{O}}(n^{2/3})$ we have*

$$\left\{ \text{KL}(\mu_K|\pi), \text{TV}(\mu_K, \pi)^2, \frac{\alpha}{2} \text{W}_2(\mu_K, \pi)^2 \right\} \leq \epsilon.$$

The stochastic-gradient oracle query complexity is $\tilde{\mathcal{O}}\left(\frac{L^2 \log(\text{KL}(\mu_0|\pi))}{\alpha^2} \left(nd^{1/2} + \frac{d^{3/2}n^{1/3}}{\epsilon}\right)\right)$.

The query complexity in [28] is $\mathcal{O}(n + n^{1/2}\epsilon^{-1})$, whereas ours is $\mathcal{O}(n + n^{1/3}\epsilon^{-1})$. The coupled term $n^{1/3}\epsilon^{-1}$ is possible because the stochastic-gradient cost Kb and the full-gradient cost nK/m are balanced. In classical SVRG-LMC, the optimal choice is $m = b = n^{1/2}$, where b is the inner batch size; in our algorithm, we have $m = n^{2/3}$ and $b = n^{1/3}$. Thus, quantum gradient estimators allow the exact gradient to be computed less frequently, which is the main mechanism behind the speedup.

2.2 Application to Optimization

In this section, we derive a finite-sum optimization algorithm by running our sampler at low temperature, equivalently by scaling f by a large inverse-temperature parameter β . The following theorem quantifies how large β should be to find an approximate optimizer of f .

► **Theorem 2.4** (Quantum Finite-sum Optimization Upper Bound). *Suppose f is a μ -strongly convex function. Then, there exists a quantum algorithm that outputs a point $\tilde{\mathbf{x}}$ such that $\mathbb{E}[f(\tilde{\mathbf{x}}) - f(\mathbf{x}^*)] \leq \epsilon$ using $\tilde{\mathcal{O}}\left(\frac{L^2 d}{\mu^2} \left(nd^{1/2} + \frac{d^{3/2}n^{1/3}}{\epsilon}\right)\right)$ queries to the quantum stochastic gradient oracle.*

This matches the upper bound of [50] in its n -dependence. The remaining parameter dependencies are suboptimal; this is expected because, in the strongly convex setting, sampling is not the most efficient optimization method in terms of parameters such as L and μ . We have simplified the proof by assuming strong convexity, but one can also impose

additional conditions, such as the Morse-type conditions used in [28], to obtain a quantum upper bound of the form $\mathcal{O}(n + n^{1/3}\epsilon^{-1})$ for a broader class of non-convex functions satisfying an LSI.

2.3 Sampling under Strong Convexity via Hamiltonian Monte Carlo

Next, we consider quantum speedups for Hamiltonian Monte Carlo (HMC) for strongly convex potentials. We replace the stochastic gradients in HMC with quantum-estimated stochastic gradients based on the SVRG and CV estimators used in Algorithms 2 and 3.

► **Assumption 2.5** (Strong Convexity). There exists a positive constant μ such that for all $\mathbf{x}, \mathbf{y} \in \mathbb{R}^d$ it holds that

$$f(\mathbf{x}) \geq f(\mathbf{y}) + \langle \nabla f(\mathbf{y}), \mathbf{x} - \mathbf{y} \rangle + \frac{\mu}{2} \|\mathbf{x} - \mathbf{y}\|^2. \quad (7)$$

For smooth objectives, we occasionally use the condition number $\kappa = L/\mu$ to express the dependence on L and μ more compactly.

Based on Assumptions 2.1 and 2.5, we give the main theorem for the quantum Hamiltonian Monte Carlo algorithm implemented with the quantum SVRG estimator.

► **Theorem 2.6** (Main Theorem for QSVRG-HMC). *Let μ_k be the distribution of $\mathbf{x}_k$ in QSVRG-HMC algorithm. Suppose that f satisfies Assumptions 2.1 and 2.5. Given that the initial point $\mathbf{x}_0$ satisfies $\|\mathbf{x}_0 - \arg \min_{\mathbf{x}} f(\mathbf{x})\| \leq \frac{d}{\mu}$, then, for $\eta = \mathcal{O}\left(\frac{\epsilon}{L^{1/2}d^{1/2}\kappa^{3/2}}\right)$, $S = \tilde{\mathcal{O}}\left(\frac{Ld^{1/2}\kappa^{3/2}}{\epsilon}\right)$, $T = \tilde{\mathcal{O}}(1)$, $b = \mathcal{O}\left(\frac{L^{1/8}\epsilon^{1/4}n^{1/2}}{d^{1/8}\kappa^{3/8}} \vee 1\right)$, and $m = n/b$, we have*

$$W_2(\mu_{ST}, \pi) \leq \epsilon.$$

The total query complexity to the stochastic gradient oracle is $\tilde{\mathcal{O}}\left(\frac{Ld^{1/2}\kappa^{3/2}}{\epsilon} + \frac{L^{9/8}d^{7/8}\kappa^{3/4}n^{1/2}}{\epsilon^{3/4}}\right)$.

We obtain an analogous guarantee for HMC with the quantum control-variate estimator.

► **Theorem 2.7** (Main Theorem for QCV-HMC). *Let μ_k be the distribution of $\mathbf{x}_k$ in QCV-HMC algorithm. Suppose that f satisfies Assumptions 2.1 and 2.5. Given that the initial point $\mathbf{x}_0$ satisfies $\|\mathbf{x}_0 - \arg \min_{\mathbf{x}} f(\mathbf{x})\| \leq \frac{d}{\mu}$, then, for $\eta = \mathcal{O}\left(\frac{\epsilon}{L^{1/2}d^{1/2}\kappa^{3/2}}\right)$, $S = \tilde{\mathcal{O}}\left(\frac{Ld^{1/2}\kappa^{3/2}}{\epsilon}\right)$, $T = \tilde{\mathcal{O}}(1)$, and $b = \mathcal{O}\left(\frac{d^{1/4}\kappa^{3/4}}{L^{1/4}\epsilon^{1/2}} \vee 1\right)$, we have*

$$W_2(\mu_{ST}, \pi) \leq \epsilon.$$

The total query complexity to the stochastic gradient oracle is $\tilde{\mathcal{O}}\left(\frac{Ld^{5/4}\kappa^{9/4}}{\epsilon^{3/2}}\right)$.

The proofs follow the same idea as in the LMC setting. We express the stochastic-gradient variance σ along the HMC trajectory in terms of b and the distance between the current iterate and the last iterate at which the full gradient was computed. We then choose b so that the per-iteration cost of quantum mean estimation, $\tilde{\mathcal{O}}(\sigma/\hat{\sigma})$, balances the amortized full-gradient calculation cost, $\tilde{\mathcal{O}}(n/m)$, since the full gradient is computed only once every m iterations.

Theorems 2.6 and 2.7 imply that when $n = \mathcal{O}(\epsilon^{-1/2})$ the best classical (SVRG-HMC) and the best quantum (QSVRG-HMC) algorithms have $\tilde{\mathcal{O}}(\epsilon^{-1})$ gradient complexity. On the other hand, when $n = \omega(\epsilon^{-1})$, quantum algorithms have better complexity than the best classical algorithms; the better of QSVRG-HMC and QCV-HMC depends on the size of n .

■ **Table 2** Summary of the results in the **stochastic gradient oracle setting** (some of the previous results use a different scaling of f and we convert the results to the same scaling as ours in the table). Here, we mainly focus on n and ϵ dependency. See Theorems 2.3, 2.6, and 2.7 for explicit dependencies on L, μ, α, d .

