16th Conference on the Theory of Quantum Computation, Communication and Cryptography

TQC 2021, July 5–8, 2021, Virtual Conference

Edited by

Min-Hsiu Hsieh

LIPIcs – Vol. 197 – TQC 2021 www.dagstuhl.de/lipics

Editors

Min-Hsiu Hsieh
Hon Hai (Foxconn) Quantum Computing Research Center, Taipei, Taiwan
minhsiuh@gmail.com

ACM Classification 2012
Theory of computation → Quantum computation theory; Theory of computation → Quantum complexity theory; Theory of computation → Quantum information theory; Theory of computation → Quantum communication complexity; Hardware → Quantum communication and cryptography; Hardware → Quantum error correction and fault tolerance

ISBN 978-3-95977-198-6

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/978-3-95977-198-6.

Publication date
June, 2021

Bibliographic information published by the Deutsche Nationalbibliothek
The Deutsche Nationalbibliothek lists this publication in the Deutsche Nationalbibliografie; detailed bibliographic data are available in the Internet at https://portal.dnb.de.

Digital Object Identifier: 10.4230/LIPIcs.TQC.2021.0

ISBN 978-3-95977-198-6 **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.

ISSN 1868-8969

https://www.dagstuhl.de/lipics

Contents

Regular Papers

16th Conference on the Theory of Quantum Computation, Communication and Cryptography (TQC 2021).
Editor: Min-Hsiu Hsieh
LIPICS Leibniz International Proceedings in Informatics
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

Preface

The 16th Conference on the Theory of Quantum Computation, Communication and Cryptography was hosted by the University of Latvia, and held online from July 5–8, 2021.

Quantum computation, quantum communication, and quantum cryptography are subfields of quantum information processing, an interdisciplinary field of information science and quantum mechanics. The TQC conference series focuses on theoretical aspects of these subfields. The objective of the conference is to bring together researchers so that they can interact with each other and share problems and recent discoveries.

A list of the previous editions of TQC follows:

- TQC 2020, University of Latvia, Latvia
- 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, a poster session, and a business meeting. The invited talks were given by Scott Aaronson (UT Austin), Srinivasan Arunachalam (IBM T. J. Watson Research Center), Cécilia Lancien (Institut de Mathématiques de Toulouse and CNRS), and Kai-Min Chung (Academia Sinica).

Conference Organization

Local Organizing Committee

- Andris Ambainis (chair)
  Latvia
- Kaspars Čikste
  Latvia
- Jelena Glušakova
  Latvia
- Juris Smotrovs
  Latvia
- Aleksandrs Rivoss
  Latvia
- Dace Šostaka
  Latvia
- Jevgēnijs Vihrovs
  Latvia
- Jānis Iraids
  Latvia
- Mārtiņš Kālis
  Latvia
- Andis Draguns
  Latvia
- Matıss Apinis
  Latvia

Program Committee

- Barbara Amaral
  Universidade de São Paulo
- Anurag Anshu
  University of California, Berkeley
- Srinivasan Arunachalam
  IBM T. J. Watson Research Center
- Juani Bermejo-Vega
  University of Granada
- Hao-Chung Cheng
  National Taiwan University
- Giulio Chiribella
  QICI, The University of Hong Kong
- Patrick Coles
  Los Alamos National Laboratory
- Matthew Coudron
  NIST/UMD
- Yfke Dulek
  QuSoft/CWI, Amsterdam
- Bill Fefferman
  University of Chicago
- Christoph Hirche
  Københavns Universitet
- Min-Hsiu Hsieh (Chair)
  Hon Hai Quantum Computing Center
- Rahul Jain
  National University of Singapore
- Anna Jenčovà
  Slovak Academy of Sciences
- Maria Kieferova
  University of Technology Sydney
- Isaac Kim
  University of Sydney
- Ludovico Lami
  Universität Ulm
- Felix Leditzky
  University of Illinois at Urbana-Champaign
- Anthony Leverrier
  Inria
- Guang-Hao Low
  Microsoft
- Chandrashekar C M
  Institute of Mathematical Sciences
- Xiongfeng Ma
  Tsinghua University
- Tomoyuki Morimae
  Kyoto University
- Anand Natarajan
  MIT
- Miguel Navascues
  IQOQI Vienna
- Yingkai Ouyang
  National University of Singapore
- Dave Touchette
  Université de Sherbrooke

16th Conference on the Theory of Quantum Computation, Communication and Cryptography (TQC 2021).
Editor: Min-Hsiu Hsieh
LIPICS Leibniz International Proceedings in Informatics
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

- Anna Vershynina
  University of Houston
- Julio I. de Vicente
  Universidad Carlos III de Madrid
- Mischa Woods
  ETH
- Penghui Yao
  Nanjing University
- Jon Yard
  IQC/Perimeter Institute
- Quntao Zhuang
  University of Arizona

Steering Committee

- Gorjan Alagic
  Maryland
- Andris Ambainis
  Latvia
- Anne Broadbent
  Ottawa
- Eric Chitambar
  UIUC
- Steven Flammia
  AWS
- Stacey Jeffery (chair)
  QuSoft, CWI
- Laura Mančinska
  Copenhagen
- Marco Tomamichel
  NUS

List of Authors

Andris Ambainis (6)
Center for Quantum Computer Science, Faculty of Computing, University of Latvia, Riga, Latvia

Kaspars Balodis (6)
Center for Quantum Computer Science, Faculty of Computing, University of Latvia, Riga, Latvia

Fernando G.S L. Brandão (7)
AWS Center for Quantum Computing, Pasadena, CA, USA; Institute for Quantum Information and Matter, California Institute of Technology, Pasadena, CA, USA

Kai-Min Chung (3)
Institute of Information Science, Academia Sinica, Taipei, Taiwan

Steven T. Flammia (8)
AWS Center for Quantum Computing, Pasadena, CA, USA; IQIM, California Institute of Technology, Pasadena, CA, USA

Daniel Stilck França (7)
QMATH, Department of Mathematical Sciences, University of Copenhagen, Denmark

Yassine Hamoudi (1)
Université de Paris, IRIF, CNRS, F-75013 Paris, France

Jānis Iraids (6)
Center for Quantum Computer Science, Faculty of Computing, University of Latvia, Riga, Latvia

William Kretschmer (2)
University of Texas at Austin, TX, USA

Richard Kueng (7)
Institute for Integrated Circuits, Johannes Kepler University Linz, Austria

François Le Gall (10)
Graduate School of Mathematics, Nagoya University, Japan

Han-Hsuan Lin (3)
Department of Computer Science, National Tsing Hua University, Hsinchu, Taiwan

Yupan Liu (4)
Shenzhen, China

Frédéric Magniez (1)
Université de Paris, IRIF, CNRS, F-75013 Paris, France

Harumichi Nishimura (10)
Graduate School of Informatics, Nagoya University, Japan

Ryan O'Donnell (8)
Computer Science Department, Carnegie Mellon University, Pittsburgh, PA, USA

Prithviraj Prabhu (5)
Department of Electrical and Computer Engineering, University of Southern California, Los Angeles, CA, USA

Ben W. Reichardt (5)
Department of Electrical and Computer Engineering, University of Southern California, Los Angeles, CA, USA

Joran van Apeldoorn (9)
Institute for Information Law, University of Amsterdam, The Netherlands; QuSoft, Centrum Wiskunde & Informatica, Amsterdam, The Netherlands

Abuzer Yakaryılmaz (10)
Center for Quantum Computer Science, University of Latvia, Rīga, Latvia; QWorld Association, Tallinn, Estonia

16th Conference on the Theory of Quantum Computation, Communication and Cryptography (TQC 2021).
Editor: Min-Hsiu Hsieh
LIPICS Leibniz International Proceedings in Informatics
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

Quantum Time-Space Tradeoff for Finding Multiple Collision Pairs

Yassine Hamoudi
Université de Paris, IRIF, CNRS, F-75013 Paris, France

Frédéric Magniez
Université de Paris, IRIF, CNRS, F-75013 Paris, France

Abstract

We study the problem of finding K collision pairs in a random function $f : [N] \to [N]$ by using a quantum computer. We prove that the number of queries to the function in the quantum random oracle model must increase significantly when the size of the available memory is limited. Namely, we demonstrate that any algorithm using S qubits of memory must perform a number T of queries that satisfies the tradeoff $T^3 S \geq \Omega(K^3 N)$. Classically, the same question has only been settled recently by Dinur [22, Eurocrypt'20], who showed that the Parallel Collision Search algorithm of van Oorschot and Wiener [32] achieves the optimal time-space tradeoff of $T^2 S = \Theta(K^2 N)$. Our result limits the extent to which quantum computing may decrease this tradeoff. Our method is based on a novel application of Zhandry's recording query technique [41, Crypto'19] for proving lower bounds in the exponentially small success probability regime. As a second application, we give a simpler proof of the time-space tradeoff $T^2 S \geq \Omega(N^3)$ for sorting N numbers on a quantum computer, which was first obtained by Klauck, Špalek and de Wolf [29].

2012 ACM Subject Classification Theory of computation → Quantum complexity theory

Keywords and phrases Quantum computing, query complexity, lower bound, time-space tradeoff

Digital Object Identifier 10.4230/LIPIcs.TQC.2021.1

Related Version *Preprint Version*: https://arxiv.org/abs/2002.08944

Funding This research was supported in part by the ERA-NET Cofund in Quantum Technologies project QuantAlgo and the French ANR Blanc project RDAM.

Acknowledgements The authors want to thank the anonymous referees for their valuable comments and suggestions which helped to improve this paper.

1 Introduction

The *efficiency* of a cryptographic attack is a hard-to-define concept that must express the interplay between different computational resources [38, 11, 12]. Arguably, the two most used criteria are the *time* complexity, measured for instance as the number of queries to a random oracle, and the *space* complexity, which is the memory size needed to perform the attack. *Time-space tradeoffs* aim at connecting these two quantities by studying how much the time increases when the available space decreases. Devising security proofs that are sensitive to memory constraints is a challenging program. Indeed, very few tools are available to study the impact of space on the security level of a scheme. A recent line of work [35, 27, 25] has made some progress for the case of *classical* attackers with bounded memory. The development of quantum computing asks the question of whether the access to quantum operations and quantum memories may lower the security levels. The answer is unclear when taking space into account. Indeed, many quantum "speed-ups" come at the cost of a dramatic increase in the space requirement [16, 6, 30]. A central question is whether a speed-up both in terms of time and space complexities is achievable for such problems?