Algorithm	Assumptions	Metric	Gradient Complexity
SG-HMC [52]	Strongly Convex	W_2	$\tilde{O}(n\epsilon^{-2})$
SVRG-HMC [52]	Strongly Convex	W_2	$\tilde{O}(n^{2/3}\epsilon^{-2/3} + \epsilon^{-1})$
SAGA-HMC [52]	Strongly Convex	W_2	$\tilde{O}(n^{2/3}\epsilon^{-2/3} + \epsilon^{-1})$
CV-HMC [52]	Strongly Convex	W_2	$\tilde{O}(\epsilon^{-2})$
SRVR-HMC [53]	Dissipative Gradients	W_2	$\tilde{O}(n + n^{1/2}\epsilon^{-2} + \epsilon^{-4})$
SVRG-LMC [28]	LSI	KL	$\tilde{O}(n + n^{1/2}\epsilon^{-1})$
SARAH-LMC [28]	LSI	KL	$\tilde{O}(n + n^{1/2}\epsilon^{-1})$
QSVRG-HMC [Theorem 2.6]	Strongly Convex	W_2	$\tilde{O}(n^{1/2}\epsilon^{-3/4} + \epsilon^{-1})$
QCV-HMC [Theorem 2.7]	Strongly Convex	W_2	$\tilde{O}(\epsilon^{-3/2})$
QSVRG-LMC [Theorem 2.3]	LSI	KL ¹	$\tilde{O}(n + n^{1/3}\epsilon^{-1})$

► **Remark 2.8.** Both the classical algorithms in [52] and quantum algorithms in this paper assume that the starting point is (d/μ) -close to the minimizer $\mathbf{x}^* = \arg \min f(\mathbf{x})$. In case this point is not given, it can be obtained using $\mathcal{O}(n)$ iterations of SGD [3].

3 Quantum Gradient Estimation in Zeroth-Order Stochastic Setting

We characterize the complexity of our algorithms in this section with respect to the stochastic zeroth-order oracle defined in Equation (4). Jordan’s quantum gradient estimation algorithm [27] constructs a superposition over d -dimensional N grid points G_d^l centered around the point $\mathbf{x}$ with side length l . Then the following quantum state is obtained by using a phase oracle $O : |\mathbf{x}\rangle \mapsto e^{itf(\mathbf{x})} |\mathbf{x}\rangle$,

$$|\psi\rangle = \frac{1}{\sqrt{N^d}} \sum_{\mathbf{y} \in G_d^l(\mathbf{x})} e^{it[f(\mathbf{x} + \frac{l}{N}(\mathbf{y} - \frac{N}{2})) - f(\mathbf{x})]} |\mathbf{y}\rangle \quad (8)$$

where t is a proper scale factor. Then the algorithm uses quantum Fourier transform to compute the full gradient. This algorithm uses only a constant number of queries to the function f , whereas one needs at least d queries to approximate the gradient classically. We refer the reader to [7, 24] and the full version of this paper for a more detailed review of Jordan’s quantum estimation algorithm. We occasionally refer to the version of this algorithm with optimized parameters as **QuantumGradient** given in [7]. We also review this algorithm in Appendix B.2 and give the description in Algorithm 7.

Jordan’s quantum gradient estimation algorithm [27] constructs a superposition over a d -dimensional grid $G_d^l(\mathbf{x})$ centered at $\mathbf{x}$, applies a phase oracle $O : |\mathbf{x}\rangle \mapsto e^{itf(\mathbf{x})} |\mathbf{x}\rangle$, and then applies a quantum Fourier transform to recover the gradient. In the deterministic setting, this requires only a constant number of function queries, whereas a classical finite-difference method needs at least d queries.

¹ Convergence in KL divergence implies convergence in squared TV and W_2 distances due to Pinsker’s and Talagrand’s inequalities.

3.1 Gradient Estimation for Smooth Potentials

We assume the following about the evaluation oracle for f .

► **Assumption 3.1** (Bounded Noise). For any $\mathbf{x} \in \mathbb{R}^d$, the stochastic zeroth-order oracle outputs an estimator $f(\mathbf{x}; \xi)$ of $f(\mathbf{x})$ such that $\mathbb{E}[f(\mathbf{x}; \xi)] = f(\mathbf{x})$, $\mathbb{E}[\nabla f(\mathbf{x}; \xi)] = \nabla f(\mathbf{x})$, and $\mathbb{E}\|\nabla f(\mathbf{x}; \xi) - \nabla f(\mathbf{x})\|^2 \leq \sigma^2$.

We note that gradients of $f(\cdot; \xi)$ are not queried by the algorithm and only appear in the assumption.

► **Assumption 3.2** (Smoothness). The potential function $f : \mathbb{R}^d \rightarrow \mathbb{R}$ has L -Lipschitz gradients. Specifically, for any $\mathbf{x}, \mathbf{y} \in \mathbb{R}^d$ it holds that

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

The assumption is broader than an additive-noise model because it also covers certain multiplicative-noise models. For example, suppose $f : \mathcal{B}_d(0, R) \rightarrow \mathbb{R}$ is L -smooth and differentiable, and the stochastic components have the form $f(\mathbf{x}; \xi) = \xi f(\mathbf{x})$, where $\mathbb{E}[\xi] = 1$ and $\mathbb{E}[(\xi - 1)^2] \leq \frac{\sigma^2}{4L^2 R^2}$. Then Assumption 3.1 is satisfied.

We assume that f can be queried with the same randomness at two points, so that $f(\mathbf{x}; \xi_i)$ and $f(\mathbf{y}; \xi_i)$ can be evaluated using the same seed ξ_i . In the finite-sum case, for instance, one can prepare $\frac{1}{\sqrt{n}} \sum_{i=1}^n |0\rangle |i\rangle$ and then, using the binary oracle, prepare $\frac{1}{\sqrt{n}} \sum_{i=1}^n |f(\mathbf{x}; i)\rangle |i\rangle$. This requires no QRAM-style data encoding, since the superposition is only over the indices. Classically, the gradient in this two-point setting can be estimated using the Gaussian smoothing technique. This involves sampling random directions from the extended space around the target point and performing two-point evaluations to approximate the gradient. Specifically, the gradient can be approximated as:

$$\mathbf{g}_{\nu, b}(\mathbf{x}) = \frac{1}{b} \sum_{i=1}^b \frac{f(\mathbf{x} + \nu \mathbf{u}_i; \xi_i) - f(\mathbf{x}; \xi_i)}{\nu} \mathbf{u}_i, \quad (9)$$

where $\mathbf{u}_i \sim \mathcal{N}(0, I_d)$ are independent and identically distributed random vectors. Authors in [4] showed that for any $\mathbf{x} \in \mathbb{R}^d$, the estimator $\mathbf{g}_{\nu, b}$ satisfies $\mathbb{E}\|\mathbf{g}_{\nu, b}(\mathbf{x}) - \nabla f(\mathbf{x})\|^2 \leq \frac{4(d+5)(\|\nabla f(\mathbf{x})\|^2 + \sigma^2)}{b} + \frac{3\nu^2 L^2 (d+3)^3}{2}$. Although the squared norm of the gradient on the right-hand side is unbounded, it is typically of order $\tilde{\mathcal{O}}(d)$ in expectation throughout the trajectory of LMC. Consequently, this method requires $b = \mathcal{O}(d^2/\epsilon^2)$ function evaluations to achieve an ϵ -accurate gradient estimate in the L_2 norm.

To our knowledge, no existing quantum algorithm estimates the gradient in this particular setting. Our goal is to develop additional tools that allow Jordan's quantum estimation algorithm [27] to be used under this oracle model. The main technical tool is the following robust phase oracle.

► **Proposition 3.3.** Let $X \in \mathbb{R}$ be a random variable such that $\mathbb{E}\|X - \mathbb{E}[X]\|^2 \leq \sigma^2$. Given two reals $t \geq 0$ and $\epsilon \in (0, 1)$, then there is a unitary operator $P_{t, \epsilon}^X : |0\rangle |0\rangle \mapsto |\phi_X\rangle |0\rangle$ acting on $\mathcal{H}_X \otimes \mathcal{H}_{aux}$ that can be implemented using $\tilde{\mathcal{O}}(t\sigma \log(1/\epsilon))$ quantum experiments and binary oracle queries to X such that

$$\| |\phi_X\rangle - e^{it\mathbb{E}[X]} |0\rangle \| \leq \epsilon,$$

with probability at least $8/9$.

This phase oracle is similar to the oracle implemented in [17], but their construction requires $\|X\| \leq 1$, whereas X may be unbounded in our setting. Thus Proposition 3.3 extends the phase oracle to unbounded random variables by using a sequence of unitaries at different truncation levels. In our gradient estimation algorithm, this oracle replaces the phase oracle in Algorithm 7, with $X = f(\mathbf{x}; \xi)$. Since Jordan’s algorithm is biased and succeeds with high probability, we postprocess its output using a multilevel Monte Carlo algorithm (See Appendix B.3 for overview) to obtain a smooth unbiased gradient estimator.