16th Conference on the Theory of Quantum Computation, Communication and Cryptography (TQC 2021).
Editor: Min-Hsiu Hsieh; Article No. 1; pp. 1:1–1:21
Leibniz International Proceedings in Informatics
LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

The focus of this work is to provide time-space tradeoff lower bounds for the problem of finding *multiple collision pairs* in a random function. The search for a single collision pair is one of the cornerstones of cryptanalysis. Classically, the birthday attack can be achieved by the mean of a *memoryless* (i.e. logarithmic-size memory) algorithm using Pollard's rho method [33]. On the other hand, the quantum BHT algorithm [16] requires fewer queries to the random function, but the product of its time and space complexities is higher than that of the classical attack! In this paper, we address the problem of finding *multiple* collision pairs. This task plays a central role in low-memory meet-in-the-middle attacks [32, 22], with applications to double and triple encryption [32], subset sum [23, 21], k-sum [37], 3-collision [28], etc. Recently, it was used to attack the post-quantum cryptography candidates NTRU [36] and SIKE [4]. The Parallel Collision Search algorithm of van Oorschot and Wiener [32] can find as many collision pairs as desired in a time that depends on the available memory. The question of whether this algorithm achieves the optimal classical time-space tradeoff has been settled positively by Chakrabarti and Chen [18] (for the case of 2-to-1 random functions) and Dinur [22] (for the case of uniformly random functions). In the quantum setting, no time-space tradeoff was known prior to our work.

We point out that time-space tradeoffs have been studied for a long time in the complexity community [14, 9, 13, 39, 10, 3, 31]. The few results known in the quantum circuit model are for the Sorting problem [29], Boolean Matrix-Vector and Matrix-Matrix Multiplication [29], and Evaluating Solutions to Systems of Linear Inequalities [8]. Apart from our work, all existing quantum tradeoffs are based on the hardness of Quantum Search. We use the machinery developed in our paper to give a simpler proof of the tradeoffs obtained in [29].

1.1 Our results

The *Collision Pairs Finding* problem asks to find a certain number K of disjoint collision pairs in a random function $f : [M] \to [N]$ where $M \geq N$. A *collision pair* (or simply *collision*) is a pair of values $x_1 \neq x_2$ such that $f(x_1) = f(x_2)$. Two collisions (x_1, x_2) and (x_3, x_4) are *disjoint* if $x_1, \dots, x_4$ are all different. We measure the time T of an algorithm solving this problem as the number of query accesses to f, and the space S as the amount of memory used. We assume that the output is produced in an online fashion, meaning that a collision can be output as soon as it is discovered. The length of the output is not counted toward the space bound and the same collision may be output several times (but it contributes only once to the total count). The requirement for the collisions to be disjoint is made to simplify our proofs later on. We note that a random function $f : [N] \to [N]$ contains $(1 - 2/e)N$ disjoint collisions on average [24].

Classically, the single-processor Parallel Collision Search algorithm [32] achieves an optimal [22] time-space tradeoff of[1] $T^2 S = \widetilde{\Theta}(K^2 N)$ for any amount of space S between $\widetilde{\Omega}(\log N)$ and $\widetilde{O}(K)$. In the quantum setting, the BHT algorithm [16] can find a single collision in time $T = \widetilde{O}(N^{1/3})$ and space $S = \widetilde{O}(N^{1/3})$. In Algorithm 2, we adapt it for finding an arbitrary number K of collisions at cost $T^2 S \leq \widetilde{O}(K^2 N)$. This is the same tradeoff as classically, except that the space parameter S can hold larger values up to $\widetilde{O}(K^{2/3} N^{1/3})$, hence the existence of a quantum speed-up when there is no memory constraint.

► **Proposition 17** (restated). *For any $1 \leq K \leq O(N)$ and $\widetilde{\Omega}(\log N) \leq S \leq \widetilde{O}(K^{2/3} N^{1/3})$, there exists a bounded-error quantum algorithm that can find K collisions in a random function $f : [N] \to [N]$ by making $T = \widetilde{O}(K\sqrt{N/S})$ queries and using S qubits of memory.*

[1] The notation $\widetilde{\ }$ is used to denote the presence of hidden polynomial factors in $\log(N)$ or $1/\log(N)$.

The BHT algorithm achieves the optimal time complexity for finding one collision [2, 40]. Our first main result is to provide a similar lower bound for the problem of finding K disjoint collisions. We prove that the optimal time complexity is $T \geq \Omega(K^{2/3}N^{1/3})$. This bound is matched by Proposition 17 when $S = \Theta(K^{2/3}N^{1/3})$. More precisely, we show that the optimal success probability decreases at an exponential rate in K below this bound. This property is of crucial importance for proving our time-space tradeoff next. We note that, similarly to [40], the bound is independent of the size M of the domain as long as $M \geq N$.

▶ **Theorem 9** (restated). *The success probability of finding K disjoint collisions in a random function $f : [M] \to [N]$ is at most $O(T^3/(K^2N))^{K/2} + 2^{-K}$ for any algorithm making T quantum queries to f and any $1 \leq K \leq N/8$.*

Our second main result is the next time-space tradeoff for the same problem of finding K collisions in a random function. We summarize the tradeoffs known for this problem in Table 1. We note that $T^2S \geq \Omega(K^2N)$ is always stronger than $T^3S \geq \Omega(K^3N)$ since $T \geq K$.

▶ **Theorem 10** (restated). *Any quantum algorithm for finding K disjoint collisions in a random function $f : [M] \to [N]$ with success probability $2/3$ must satisfy a time-space tradeoff of $T^3S \geq \Omega(K^3N)$, where $1 \leq K \leq N/8$.*

We obtain that $T \geq \Omega(N^{4/3})$ quantum queries are needed to find almost all collisions when $S = O(\log N)$, whereas $T = N$ classical queries are sufficient when there is no space restriction. We further show that any improvement to this lower bound would imply a breakthrough for the *Element Distinctness* problem, which consists of finding a single collision in a random function $f : [N] \to [N^2]$ (or, more generally, deciding if a function contains a collision). It is a long-standing open question to prove a time-space lower bound for this problem. Although there is some progress in the classical case [13, 39, 10], no result is known in the quantum setting. We give a reduction that converts any tradeoff for finding multiple collisions into a tradeoff for Element Distinctness. We state a particular case of our reduction below.

▶ **Corollary 14** (restated). *Suppose that there exists $\epsilon > 0$ such that any quantum algorithm for finding $\widetilde{\Omega}(N)$ disjoint collisions in a random function $f : [10N] \to [N]$ must satisfy a time-space tradeoff of $TS^{1/3} \geq \widetilde{\Omega}(N^{4/3+\epsilon})$. Then, any quantum algorithm for solving Element Distinctness on domain size N must satisfy a time-space tradeoff of $TS^{1/3} \geq \widetilde{\Omega}(N^{2/3+2\epsilon})$.*

We point out that $TS^{1/3} \geq \Omega(N^{2/3})$ can already be deduced from the query complexity of Element Distinctness [2] and $S \geq 1$. We conjecture that our current tradeoff for finding K collisions can be improved to $T^2S \geq \Omega(K^2N)$, which would imply $T^2S \geq \widetilde{\Omega}(N^2)$ for Element Distinctness (Corollary 16). This result would be optimal [6].

Finally, we adapt the machinery developed in our paper to study the K-Search problem, which consists of finding K preimages of 1 in a function $f : [M] \to \{0, 1\}$ where $f(x) = 1$ with probability K/N for each x. Several variants of this problem have been considered in the literature before [29, 7, 34], where it was shown that the success probability must be exponentially small in K when the number of quantum queries is smaller than $O(\sqrt{KN})$. Our proof is the first one to consider this particular input distribution, and it is arguably simpler and more intuitive than previous work.

▶ **Theorem 18** (restated). *The success probability of finding $K \leq N/8$ preimages of 1 in a random function $f : [M] \to \{0, 1\}$ where $f(x) = 1$ with probability K/N for each $x \in [M]$ is at most $O(T^2/(KN))^{K/2} + 2^{-K}$ for any algorithm using T quantum queries to f.*

As an application, we reprove the quantum time-space tradeoff for sorting N numbers [29].

▶ **Theorem 24** (restated). *Any quantum algorithm for sorting a function $f : [N] \to \{0, 1, 2\}$ with success probability $2/3$ must satisfy a time-space tradeoff of $T^2S \geq \Omega(N^3)$.*

Table 1 Complexity to find K disjoint collisions in a random function $f : [M] \to [N]$.

	Classical complexity	Quantum complexity
Upper bound:	$T^2 S \leq \widetilde{O}(K^2 N)$ *when* $\widetilde{\Omega}(\log N) \leq S \leq \widetilde{O}(K)$ Parallel Collision Search [32]	$T^2 S \leq \widetilde{O}(K^2 N)$ *when* $\widetilde{\Omega}(\log N) \leq S \leq \widetilde{O}(K^{2/3} N^{1/3})$ Proposition 17
Lower bound:	$T^2 S \geq \Omega(K^2 N)$ [22]	$T^3 S \geq \Omega(K^3 N)$ Theorem 10

1.2 Our techniques

Recording Query Technique. We use the recording query framework of Zhandry [41] to upper bound the success probability of a query-bounded algorithm in finding K collision pairs. This method intends to reproduce the classical strategy where the queries made by an algorithm (the *attacker*) are recorded and answered with on-the-fly simulation of the oracle. Zhandry brought this technique to the quantum random oracle model by showing that, for the uniform input distribution, one can record *in superposition* the queries made by a quantum algorithm. Our first technical contribution (Section 3) is to simplify the analysis of Zhandry's technique and, as a byproduct, to generalize it to any product distribution on the input. We notice that there has been other independent work on extending Zhandry's recording technique [26, 20, 19]. Our approach does not require moving to the Fourier domain (as in [20] for instance). It is based on defining a "recording query operator" that is specific to the input distribution under consideration. This operator can replace the standard quantum query operator without changing the success probability of the algorithm, but with the effect of "recording" the quantum queries in an additional register. We detail two recording query operators corresponding to the uniform distribution (Lemma 5) and to the product of Bernoulli distributions (Lemma 20).