► **Theorem 3.4.** *Suppose that the potential function f satisfies Assumptions 3.1 and 3.2 and further suppose that $\|\nabla f(\mathbf{x})\| \leq M^2$ for all $\mathbf{x} \in \mathcal{B}(x^*, \text{poly}(d))$. Then, given a real $\mathbf{x} \in \mathcal{B}(x^*, \text{poly}(d))$ and $\hat{\sigma} > 0$, there exists a quantum algorithm that outputs a random vector $\mathbf{g}$ such that*

$$\mathbb{E}[\mathbf{g}] = \nabla f(\mathbf{x}), \quad \text{and} \quad \mathbb{E}\|\mathbf{g} - \nabla f(\mathbf{x})\|^2 \leq \hat{\sigma}^2$$

using $\tilde{O}(\frac{\sigma d}{\hat{\sigma}})$ queries to the stochastic evaluation oracle.

Although Theorem 3.4 is motivated by the sampling task, it might be of independent interest for various optimization problems even in non-smooth settings. For example, suppose that f_ξ is a non-smooth but locally L -Lipschitz function on the grid G_d^l . We define $f_v(\mathbf{x}) = \mathbb{E}_{\xi \in \Xi, \mathbf{u} \sim \mathcal{B}(0,1)}[f(\mathbf{x} + v\mathbf{u}; \xi)]$. Then, let $\mathbf{y} \in G_d^l$, and we have $\mathbb{E}\|\nabla f(\mathbf{y} + v\mathbf{u}) - \nabla f_v(\mathbf{y})\|^2 \leq 4L^2$. It is known that f_v is a smooth function with smoothness parameter $O(Ld^{1/2}v^{-1})$. Hence, by Theorem 3.4 our algorithm outputs an unbiased estimator $\mathbf{g}$ such that $\mathbb{E}[\mathbf{g}] = \nabla f_v(\mathbf{x})$ and $\mathbb{E}\|\mathbf{g} - \nabla f_v(\mathbf{x})\|^2 \leq \hat{\sigma}^2$ using $\tilde{O}(\frac{Ld}{\hat{\sigma}})$ queries to f_ξ . This result has recently been established in [30], and it is a special case of Theorem 3.4. Hence, our quantum gradient estimation can give speedups beyond the settings considered in [30], but this is outside the scope of this work.

3.2 Gradient Estimation under Additional Smoothness Assumption

In this section, we impose a slightly stronger smoothness assumption on the stochastic functions f_ξ , which further improves the dimension dependence.

► **Assumption 3.5** (Lipschitz Stochastic Gradients). The stochastic component $f(\cdot; \xi) : \mathbb{R}^d \rightarrow \mathbb{R}$ has $L(\xi)$ -Lipschitz gradients for any $\xi \in \Xi$. Specifically, it holds that

$$\|\nabla f(\mathbf{x}; \xi) - \nabla f(\mathbf{y}; \xi)\| \leq L(\xi)\|\mathbf{x} - \mathbf{y}\|, \quad (10)$$

and the expected Lipschitz constant satisfies $\mathbb{E}[L(\xi)] = L$.

Assumption 3.5 is weaker than assuming that each stochastic function f_ξ has L -Lipschitz gradients, and it is straightforward to show that Assumption 3.5 implies that f has Lipschitz gradients.

In this section, we assume the following sampling oracle,

$$O_\xi : |\mathbf{x}\rangle |0\rangle \mapsto \sum_{\xi \in \Xi} \sqrt{\Pr(\xi)} |\mathbf{x}\rangle |\xi\rangle. \quad (11)$$

² One can show that the norm of the gradient is bounded by a function of problem parameters in such a large ball throughout the trajectory of HMC or LMC due to smoothness. Since the dependency on M is logarithmic, we do not give an explicit bound on M .

8:14 Quantum Speedups for Sampling and Optimization

Instead of implementing an accurate phase oracle, one can first estimate $\nabla f(\mathbf{x}; \xi)$ and then use quantum mean estimation to compute $\nabla f(\mathbf{x})$. Quantum mean estimation computes the mean of a random variable with quadratically fewer samples than classical mean estimation (Lemma B.1).

However, Assumption 3.5 implies that f_ξ might not be a smooth function (even if f is smooth), which is the requirement in [7]. Hence, Jordan's algorithm might fail to compute the gradient for ∇f_ξ with small probability regardless of the parameter choice in quantum gradient estimation. To address this, we propose a robust version of the quantum gradient estimation algorithm. Our final algorithm achieves $\tilde{O}(d^{1/2}\epsilon^{-1})$ query complexity to estimate the gradient up to ϵ error.

■ Algorithm 4 QuantumStochasticGradient.

-
- 0: **Input:** Access to stochastic functions $f_{\xi \in \Xi}$, variance σ^2 , target ϵ , smoothness parameter L , point $\mathbf{x}$.
Define $\beta = \frac{164L\sigma^2}{\epsilon^2}$, $D = \frac{40\sigma^2}{\epsilon}$, $\epsilon' = \frac{\epsilon^2}{\beta^2 d^3 (12000)^2}$.
 - 1: Sample ξ_0 at random from Ξ .
 - 2: Compute $\mathbf{s} = \text{QuantumGradient}(f_{\xi_0}, \epsilon', M, \beta, \mathbf{x})$ (See Algorithm 7).
 - 3: Let $\mathcal{A}$ be an algorithm that runs $\mathbf{g} = \text{QuantumGradient}(f_\xi, \epsilon', M, \beta, \mathbf{x})$ with random $\xi \in \Xi$ and outputs $\mathbf{g}$ if $\|\mathbf{g} - \mathbf{s}\| \leq D$, otherwise it outputs $\mathbf{s}$. Further suppose that $\mathcal{A}$ does not make any measurement.
 - 4: Run $\mathcal{A}$ on the superposition state to obtain $\sum_{\xi \in \Xi} \sqrt{\Pr(\xi)} |\mathcal{A}(\mathbf{x})\rangle |\xi\rangle$.
 - 5: Output $\mathbf{v} = \text{QuantumMeanEstimation}(\mathcal{A}, \epsilon/4, \delta)$.
-

The algorithm first samples ξ , runs the gradient estimation algorithm, and stores the output $\mathbf{s}$. The key idea is to use $\mathbf{s}$ to replace outlier estimates and then show that this replacement does not significantly increase the error with high probability. Next, we run $\mathcal{A}$ in superposition on $|\psi_1\rangle = \sum_{\xi \in \Xi} \sqrt{\Pr(\xi)} |\mathbf{x}\rangle |\xi\rangle$. The final step of Jordan's algorithm produces the following quantum state:

$$\sum_{\xi \in \Xi} \sqrt{\Pr(\xi)} |\tilde{\mathbf{g}}(\mathbf{x}; \xi)\rangle |\mathbf{x}\rangle |\xi\rangle + |\mathcal{X}_1\rangle, \quad (12)$$

where $|\mathcal{X}_1\rangle$ is another garbage state with a small amplitude and

$$\tilde{\mathbf{g}}(\mathbf{x}, \xi) = \begin{cases} \mathbf{g}(\mathbf{x}, \xi) & \text{if } \|\mathbf{g}(\mathbf{x}, \xi) - \mathbf{s}\| \leq D, \\ \mathbf{s} & \text{otherwise.} \end{cases} \quad (13)$$

is the corrected gradient estimate.

Finally, we estimate the mean of the first register and output the resulting vector $\mathbf{v}$ as the gradient estimate. The replacement step works because $\mathbf{s}$ is within a controlled distance of the true gradient with high probability, so truncating large-error estimates changes the expectation by only a small amount. This truncation is nevertheless necessary because rare large errors can otherwise dominate the mean. A rigorous probabilistic analysis gives the following guarantee.

► **Lemma 3.6.** *Under Assumptions 3.1 and 3.5, Algorithm 4 returns a vector $\mathbf{v}$ such that*

$$\|\mathbf{v} - \nabla f(\mathbf{x})\| \leq \epsilon \quad (14)$$

with high probability using $\tilde{O}(\sigma d^{1/2} \epsilon^{-1})$ queries to the stochastic evaluation oracle.

Next, we can postprocess the output of Algorithm 4 using MLMC to obtain a smooth and unbiased estimate.

► **Theorem 3.7** (Smooth Gradient). *Suppose that the potential function f satisfies Assumptions 3.1 and 3.5 and further suppose that $\|\nabla f(\mathbf{x})\| \leq M$ for all $\mathbf{x} \in \mathcal{B}(x^*, \text{poly}(d))$. Then, given a real $\mathbf{x} \in \mathcal{B}(x^*, \text{poly}(d))$ and $\hat{\sigma} > 0$, there exists a quantum algorithm that outputs a random vector $\mathbf{g}$ such that*

$$\mathbb{E}[\mathbf{g}] = \nabla f(\mathbf{x}), \quad \text{and} \quad \mathbb{E}\|\mathbf{g} - \nabla f(\mathbf{x})\|^2 \leq \hat{\sigma}^2 \quad (15)$$

using $\tilde{\mathcal{O}}(\frac{\sigma d^{1/2}}{\hat{\sigma}})$ queries to the stochastic evaluation oracle in expectation.