Finding collisions with time-bounded algorithms. Our application of the recording technique to the Collision Pairs Finding problem has two stages. We first bound the probability that the algorithm has forced the recording of many collisions after T queries. Namely, we show that the norm of the quantum state that records a new collision at the t-th query is on the order of $\sqrt{t/N}$ (Proposition 7). This is related to the probability that a new random value collides with one of the at most t previously recorded queries. The reason why the collisions have to be disjoint is to avoid the recording of more than one new collision in one query. By solving a simple recurrence relation, one gets that the amplitude of the basis states that have recorded at least $K/2$ collisions after T queries is at most $O(T^{3/2}/(K\sqrt{N}))^{K/2}$. We note that Liu and Zhandry [30, Theorem 5] carried out a similar analysis for the multi-collision finding problem, where they obtained a similar bound of $O(T^{3/2}/\sqrt{N})^{K/2}$. The second stage of our proof relates the probability of having recorded many collisions to the actual success probability of the algorithm. If we used previous approaches (notably [41, Lemma 5]), this step would degrade the upper bound on the success probability by adding a term that is polynomial in K/N. We preserve the exponentially small dependence on K by doing a more careful analysis of the relation between the recording and the standard query models (Proposition 8). We adopt a similar approach for analyzing the K-Search problem in Appendix A.

Finding collisions with time-space bounded algorithms. We convert the above time-only bound into a time-space tradeoff by using the time-segmentation method [14, 29]. Given a quantum circuit that solves the Collision Pairs Finding problem in time T and space S, we slice it into $T/(S^{2/3}N^{1/3})$ consecutive subcircuits, each of them using $S^{2/3}N^{1/3}$ queries. If no slice can output more than $\Omega(S)$ collisions with high probability then there must be at least $\Omega(K/S)$ slices in total, thus proving the desired tradeoff. Our previous lower bound implies that it is impossible to find $\Omega(S)$ collisions with probability larger than 4^{-S} in time $S^{2/3}N^{1/3}$. We must take into account that the initial memory at the beginning of each slice carries out information from previous stages. As in previous work [1, 29], we can "eliminate" this memory by replacing it with the completely mixed state while decreasing the success probability by a factor of 2^{-S}. Thus, if a slice outputs $\Omega(S)$ collisions then it can be used to contradict the lower bound proved before.

Element Distinctness. We connect the Collision Pairs Finding and Element Distinctness problems by showing how to transform a low-space algorithm for the latter into one for the former (Proposition 12). If there is a time-$\bar{T}$ space-$\bar{S}$ algorithm for Element Distinctness on domain size $\sqrt{N}$ then we find $\widetilde{\Omega}(N)$ collisions in a random function $f : [N] \to [N]$ by repeatedly sampling a subset $H \subset [N]$ of size $\sqrt{N}$ and using that algorithm on the function f restricted to the domain H. Among other things, we must ensure that the same collision does not occur many times and that storing H does not use too much memory (it turns out that 4-wise independence is sufficient for our purpose). We end up with an algorithm with time $T = O(N\bar{T})$ and space $S = O(\bar{S})$. Consequently, if the Element Distinctness problem on domain size $\sqrt{N}$ can be solved with a time-space tradeoff of $\bar{T}\bar{S}^{1/3} \leq O(N^{1/3+\epsilon})$, then there is an algorithm for finding $\widetilde{\Omega}(N)$ collisions that satisfies a tradeoff of $TS^{1/3} \leq O(N^{4/3+\epsilon})$.

2 Models of computation

We first present the standard model of quantum query complexity in Section 2.1. This model is used for investigating the *time complexity* of the Collision Pairs Finding problem in Section 4, and of the K-Search problem in Appendix A. Then, we describe the more general circuit model that also captures the *space complexity* in Section 2.2. It is used in Section 5 and Appendix B for studying time-space tradeoffs.

2.1 Query model

The (standard) model of quantum query complexity [17] measures the number of quantum queries an algorithm (also called an "attacker") needs to make on an input $f : [M] \to [N]$ to find an output z satisfying some fixed relation $\mathrm{R}(f, z)$. This model is presented below.

Quantum Query Algorithm. A T-query quantum algorithm is specified by a sequence $U_0, \ldots, U_T$ of unitary transformations acting on the algorithm's memory. The state $|\psi\rangle$ of the algorithm is made of three registers $\mathcal{Q}$, $\mathcal{P}$, $\mathcal{W}$ where the *query register* $\mathcal{Q}$ holds $x \in [M]$, the *phase register* $\mathcal{P}$ holds $p \in [N]$ and the *working register* $\mathcal{W}$ holds some value w. We represent a basis state in the corresponding Hilbert space as $|x, p, w\rangle_{\mathcal{QPW}}$. We may drop the subscript $\mathcal{QPW}$ when it is clear from the context. The state $|\psi_t^f\rangle$ of the algorithm after $t \leq T$ queries to some input function $f : [M] \to [N]$ is

$$|\psi_t^f\rangle = U_t \mathcal{O}_f U_{t-1} \cdots U_1 \mathcal{O}_f U_0 |0\rangle$$

where the oracle $\mathcal{O}_f$ is defined by $\mathcal{O}_f|x, p, w\rangle = \omega_N^{pf(x)}|x, p, w\rangle$ and $\omega_N = e^{\frac{2i\pi}{N}}$.

The *output* of the algorithm is written on a substring z of the value w. The *success probability* σ_f of the quantum algorithm on f is the probability that the output value z obtained by measuring the working register of $|\psi_T^f\rangle$ in the computational basis satisfies the relation $\mathrm{R}(f,z)$. Thus, if we let Π^f_{succ} be the projector whose support consists of all basis states $|x,p,w\rangle$ such that the output substring z of w satisfies $\mathrm{R}(f,z)$, then $\sigma_f = \left\|\Pi^f_{\mathrm{succ}}|\psi_T^f\rangle\right\|^2$.

Oracle's Register. Here, we describe the variant used in the adversary method [5] and in Zhandry's work [41]. It is represented as an interaction between an *algorithm* that aims at finding a correct output z, and a superposition of *oracle*'s inputs that respond to the queries from the algorithm.

The memory of the oracle is made of an *input register* $\mathcal{F}$ holding the description of a function $f : [M] \to [N]$. This register is divided into M subregisters $\mathcal{F}_1, \ldots, \mathcal{F}_M$ where $\mathcal{F}_x$ holds $f(x) \in [N]$ for each $x \in [M]$. The basis states in the corresponding Hilbert space are $|f\rangle_{\mathcal{F}} = \otimes_{x\in[M]}|f(x)\rangle_{\mathcal{F}_x}$. Given an input distribution D on the set of functions $[N]^M$, the *oracle's initial state* is the state $|\mathrm{init}\rangle_{\mathcal{F}} = \sum_{f\in[N]^M} \sqrt{\Pr[f \leftarrow D]}|f\rangle$.

The *query operator* $\mathcal{O}$ is a unitary transformation acting on the memory of the algorithm and the oracle. Its action is defined on each basis state by $\mathcal{O}|x,p,w\rangle|f\rangle = (\mathcal{O}_f|x,p,w\rangle)|f\rangle$.

The joint state $|\psi_t\rangle$ of the algorithm and the oracle after t queries is equal to $|\psi_t\rangle = U_t\mathcal{O}U_{t-1}\cdots U_1\mathcal{O}U_0(|0\rangle|\mathrm{init}\rangle) = \sum_{f\in[N]^M} \sqrt{\Pr[f \leftarrow D]}|\psi_t^f\rangle|f\rangle$, where the unitaries U_i have been extended to act as the identity on $\mathcal{F}$. The *success probability* σ of a quantum algorithm on an input distribution D is the probability that the output value z and the input f obtained by measuring the working and input registers of the final state $|\psi_T\rangle$ satisfy the relation $\mathrm{R}(f,z)$. In other words, if we let Π_{succ} be the projector whose support consists of all basis states $|x,p,w\rangle|f\rangle$ such that the output substring z of w satisfies $\mathrm{R}(f,z)$, then $\sigma = \|\Pi_{\mathrm{succ}}|\psi_T\rangle\|^2$.

2.2 Space-bounded model

Our model of space-bounded computation is identical to the one described in [29, 8]. We use the quantum circuit model augmented with the oracle gates of the query model defined in the previous section. The *time complexity*, denoted by T, is the number of gates in the circuit. In practice, we lower bound it by the number of oracle gates only. The *space complexity*, denoted by S, is the number of qubits on which the circuit is operating. The result of the computation is written on some dedicated output qubits that may not be used later on, and that are *not counted toward the space bound*. In particular, the size of the output can be larger than S. Furthermore, we assume that the output qubits are updated at some predefined output gates in the circuit.

We notice that, by the deferred measurement principle, any space-bounded computation that uses T queries can be transformed into a T-query unitary algorithm as defined in Section 2.1. Thus, any lower bound on the query complexity of a problem is also a lower bound on the time complexity of that problem in the space-bounded model. This explains our use of the query model in Section 4 and Appendix A.

3 Recording model

The quantum recording query model is a modification of the standard query model defined in Section 2.1 that is unnoticeable by the algorithm, but that allows us to track more easily the progress made toward solving the problem under consideration. The original recording model was formulated by Zhandry in [41]. Here, we propose a simplified and more general version of this framework that only requires the initial oracle's state $|\text{init}\rangle_{\mathcal{F}}$ to be a product state $\otimes_{x\in[M]}|\text{init}_x\rangle_{\mathcal{F}_x}$ (instead of the uniform distribution over all basis states as in [41]).

Construction. The range $[N]$ is augmented with a new symbol $\perp$. The input register $\mathcal{F}$ of the oracle can now contain $f : [M] \to [N] \cup \{\perp\}$, where $f(x) = \perp$ represents the absence of knowledge from the algorithm about the image of x. Unlike in the standard query model, the oracle's initial state is independent of the input distribution and is fixed to be $|\perp^M\rangle_{\mathcal{F}}$ (which represents the fact that the algorithm knows nothing about the input initially). We extend the query operator $\mathcal{O}$ defined in the standard query model by setting

$$\mathcal{O}|x,p,w\rangle|f\rangle = |x,p,w\rangle|f\rangle \quad \text{when } f(x) = \perp.$$

Given a product input distribution $D = D_1 \otimes \cdots \otimes D_M$ on the set $[N]^M$, the oracle's initial state in the standard query model can be decomposed as the product state $|\text{init}\rangle_{\mathcal{F}} = \otimes_{x\in[M]}|\text{init}_x\rangle_{\mathcal{F}_x}$ where $|\text{init}_x\rangle_{\mathcal{F}_x} := \sum_{y\in[N]} \sqrt{\Pr[y \leftarrow D_x]}|y\rangle_{\mathcal{F}_x}$. The "recording query operator" $\mathcal{R}$ is defined with respect to a family $(\mathcal{S}_x)_{x\in[M]}$ of unitary operators satisfying $\mathcal{S}_x|\perp\rangle_{\mathcal{F}_x} = |\text{init}_x\rangle_{\mathcal{F}_x}$ for all x as follows.