We remark that our focus is on query complexity. Quantum mean estimation and Jordan's algorithm use $\mathcal{O}(\text{poly}(d, \log(1/\epsilon)))$ gates [17, 9], and our algorithm does not change this circuit structure.

4 Quantum Speedups for Sampling via Evaluation Oracle

We apply the quantum gradient estimation algorithms from the previous sections to establish convergence of HMC and LMC with inexact gradients in the strongly convex and LSI settings. The gradient estimation error creates a tradeoff: larger error increases the number of sampler iterations, while smaller error requires more evaluation queries. We optimize this error to minimize the total query cost. We first treat approximate sampling from strongly log-concave distributions using HMC, with error measured in the Wasserstein metric.

► **Theorem 4.1** (Main Theorem for QZ-HMC). *Let μ_k be the distribution of $\mathbf{x}_k$ in QZ-HMC algorithm. Suppose that the Assumptions 2.5, 3.1, and 3.2 hold. Given that the initial point $\mathbf{x}_0$ satisfies $\|\mathbf{x}_0 - \arg \min_{\mathbf{x}} f(\mathbf{x})\| \leq \frac{d}{\mu}$, if we set the step size $\eta = \mathcal{O}(\frac{\epsilon}{d^{1/2}\kappa^{3/2}})$, $S = \tilde{\mathcal{O}}(\frac{Ld^{1/2}\kappa^{3/2}}{\epsilon})$, $T = \tilde{\mathcal{O}}(1)$, and $\hat{\sigma}^2 = \mathcal{O}(\frac{L^{3/2}d^{1/2}\epsilon}{\kappa^{3/2}})$, we have*

$$W_2(\mu_{ST}, \pi) \leq \epsilon.$$

In addition, under Assumptions 3.1 and 3.2, the query complexity to the stochastic evaluation oracle is $\tilde{\mathcal{O}}(\frac{d^{5/4}\sigma}{\epsilon^{3/2}})$ or under Assumptions 3.1 and 3.5 the query complexity to the stochastic evaluation oracle is $\tilde{\mathcal{O}}(\frac{d^{3/4}\sigma}{\epsilon^{3/2}})$.

Because strong convexity and the Wasserstein metric can be restrictive, we next consider non-convex potentials.

► **Theorem 4.2** (Main Theorem for QZ-LMC). *Suppose that the Assumptions 2.5, 3.1, and 3.2 hold. Let μ_k be the distribution of $\mathbf{x}_k$ in QZ-LMC algorithm. Then, if we set the step size $\eta = \mathcal{O}(\frac{\epsilon\alpha}{dL^2})$, $K = \tilde{\mathcal{O}}(\frac{dL^2 \log(\text{KL}(\mu_0||\pi))}{\epsilon\alpha^2})$, and $\hat{\sigma}^2 = \mathcal{O}(\alpha\epsilon)$, we have*

$$\left\{ \text{KL}(\mu_K||\pi), \text{TV}(\mu_K, \pi)^2, \frac{\alpha}{2} W_2(\mu_K, \pi)^2 \right\} \leq \epsilon.$$

In addition, under Assumptions 3.1 and 3.2, the query complexity to the stochastic evaluation oracle is $\tilde{\mathcal{O}}(\frac{d^2 L^2 \sigma}{\alpha^{5/2} \epsilon^{3/2}})$, or under Assumptions 3.1 and 3.5 the query complexity to the stochastic evaluation oracle is $\tilde{\mathcal{O}}(\frac{d^{3/2} L^2 \sigma}{\alpha^{5/2} \epsilon^{3/2}})$.

This result holds for several distance metrics. Compared with classical results, [39] analyzes zeroth-order LMC under Assumptions 3.1 and 3.2 and obtains evaluation complexity $\mathcal{O}(d^3\sigma^2/\epsilon^4)$ for convergence in W_2 (Theorem 3.2 of [39]). Our algorithm uses $\tilde{\mathcal{O}}_{\frac{d^2\sigma}{\epsilon^{3/2}}}$ evaluation queries under the same assumptions, yielding polynomial improvements in d , σ , and ϵ .

5 Application to Non-smooth and Non-convex Optimization

Although the previous sampling results assume smoothness (Lipschitz gradients), they also yield algorithms for non-smooth optimization. We consider objectives satisfying the following approximate-convexity and Lipschitz continuity assumptions.

► **Assumption 5.1** (Approximate Convexity). Let f be a differentiable function, we say that $f : \mathbb{R}^d \rightarrow \mathbb{R}$ is an ϵ -approximately convex function, if there exists a strongly convex function F such that for all $\mathbf{x}$,

$$|F(\mathbf{x}) - f(\mathbf{x})| \leq \frac{\epsilon}{d}. \quad (16)$$

► **Assumption 5.2** (Lipschitz Continuity). For all $\mathbf{x}, \mathbf{y} \in \mathbb{R}^d$, $f : \mathbb{R}^d \rightarrow \mathbb{R}$ satisfies,

$$|f(\mathbf{x}) - f(\mathbf{y})| \leq M\|\mathbf{x} - \mathbf{y}\|. \quad (17)$$

The goal is to find an approximate minimizer $\mathbf{x}^*$ such that $|f(\mathbf{x}^*) - \min_{\mathbf{x}} f(\mathbf{x})| \leq \epsilon$. This setting arises naturally in empirical risk minimization (ERM), where the objective is defined through empirical observations. Even if the population objective is smooth and strongly convex, subsampling can make the empirical objective non-smooth and non-convex. Similar settings also arise in the study of escaping local minima, both classically [5] and quantumly [29], under access to a stochastic evaluation oracle. Our approach differs from previous techniques by sampling from a Gibbs distribution associated with a smooth approximation of f .

Since f is not smooth, we consider the smoothed approximation

$$f_v(\mathbf{x}) = \mathbb{E}_{\mathbf{u} \sim \mathcal{B}_d(0,1)}[f(\mathbf{x} + v\mathbf{u})] \quad (18)$$

where $\mathcal{B}$ is the unit L_2 ball around the center. The local properties of f_v are known and given by the following proposition.

► **Proposition 5.3.** *If f satisfies Assumption 5.2, then f_v satisfies $|f_v(\cdot) - f(\cdot)| \leq vM$ and $|\nabla f_v(\mathbf{x}) - \nabla f_v(\mathbf{y})| \leq L\|\mathbf{x} - \mathbf{y}\|$, where $L = cM\sqrt{d}v^{-1}$ for some constant $c > 0$.*

First, note that $\mathbb{E}_{\mathbf{u}}\|\nabla f(\mathbf{x} + v\mathbf{u}) - \nabla f_v(\mathbf{x})\|^2 \leq 4M^2$, since $\|\nabla f(\mathbf{x})\| \leq M$ by Lipschitz continuity.

The essential idea is that the Gibbs distribution localizes around the global minimizers of f as β increases. Another challenge in sampling from f_v is that the log-Sobolev constant may depend on the dimension. Using the Holley-Stroock LSI perturbation result [25], we show how this dimension dependence enters the total evaluation complexity. We now state the main result.

► **Theorem 5.4.** *Suppose that f satisfies Assumptions 5.1 and 5.2. Then, there exists a quantum algorithm that returns ϵ approximate minimizer for f with high probability using $\tilde{\mathcal{O}}\left(\frac{d^{9/2}}{\epsilon^{3/2}}\right)$ queries to the stochastic evaluation oracle for f .*

The closest result to our setting is [29], whose stochastic query complexity is $\tilde{O}(d^5/\epsilon)$, although the assumptions are not directly comparable. The authors assume additive sub-Gaussian noise and convexity of F on a bounded domain, but not strong convexity. Because these assumptions differ from ours, the comparison should be interpreted cautiously. Under our assumptions, the algorithm improves the dimension dependence at the cost of worse ϵ -dependence, a common tradeoff for sampling-based optimization methods.

We also note the work by Augustino et al. [2] and Chakrabarti et al. [9], which also investigate zeroth-order stochastic convex optimization under assumptions similar to those in [29]. They propose algorithms with query complexities $\tilde{O}(d^{9/2}/\epsilon^7)$ and $\tilde{O}(d^3/\epsilon^5)$, respectively. Although both approaches have worse ϵ -dependence than ours, their assumptions and problem settings differ significantly from ours.

References

- 1 Simon Apers and Alain Sarlette. Quantum fast-forwarding: Markov chains and graph property testing. *Quantum Info. Comput.*, 19(3–4):181–213, March 2019. doi:10.26421/QIC19.3–4–1.
- 2 Brandon Augustino, Dylan Herman, Enrico Fontana, Junhyung Lyle Kim, Jacob Watkins, Shouvanik Chakrabarti, and Marco Pistoia. Fast convex optimization with quantum gradient methods, 2025. doi:10.48550/arXiv.2503.17356.
- 3 Jack Baker, Paul Fearnhead, Emily B. Fox, and Christopher Nemeth. Control variates for stochastic gradient MCMC. *Statistics and Computing*, 29(3):599–615, May 2019. doi:10.1007/s11222-018-9826-2.
- 4 Krishnakumar Balasubramanian and Saeed Ghadimi. Zeroth-order nonconvex stochastic optimization: Handling constraints, high dimensionality, and saddle points. *Found. Comput. Math.*, 22(1):35–76, February 2022. doi:10.1007/s10208-021-09499-8.
- 5 Alexandre Belloni, Tengyuan Liang, Hariharan Narayanan, and Alexander Rakhlin. Escaping the local minima via simulated annealing: Optimization of approximately convex functions. In Peter Grünwald, Elad Hazan, and Satyen Kale, editors, *Proceedings of The 28th Conference on Learning Theory*, volume 40 of *Proceedings of Machine Learning Research*, pages 240–265, Paris, France, 03–06 July 2015. PMLR. URL: <https://proceedings.mlr.press/v40/Belloni15.html>.
- 6 Shouvanik Chakrabarti, Andrew M. Childs, Shih-Han Hung, Tongyang Li, Chunhao Wang, and Xiaodi Wu. Quantum algorithm for estimating volumes of convex bodies. *ACM Transactions on Quantum Computing*, 4(3), May 2023. doi:10.1145/3588579.
- 7 Shouvanik Chakrabarti, Andrew M. Childs, Tongyang Li, and Xiaodi Wu. Quantum algorithms and lower bounds for convex optimization. *Quantum*, 4:221, January 2020. doi:10.22331/q-2020-01-13-221.
- 8 Shouvanik Chakrabarti, Dylan Herman, Guneykan Ozgul, Shuchen Zhu, Brandon Augustino, Tianyi Hao, Zichang He, Ruslan Shaydulin, and Marco Pistoia. Generalized short path algorithms: Towards super-quadratic speedup over Markov chain search for combinatorial optimization, 2024. doi:10.48550/arXiv.2410.23270.
- 9 Shouvanik Chakrabarti, Dylan Herman, Jacob Watkins, Enrico Fontana, Brandon Augustino, Junhyung Lyle Kim, and Marco Pistoia. On speedups for convex optimization via quantum dynamics, 2025. doi:10.48550/arXiv.2503.24332.
- 10 David Chandler. Introduction to modern statistical. *Mechanics. Oxford University Press, Oxford, UK*, 5(449):11, 1987.
- 11 Niladri Chatterji, Nicolas Flammarion, Yian Ma, Peter Bartlett, and Michael Jordan. On the theory of variance reduction for stochastic gradient Monte Carlo. In Jennifer Dy and Andreas Krause, editors, *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 764–773. PMLR, 10–15 July 2018. URL: <https://proceedings.mlr.press/v80/chatterji18a.html>.

- 12 Tianqi Chen, Emily Fox, and Carlos Guestrin. Stochastic gradient Hamiltonian Monte Carlo. In Eric P. Xing and Tony Jebara, editors, *Proceedings of the 31st International Conference on Machine Learning*, volume 32 of *Proceedings of Machine Learning Research*, pages 1683–1691, Beijing, China, 22–24 June 2014. PMLR. URL: <https://proceedings.mlr.press/v32/cheni14.html>.
- 13 Xi Chen, Simon S. Du, and Xin T. Tong. On stationary-point hitting time and ergodicity of stochastic gradient langevin dynamics. *Journal of Machine Learning Research*, 21(68):1–41, 2020. URL: <http://jmlr.org/papers/v21/19-327.html>.
- 14 Sinho Chewi, Murat A Erdogdu, Mufan Li, Ruoqi Shen, and Shunshi Zhang. Analysis of Langevin Monte carlo from Poincare to log-Sobolev. In Po-Ling Loh and Maxim Raginsky, editors, *Proceedings of Thirty Fifth Conference on Learning Theory*, volume 178 of *Proceedings of Machine Learning Research*, pages 1–2. PMLR, 02–05 July 2022. URL: <https://proceedings.mlr.press/v178/chewi22a.html>.
- 15 Sinho Chewi and Austin J. Stromme. The ballistic limit of the log-Sobolev constant equals the Polyak-Łojasiewicz constant, 2024. doi:10.48550/arXiv.2411.11415.
- 16 Andrew M. Childs, Tongyang Li, Jin-Peng Liu, Chunhao Wang, and Ruizhe Zhang. Quantum algorithms for sampling log-concave distributions and estimating normalizing constants. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh, editors, *Advances in Neural Information Processing Systems*, volume 35, pages 23205–23217. Curran Associates, Inc., 2022. URL: https://proceedings.neurips.cc/paper_files/paper/2022/file/933e953353c25ec70477ef28e45a2dcc-Paper-Conference.pdf.
- 17 Arjan Cornelissen, Yassine Hamoudi, and Sofiene Jerbi. Near-optimal quantum algorithms for multivariate mean estimation. In *Proceedings of the 54th Annual ACM SIGACT Symposium on Theory of Computing*, STOC ’22. ACM, June 2022. doi:10.1145/3519935.3520045.
- 18 Ben Cousins and Santosh Vempala. Gaussian cooling and $O^*(n^3)$ algorithms for volume and Gaussian volume. *SIAM Journal on Computing*, 47(3):1237–1273, 2018. doi:10.1137/15M1054250.
- 19 Aniket Das, Dheeraj M. Nagaraj, and Anant Raj. Utilising the CLT structure in stochastic gradient based sampling : Improved analysis and faster algorithms. In Gergely Neu and Lorenzo Rosasco, editors, *Proceedings of Thirty Sixth Conference on Learning Theory*, volume 195 of *Proceedings of Machine Learning Research*, pages 4072–4129. PMLR, 12–15 July 2023. URL: <https://proceedings.mlr.press/v195/das23b.html>.
- 20 Aaron Defazio, Francis Bach, and Simon Lacoste-Julien. SAGA: A fast incremental gradient method with support for non-strongly convex composite objectives. In Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, and K.Q. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 27. Curran Associates, Inc., 2014. URL: https://proceedings.neurips.cc/paper_files/paper/2014/file/ede7e2b6d13a41ddf9f4bdef84fdc737-Paper.pdf.
- 21 Kumar Avinava Dubey, Sashank J. Reddi, Sinead A Williamson, Barnabas Poczos, Alexander J Smola, and Eric P Xing. Variance reduction in stochastic gradient langevin dynamics. In D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 29. Curran Associates, Inc., 2016. URL: https://proceedings.neurips.cc/paper_files/paper/2016/file/9b698eb3105bd82528f23d0c92dedfc0-Paper.pdf.
- 22 Alain Durmus and Eric Moulines. High-dimensional Bayesian inference via the unadjusted Langevin algorithm, 2018. doi:10.48550/arXiv.1605.01559.
- 23 Daan Frenkel and Berend Smit. *Understanding Molecular Simulation: From Algorithms to Applications*, volume 1 of *Computational Science Series*. Academic Press, San Diego, second edition, 2002.
- 24 András Gilyén, Srinivasan Arunachalam, and Nathan Wiebe. *Optimizing quantum optimization algorithms via faster quantum gradient computation*, pages 1425–1444. Society for Industrial and Applied Mathematics, January 2019. doi:10.1137/1.9781611975482.87.