▶ **Definition 1.** *Given M unitary operators $\mathcal{S}_1, \ldots, \mathcal{S}_M$ acting on $\mathcal{F}_1, \ldots, \mathcal{F}_M$ respectively, consider the operator $\mathcal{S}$ acting on all the registers $\mathcal{QPWF}$ such that,*

$$\mathcal{S} = \sum_{x\in[M]} |x\rangle\langle x|_{\mathcal{Q}} \otimes I_{\mathcal{PWF}_1\ldots\mathcal{F}_{x-1}} \otimes \mathcal{S}_x \otimes I_{\mathcal{F}_{x+1}\ldots\mathcal{F}_M}.$$

Then, the recording query operator *$\mathcal{R}$ with respect to $(\mathcal{S}_x)_{x\in[M]}$ is defined as $\mathcal{R} = \mathcal{S}^\dagger\mathcal{O}\mathcal{S}$.*

Later in this paper, we describe two recording query operators related to the uniform distribution (Lemma 5) and to the product of Bernoulli distributions (Lemma 20).

Indistinguishability. The joint state of the algorithm and the oracle after t queries in the recording query model is defined as $|\phi_t\rangle = U_t\mathcal{R}U_{t-1}\cdots U_1\mathcal{R}U_0\big(|0\rangle|\perp^M\rangle\big)$. Notice that the query operator $\mathcal{R}$ can only change the value of $f(x')$ (contained in the register $\mathcal{F}_{x'}$) when it is applied to a state $|x,p,w\rangle|f\rangle$ such that $x = x'$. As a result, we have the following fact.

▶ **Fact 2.** *The state $|\phi_t\rangle$ is a linear combination of basis states $|x,p,w\rangle|f\rangle$ where f contains at most t entries different from $\perp$.*

The entries of f that are different from $\perp$ represent what the oracle has learned (or "recorded") from the algorithm's queries so far. In the next theorem, we show that $|\phi_t\rangle$ is related to the state $|\psi_t\rangle$ (defined in Section 2.1) by $|\psi_t\rangle = \big(I_{\mathcal{QPW}} \otimes_{x\in[M]} \mathcal{S}_x\big)|\phi_t\rangle$. In particular, the states $|\psi_t\rangle$ and $|\phi_t\rangle$ cannot be distinguished by the algorithm since the reduced states on the algorithm's registers are identical.

▶ **Theorem 3.** *Let $(U_0, \ldots, U_T)$ be a T-query quantum algorithm. Given M unitary operators $\mathcal{S}_1, \ldots, \mathcal{S}_M$ acting on the oracle's registers $\mathcal{F}_1, \ldots, \mathcal{F}_M$ respectively, let $\mathcal{R}$ denote the recording query operator associated with $(\mathcal{S}_x)_{x\in[M]}$, and define the initial state $|\text{init}\rangle_{\mathcal{F}} = \big(\otimes_{x\in[M]}\mathcal{S}_x\big)|\perp^M\rangle$. Then, the states*

$$\begin{cases} |\psi_t\rangle = U_t\mathcal{O}U_{t-1}\cdots U_1\mathcal{O}U_0\big(|0\rangle|\text{init}\rangle\big) \\ |\phi_t\rangle = U_t\mathcal{R}U_{t-1}\cdots U_1\mathcal{R}U_0\big(|0\rangle|\perp^M\rangle\big) \end{cases}$$

after $t \leq T$ queries in the standard and recording query models respectively satisfy

$$|\psi_t\rangle = \mathcal{T}|\phi_t\rangle \quad \textit{where} \quad \mathcal{T} = I_{\mathcal{QPW}} \bigotimes_{x\in[M]} \mathcal{S}_x.$$

Proof. We start by introducing the intermediate operator $\bar{\mathcal{R}} = \mathcal{T}^\dagger \mathcal{O} \mathcal{T}$. Observe that for any basis state $|x, p, w\rangle|f\rangle$ the operators $\bar{\mathcal{R}}$ and $\mathcal{R}$ act the same way on the registers $\mathcal{QPF}_x$ and they do not depend on the other registers. Thus, we have $\bar{\mathcal{R}} = \mathcal{R}$. We also observe that U_i and $\mathcal{T}$ commute for all i since they depend on disjoint registers. Consequently, we have that