- 25 Richard Holley and Daniel W. Stroock. Logarithmic sobolev inequalities and stochastic ising models. *Journal of Statistical Physics*, 46:1159–1194, 1987. URL: <https://link.springer.com/article/10.1007/BF01011161>.
- 26 Rie Johnson and Tong Zhang. Accelerating stochastic gradient descent using predictive variance reduction. In C.J. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K.Q. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 26. Curran Associates, Inc., 2013. URL: https://proceedings.neurips.cc/paper_files/paper/2013/file/ac1dd209cbcc5e5d1c6e28598e8cbb8-Paper.pdf.
- 27 Stephen P. Jordan. Fast quantum algorithm for numerical gradient estimation. *Physical Review Letters*, 95(5), July 2005. doi:10.1103/physrevlett.95.050501.
- 28 Yuri Kinoshita and Taiji Suzuki. Improved convergence rate of stochastic gradient Langevin dynamics with variance reduction and its application to optimization. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh, editors, *Advances in Neural Information Processing Systems*, volume 35, pages 19022–19034. Curran Associates, Inc., 2022. URL: https://proceedings.neurips.cc/paper_files/paper/2022/file/78e839f96568985d18463044a064ea0f-Paper-Conference.pdf.
- 29 Tongyang Li and Ruizhe Zhang. Quantum speedups of optimizing approximately convex functions with applications to logarithmic regret stochastic convex bandits. In *Proceedings of the 36th International Conference on Neural Information Processing Systems, NIPS '22*, Red Hook, NY, USA, 2024. Curran Associates Inc.
- 30 Chengchang Liu, Chaowen Guan, Jianhao He, and John C.S. Lui. Quantum algorithms for non-smooth non-convex optimization. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024.
- 31 László Lovász and Santosh Vempala. Simulated annealing in convex bodies and an $O^*(n^4)$ volume algorithm. *Journal of Computer and System Sciences*, 72(2):392–417, 2006. JCSS FOCS 2003 Special Issue. doi:10.1016/j.jcss.2005.08.004.
- 32 Yi-An Ma, Yuansi Chen, Chi Jin, Nicolas Flammarion, and Michael I. Jordan. Sampling can be faster than optimization. *Proceedings of the National Academy of Sciences*, 116(42):20881–20885, September 2019. doi:10.1073/pnas.1820003116.
- 33 Frederic Magniez, Ashwin Nayak, Jeremie Roland, and Miklos Santha. Search via quantum walk. In *Proceedings of the Thirty-Ninth Annual ACM Symposium on Theory of Computing, STOC '07*, pages 575–584, New York, NY, USA, 2007. Association for Computing Machinery. doi:10.1145/1250790.1250874.
- 34 Yurii Nesterov and Vladimir Spokoiny. Random gradient-free minimization of convex functions. *Found. Comput. Math.*, 17(2):527–566, April 2017. doi:10.1007/s10208-015-9296-2.
- 35 Lam M. Nguyen, Jie Liu, Katya Scheinberg, and Martin Takáč. SARAH: A novel method for machine learning problems using stochastic recursive gradient. In Doina Precup and Yee Whye Teh, editors, *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pages 2613–2621. PMLR, 06–11 August 2017. URL: <https://proceedings.mlr.press/v70/nguyen17b.html>.
- 36 Guneykan Ozgul, Xiantao Li, Mehrdad Mahdavi, and Chunhao Wang. Stochastic quantum sampling for non-logconcave distributions and estimating partition functions. In Ruslan Salakhutdinov, Zico Kolter, Katherine Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp, editors, *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pages 38953–38982. PMLR, 21–27 July 2024. URL: <https://proceedings.mlr.press/v235/ozgul24a.html>.
- 37 Guneykan Ozgul, Xiantao Li, Mehrdad Mahdavi, and Chunhao Wang. Quantum speedups for sampling and non-convex optimization with stochastic oracles, 2025. doi:10.48550/arXiv.2504.03626.
- 38 Maxim Raginsky, Alexander Rakhlin, and Matus Telgarsky. Non-convex learning via stochastic gradient Langevin dynamics: a nonasymptotic analysis. In Satyen Kale and Ohad Shamir, editors, *Proceedings of the 2017 Conference on Learning Theory*, volume 65 of *Proceedings of Machine Learning Research*, pages 1674–1703. PMLR, 07–10 July 2017. URL: <https://proceedings.mlr.press/v65/raginsky17a.html>.

- 39 Abhishek Roy, Lingqing Shen, Krishnakumar Balasubramanian, and Saeed Ghadimi. Stochastic zeroth-order discretizations of Langevin diffusions for Bayesian inference, 2021. doi:10.48550/arXiv.1902.01373.
- 40 Aaron Sidford and Chenyi Zhang. Quantum speedups for stochastic optimization. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine, editors, *Advances in Neural Information Processing Systems*, volume 36, pages 35300–35330. Curran Associates, Inc., 2023. URL: https://proceedings.neurips.cc/paper_files/paper/2023/file/6ed9931d6e1fb6a85efa1b2c014a47e1-Paper-Conference.pdf.
- 41 R. Somma, S. Boixo, and H. Barnum. Quantum simulated annealing, 2007. doi:10.48550/arXiv.0712.1008.
- 42 R. D. Somma, S. Boixo, H. Barnum, and E. Knill. Quantum simulations of classical annealing processes. *Phys. Rev. Lett.*, 101:130504, September 2008. doi:10.1103/PhysRevLett.101.130504.
- 43 M. Szegedy. Quantum speed-up of Markov chain based algorithms. In *45th Annual IEEE Symposium on Foundations of Computer Science*, pages 32–41, 2004. doi:10.1109/FOCS.2004.53.
- 44 Joran van Apeldoorn, András Gilyén, Sander Gribling, and Ronald de Wolf. Convex optimization using quantum oracles. *Quantum*, 4:220, January 2020. doi:10.22331/q-2020-01-13-220.
- 45 Santosh Vempala and Andre Wibisono. Rapid convergence of the unadjusted Langevin algorithm: Isoperimetry suffices. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019. URL: https://proceedings.neurips.cc/paper_files/paper/2019/file/65a99bb7a3115fdede20da98b08a370f-Paper.pdf.
- 46 Yu-Xiang Wang, Stephen Fienberg, and Alex Smola. Privacy for free: Posterior sampling and stochastic gradient Monte Carlo. In Francis Bach and David Blei, editors, *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *Proceedings of Machine Learning Research*, pages 2493–2502, Lille, France, 07–09 July 2015. PMLR. URL: <https://proceedings.mlr.press/v37/wangg15.html>.
- 47 Max Welling and Yee Whye Teh. Bayesian learning via stochastic gradient Langevin dynamics. In *Proceedings of the 28th International Conference on International Conference on Machine Learning*, ICML'11, pages 681–688, Madison, WI, USA, 2011. Omnipress. URL: https://icml.cc/2011/papers/398_icmlpaper.pdf.
- 48 Pawel Wocjan and Anura Abeyesinghe. Speedup via quantum sampling. *Phys. Rev. A*, 78:042336, October 2008. doi:10.1103/PhysRevA.78.042336.
- 49 Pan Xu, Jinghui Chen, Difan Zou, and Quanquan Gu. Global convergence of Langevin dynamics based algorithms for nonconvex optimization. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc., 2018. URL: https://proceedings.neurips.cc/paper_files/paper/2018/file/9c19a2aa1d84e04b0bd4bc888792bd1e-Paper.pdf.
- 50 Yexin Zhang, Chenyi Zhang, Cong Fang, Liwei Wang, and Tongyang Li. Quantum algorithms and lower bounds for finite-sum optimization. In Ruslan Salakhutdinov, Zico Kolter, Katherine Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp, editors, *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pages 60244–60270. PMLR, 21–27 July 2024. URL: <https://proceedings.mlr.press/v235/zhang24bz.html>.
- 51 Yuchen Zhang, Percy Liang, and Moses Charikar. A hitting time analysis of stochastic gradient Langevin dynamics. In Satyen Kale and Ohad Shamir, editors, *Proceedings of the 2017 Conference on Learning Theory*, volume 65 of *Proceedings of Machine Learning Research*, pages 1980–2022. PMLR, 07–10 July 2017. URL: <https://proceedings.mlr.press/v65/zhang17b.html>.

- 52 Difan Zou and Quanquan Gu. On the convergence of Hamiltonian Monte Carlo with stochastic gradients. In Marina Meila and Tong Zhang, editors, *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 13012–13022. PMLR, 18–24 July 2021. URL: <https://proceedings.mlr.press/v139/zou21b.html>.
- 53 Difan Zou, Pan Xu, and Quanquan Gu. Stochastic gradient Hamiltonian Monte Carlo methods with recursive variance reduction. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019. URL: https://proceedings.neurips.cc/paper_files/paper/2019/file/c3535febaff29fcb7c0d20cbe94391c7-Paper.pdf.
- 54 Difan Zou, Pan Xu, and Quanquan Gu. Faster convergence of stochastic gradient Langevin dynamics for non-log-concave sampling. In Cassio de Campos and Marloes H. Maathuis, editors, *Proceedings of the Thirty-Seventh Conference on Uncertainty in Artificial Intelligence*, volume 161 of *Proceedings of Machine Learning Research*, pages 1152–1162. PMLR, 27–30 July 2021. URL: <https://proceedings.mlr.press/v161/zou21a.html>.

A Overview of Classical Sampling Algorithms

One widely used method for sampling from the Gibbs-Boltzmann distribution is to use the Langevin Monte Carlo (LMC) algorithm:

$$\mathbf{x}_{t+1} = \mathbf{x}_t - \eta_t \nabla f(\mathbf{x}_t) + \sqrt{2\eta_t} \epsilon_t, \quad (19)$$

where $\eta_t > 0$ is the step size and ϵ_t is isotropic Gaussian noise. This update rule is reminiscent of the gradient descent update rule, but there is an additional noise term that makes the algorithm stochastic. LMC results from the *Euler-Maruyama* discretization of a continuous-time Langevin diffusion equation:

$$d\mathbf{x}_t = -\nabla f(\mathbf{x}_t)dt + \sqrt{2}d\mathbf{B}(t), \quad (20)$$

where $\mathbf{B}(t)$ is the standard Brownian motion. It is not hard to show that the distribution π is actually stationary for continuous dynamics. Therefore, after iterating the LMC algorithm sufficiently long times with a small step size, the iterates converge to the Gibbs-Boltzmann distribution. We also note that this algorithm is referred to as the Unadjusted Langevin Algorithm (ULA). Therefore, we sometimes use ULA and LMC interchangeably. Although effective, the computational cost of each iteration in this algorithm becomes prohibitive when the computation of the gradient is costly (such as in the finite-sum setting) or not directly available (such as in the zeroth-order setting, where one needs to compute the gradient using a finite difference formula). To alleviate the computational burden, stochastic gradient-based samplers such as Stochastic Gradient Langevin Dynamics (SGLD) [47] have been proposed. Instead of computing the full gradient, these algorithms use stochastic approximation to the gradient. In the stochastic gradient oracle setting, a stochastic estimate $\mathbf{g}_t$ can be obtained by randomly sampling a component $i \in [n]$ and computing $\nabla f_i(\mathbf{x}_t)$. In the stochastic zeroth-order setting, the stochastic gradient can also be obtained by using finite difference formulas by evaluating the function at two close points [34].

Another method that is commonly used in sampling is the Hamiltonian Monte Carlo (HMC) algorithm, which uses the principles of Hamiltonian dynamics to propose new states in a Markov Chain. It introduces the Hamiltonian $H(\mathbf{x}, \mathbf{p}) = f(\mathbf{x}) + \frac{1}{2}\|\mathbf{p}\|^2$ with auxiliary momentum variables and updates the position ($\mathbf{x}$) and momentum ($\mathbf{p}$) by simulating Hamiltonian dynamics, which follows the equations:

$$\frac{d\mathbf{x}}{dt} = \frac{\partial H}{\partial \mathbf{p}}, \quad \frac{d\mathbf{p}}{dt} = -\frac{\partial H}{\partial \mathbf{x}}. \quad (21)$$

Algorithm 5 SG-LMC.

input The stochastic gradient oracle $O_{\nabla f}$, initial point $\mathbf{x}_0$, step size η , number of steps K
output Approximate sample from $\pi \propto e^{-f(\mathbf{x})}$
for $k = 0$ to K **do**
 Sample $\epsilon_t \sim \mathcal{N}(0, I)$
 $\mathbf{x}_{k+1} = \mathbf{x}_k - \eta_t k \mathbf{g}(\mathbf{x}_k, \xi_k) + \sqrt{2\eta_k} \epsilon_k$,
end for
Return $\mathbf{x}^K$

After a series of updates, the momentum $\mathbf{p}_{k+1}$ is refreshed by sampling from $\mathcal{N}(0, I)$. This discretization ensures symplecticity, preserving volume in phase space and allowing the algorithm to make large, energy-conserving moves through the parameter space. Similar to SGLD, Stochastic Hamiltonian Monte Carlo (SG-HMC) [12] has been proposed, where one can replace the gradients with stochastic gradients (See Algorithm 6).

Algorithm 6 SG-HMC.

input The stochastic gradient oracle $O_{\nabla f}$, initial point $\mathbf{x}_0$, step size η , number of leapfrog steps S , number of HMC proposals T
output Approximate sample from $\pi \propto e^{-f(\mathbf{x})}$
for $t = 0$ to T **do**
 Sample $\mathbf{p}_{St} \sim \mathcal{N}(0, I)$
 for $s = 0$ to $S - 1$ **do**
 $k = St + s$
 $\mathbf{x}_{k+1} = \mathbf{x}_k + \eta \mathbf{p}_k - \frac{\eta^2}{2} \mathbf{g}(\mathbf{x}_k, \xi_k)$
 $\mathbf{p}_{k+1} = \mathbf{p}_k - \frac{\eta}{2} \mathbf{g}(\mathbf{x}_k, \xi_k) - \frac{\eta}{2} \mathbf{g}(\mathbf{x}_{k+1}, \xi_{k+1/2})$
 end for
end for
Return $\mathbf{x}^T$

The stochastic gradients $\mathbf{g}(\mathbf{x}, \xi)$ in Algorithm 6 can be obtained using different techniques. While stochastic gradient methods reduce computation at each iteration, they introduce variance into the gradient estimates, which can degrade the quality of the samples and slow down convergence. To reduce the variance, further techniques have been proposed, such as SVRG (stochastic variance reduced gradients), CV (control variates). In this paper, we will also employ quantum computational techniques to further reduce the variance in the gradients.

B Overview of Quantum Algorithms

B.1 Quantum Mean Estimation

Quantum mean estimation is a technique to estimate the mean of a d -dimensional random variable X up to ϵ accuracy using $\tilde{\mathcal{O}}(d^{1/2}/\epsilon)$ queries, which is a quadratic improvement in ϵ compared to classical algorithms [17]. Although the quantum mean estimation algorithm is biased, [40] developed an unbiased quantum mean estimation algorithm. Specifically, for a multi-dimensional variable with mean μ and variance σ^2 , unbiased quantum mean estimation outputs an estimate $\hat{\mu}$ such that $\mathbb{E}[\hat{\mu}] = \mu$ and $\mathbb{E}[\|\hat{\mu} - \mu\|^2] \leq \hat{\sigma}^2$ using $\tilde{\mathcal{O}}(d^{1/2}\sigma/\hat{\sigma})$ queries.

► **Lemma B.1** (Unbiased quantum mean estimation [40]). *Let X be a d -dimensional random variable with $\text{Var}[X] \leq \sigma^2$, and suppose a quantum sampling oracle for X is available. For any target variance $\hat{\sigma}^2$, there is a procedure that outputs an unbiased estimate of $\mathbb{E}X$ with variance at most $\hat{\sigma}^2$ using $\tilde{O}(d^{1/2}\sigma/\hat{\sigma})$ oracle queries.*

B.2 Jordan's Gradient Estimation Algorithm

Jordan's algorithm [27] approximates the gradient using a finite difference formula on a small grid around the point of interest and encodes the estimate into the quantum phase. Then, the algorithm applies an inverse quantum Fourier transform to estimate the gradient. Although Jordan's original analysis implicitly assumes that higher-order derivatives of the function are negligible, Gilyén, Arunachalam, and Wiebe [24] analyzed the algorithm and extended it to handle functions in the Gevrey class, using central difference formulas and a binary oracle model commonly encountered in variational quantum algorithms. The closest analysis of Jordan's algorithm to our setting was provided by [7], who demonstrated that Algorithm 7 achieves constant query complexity for functions with Lipschitz gradients, provided that the function values can be queried with high precision.

■ **Algorithm 7** $\text{QuantumGradient}(f, \epsilon, L, \beta, x_0)$.

0: **Input:** Function f , evaluation error ϵ , gradient norm bound L , smoothness parameter β , and point $\mathbf{x}_0$.

Define

- $l = 2\sqrt{\epsilon/\beta d}$ to be the size of the grid used,
- $b \in \mathbb{N}$ such that $\frac{24\pi\sqrt{d\epsilon\beta}}{L} \leq \frac{1}{2^b} = \frac{1}{N} \leq \frac{48\pi\sqrt{d\epsilon\beta}}{L}$,
- $b_0 \in \mathbb{N}$ such that $\frac{N\epsilon}{2Ll} \leq \frac{1}{2^{b_0}} = \frac{1}{N_0} \leq \frac{N\epsilon}{Ll}$,
- $F(x) = \frac{N}{2Ll}[f(x_0 + \frac{l}{N}(x - N/2)) - f(x_0)]$, and,
- $\gamma: \{0, 1, \dots, N-1\} \rightarrow G := \{-N/2, -N/2+1, \dots, N/2-1\}$ s.t. $\gamma(x) = x - N/2$.

Let O_F denote a unitary operation acting as $O_F|x\rangle = e^{2\pi i \tilde{F}(x)}|x\rangle$, where $|\tilde{F}(x) - F(x)| \leq \frac{1}{N_0}$, with x represented using b bits and $\tilde{F}(x)$ represented using b_0 bits.

1: Start with n b -bit registers set to 0 and Hadamard transform each to obtain

$$\frac{1}{\sqrt{N^n}} \sum_{x_1, \dots, x_n \in \{0, 1, \dots, N-1\}} |x_1, \dots, x_n\rangle;$$

2: Perform the operation O_F and the map $|x\rangle \mapsto |\gamma(x)\rangle$ to obtain

$$\frac{1}{N^{n/2}} \sum_{\mathbf{g} \in G^n} e^{2\pi i \tilde{F}(\mathbf{g})} |\mathbf{g}\rangle;$$

3: Apply the inverse QFT over G to each of the registers

4: Measure the final state to get $k_1, k_2, \dots, k_n$ and report $\tilde{\mathbf{g}} = \frac{2L}{N}(k_1, k_2, \dots, k_n)$ as the result.

The following lemma from [7] demonstrates that Algorithm 7 achieves $\tilde{O}(1)$ query complexity for evaluating the gradient of a β -smooth function with high probability.

► **Lemma B.2** (Lemma 2.2 in [7]). *Let $f: \mathbb{R}^d \rightarrow \mathbb{R}$ be a function that is accessible via an evaluation oracle with error at most ϵ . Assume that $\|\nabla f\| \leq L$ and f is β -smooth in $B_\infty(x, 2\sqrt{\epsilon/\beta})$. Let $\tilde{\mathbf{g}}$ be the output of $\text{QuantumGradient}(f, \epsilon, M, \beta, \mathbf{x}_0)$ (as defined in Algorithm 7). Then:*

$$\Pr \left[|\tilde{\mathbf{g}}_i - \nabla f(\mathbf{x})_i| > 1500\sqrt{d\epsilon\beta} \right] < \frac{1}{3}, \quad \forall i \in [d]. \quad (22)$$

Although Algorithm 7 results in an accurate estimate for the gradient with high probability, it is possible to run the algorithm multiple times and take the coordinate-wise median of the outputs to obtain a smooth estimate for the gradient (Lemma 2.3 in [7]) when the norm of the gradient is bounded. To estimate the gradient up to δ error (in $L2$ norm), it is required to have an evaluation oracle with error at most $\mathcal{O}(\delta^2/d^2)$, which might not be feasible if the noisy evaluation oracle is stochastic.

Our algorithms work in the stochastic setting where we prove that we can create an accurate evaluation oracle under Assumptions 3.1 and 3.2. Furthermore, the function f needs to be smooth; however, under Assumptions 3.1 and 3.5, the smoothness constant is not bounded, and this might cause unbounded error. We propose a robust version of Algorithm 7 so that we can still estimate the gradient accurately (See the step-by-step description in Section 3.2). We also note that the oracle O_F is known as the phase oracle. Our oracle (Equation (4)) can be converted to a phase oracle efficiently.

Second, Jordan’s quantum gradient estimation algorithm [27, 24, 7] estimates all coordinates of a gradient by querying a phase oracle on a grid around the target point and applying a quantum Fourier transform. In the deterministic setting, if f is sufficiently smooth and the phase oracle is accurate enough, the algorithm estimates $\nabla f(\mathbf{x})$ with polylogarithmic overhead. The stochastic zeroth-order setting requires additional work because the phase oracle must represent $\mathbb{E}_\xi[f(\mathbf{x}; \xi)]$ even when the random function value is not bounded.

More concretely, Jordan’s algorithm prepares a uniform superposition over a grid $G_d^l(\mathbf{x})$ centered at $\mathbf{x}$, applies phases proportional to finite differences of f on this grid, and then applies an inverse quantum Fourier transform. If the function is nearly linear on the grid, the resulting frequency encodes the gradient. The deterministic analyses show that Lipschitz-gradient functions can be handled by choosing the grid width and evaluation precision appropriately. In our stochastic setting, the same high-level structure is used, but the phase query itself must be synthesized from noisy evaluations. This is the role of the robust phase oracle in Proposition 3.3.

B.3 Overview of Multi-Level Monte Carlo Algorithm

In this section, we give a brief overview of a technique known as the Multi-Level Monte Carlo algorithm. Without using this technique, our gradient estimation algorithms in the zeroth-order setting would not provide an unbiased estimate for the gradient. Suppose that we have an algorithm `BiasedStochasticGradient`($\mathbf{x}, \sigma$) that outputs $\mathbf{v}$ such that $\mathbb{E}\|\mathbf{v} - \nabla f(\mathbf{x})\| \leq \hat{\sigma}^2$ with cost $\tilde{\mathcal{O}}\left(\frac{C}{\hat{\sigma}}\right)$ where C is a function of other problem parameters. Consider the following algorithm.

While stochastic gradient methods reduce computation at each iteration, they introduce variance into the gradient estimates, which can degrade the quality of the samples and slow down convergence.

■ **Algorithm 8** `UnbiasedStochasticGradient`.

- 0: **Input:** Estimator `BiasedStochasticGradient`, target variance $\hat{\sigma}^2$
Output: An unbiased estimate $\mathbf{g}$ of $\nabla f(\mathbf{x})$ with variance at most $\hat{\sigma}^2$
- 1: Set $\mathbf{g}_0 \leftarrow \text{BiasedStochasticGradient}(\mathbf{x}, \hat{\sigma}/10)$
 - 2: Randomly sample $j \sim \text{Geom}\left(\frac{1}{2}\right) \in \mathbb{N}$
 - 3: $\mathbf{g}_j \leftarrow \text{BiasedStochasticGradient}(\mathbf{x}, 2^{-3j/4}\hat{\sigma}/10)$
 - 4: $\mathbf{g}_{j-1} \leftarrow \text{BiasedStochasticGradient}(\mathbf{x}, 2^{-3(j-1)/4}\hat{\sigma}/10)$
 - 5: $\mathbf{g} \leftarrow \mathbf{g}_0 + 2^j(\mathbf{g}_j - \mathbf{g}_{j-1})$
 - 5: Return $\mathbf{g}$
-

► **Lemma B.3.** *Given access to an algorithm `BiasedStochasticGradient` that outputs a random vector $\mathbf{v}$ such that $\mathbb{E}\|\mathbf{v} - \nabla f(\mathbf{x})\| \leq \hat{\sigma}^2$ with a cost $\tilde{O}\left(\frac{C}{\hat{\sigma}}\right)$, Algorithm 8 outputs a vector $\mathbf{g}$ such that $\mathbb{E}[\mathbf{g}] = \nabla f(\mathbf{x})$ and $\mathbb{E}\|\mathbf{g} - \nabla f(\mathbf{x})\| \leq \hat{\sigma}^2$ with an expected cost $\tilde{O}\left(\frac{C}{\hat{\sigma}}\right)$.*

Disclaimer

This paper is not a product of the Research Department of JPMorgan Chase & Co. or its affiliates. Neither JPMorgan Chase & Co. nor any of its affiliates makes any explicit or implied representation or warranty and none of them accept any liability in connection with this paper, including, without limitation, with respect to the completeness, accuracy, or reliability of the information contained herein and the potential legal, compliance, tax, or accounting effects thereof. This document is not intended as investment research or investment advice, or as a recommendation, offer, or solicitation for the purchase or sale of any security, financial instrument, financial product or service, or to be used in any way for evaluating the merits of participating in any transaction.