$$\begin{aligned}
|\psi_t\rangle &= U_t \mathcal{O} U_{t-1} \mathcal{O} \cdots U_1 \mathcal{O} U_0 \cdot \mathcal{T}\big(|0\rangle|\bot^M\rangle\big) && \text{since } \mathcal{T}\big(|0\rangle|\bot^M\rangle\big) = |0\rangle|\text{init}\rangle \\
&= \mathcal{T}\mathcal{T}^\dagger U_t \mathcal{O} \cdot \mathcal{T}\mathcal{T}^\dagger U_{t-1} \mathcal{O} \cdots \mathcal{T}\mathcal{T}^\dagger U_1 \mathcal{O} \cdot \mathcal{T}\mathcal{T}^\dagger U_0 \cdot \mathcal{T}\big(|0\rangle|\bot^M\rangle\big) && \text{since } \mathcal{T}\mathcal{T}^\dagger = I \\
&= \mathcal{T} U_t \mathcal{T}^\dagger \cdot \mathcal{O} \cdot \mathcal{T} U_{t-1} \mathcal{T}^\dagger \cdot \mathcal{O} \cdots \mathcal{T} U_1 \mathcal{T}^\dagger \cdot \mathcal{O} \cdot \mathcal{T} U_0 \big(|0\rangle|\bot^M\rangle\big) && \text{by commutation} \\
&= \mathcal{T} U_t \bar{\mathcal{R}} U_{t-1} \cdots U_1 \bar{\mathcal{R}} U_0 \big(|0\rangle|\bot^M\rangle\big) && \text{by definition of } \bar{\mathcal{R}} \\
&= \mathcal{T} U_t \mathcal{R} U_{t-1} \cdots U_1 \mathcal{R} U_0 \big(|0\rangle|\bot^M\rangle\big) && \text{since } \bar{\mathcal{R}} = \mathcal{R} \\
&= \mathcal{T}|\phi_t\rangle && \text{by definition of } |\phi_t\rangle.
\end{aligned}$$

◀

4 Time lower bound for Collision Pairs Finding

In this section, we upper bound the success probability of finding K disjoint collisions in the query-bounded model of Section 2.1. The proof uses the recording model of Section 3. We first describe in Section 4.1 the recording query framework associated with this problem. In Section 4.2, we study the probability that an algorithm has recorded at least $k \leq K$ collisions after $t \leq T$ queries. We prove by induction on t and k that this quantity is exponentially small in k when $t \leq O(k^{2/3} N^{1/3})$ (Proposition 7). Finally, in Section 4.3, we relate this progress measure to the actual success probability (Proposition 8), and we conclude that the latter quantity is exponentially small in K after $T \leq O(K^{2/3} N^{1/3})$ queries (Theorem 9).

4.1 Recording query operator

We describe a recording operator that corresponds to the uniform distribution on the set of functions $f : [M] \to [N]$. In the standard query model, the oracle's initial state is $|\text{init}\rangle_{\mathcal{F}} = \otimes_{x \in [M]} \big(\frac{1}{\sqrt{N}} \sum_{y \in [N]} |y\rangle_{\mathcal{F}_x}\big)$. Consequently, in the recording model, we choose the unitary transformations $\mathcal{S}_1, \cdots, \mathcal{S}_M$ to be defined as follows.

▶ **Definition 4.** *For any $x \in [M]$, we define the unitary $\mathcal{S}_x$ acting on the register $\mathcal{F}_x$ to be*

$$\mathcal{S}_x : \begin{cases} |\bot\rangle_{\mathcal{F}_x} & \longmapsto \frac{1}{\sqrt{N}} \sum_{y \in [N]} |y\rangle_{\mathcal{F}_x} \\ \frac{1}{\sqrt{N}} \sum_{y \in [N]} |y\rangle_{\mathcal{F}_x} & \longmapsto |\bot\rangle_{\mathcal{F}_x} \\ \frac{1}{\sqrt{N}} \sum_{y \in [N]} \omega_N^{py} |y\rangle_{\mathcal{F}_x} & \longmapsto \frac{1}{\sqrt{N}} \sum_{y \in [N]} \omega_N^{py} |y\rangle_{\mathcal{F}_x} \quad \textit{for } p = 1, \ldots, N-1. \end{cases}$$

These unitaries verify $\mathcal{T}|\bot^M\rangle = |\text{init}\rangle$ where $\mathcal{T} = \otimes_{x \in [M]} \mathcal{S}_x$, as required by Theorem 3. The recording query operator is $\mathcal{R} = \mathcal{SOS}$ (Definition 1) since $\mathcal{S}^\dagger = \mathcal{S}$. The next lemma gives an explicit characterization of the action of $\mathcal{R}$ on a basis state.

▶ **Lemma 5.** *If the recording query operator $\mathcal{R}$ associated with Definition 4 is applied to a basis state $|x, p, w\rangle|f\rangle$ where $p \neq 0$ then the register $|f(x)\rangle_{\mathcal{F}_x}$ is mapped to*

$$\begin{cases} \sum_{y \in [N]} \frac{\omega_N^{py}}{\sqrt{N}} |y\rangle & \textit{if } f(x) = \bot \\ \frac{\omega_N^{pf(x)}}{N} |\bot\rangle + \frac{1 + \omega_N^{pf(x)}(N-2)}{N} |f(x)\rangle + \sum_{y \in [N] \setminus \{f(x)\}} \frac{1 - \omega_N^{py} - \omega_N^{pf(x)}}{N} |y\rangle & \textit{otherwise} \end{cases}$$

and the other registers are unchanged. If $p = 0$ then none of the registers are changed.

Proof. By definition, the unitary $\mathcal{S}_x$ maps $|\bot\rangle_{\mathcal{F}_x} \mapsto \frac{1}{\sqrt{N}} \sum_{y \in [N]} |y\rangle$ and $|y\rangle_{\mathcal{F}_x} \mapsto \frac{1}{\sqrt{N}} |\bot\rangle + \frac{1}{\sqrt{N}} \sum_{p' \in [N] \setminus \{0\}} \omega_N^{-p'y} |\widehat{p'}\rangle$ where $y \in [N]$ and $|\widehat{p'}\rangle := \frac{1}{\sqrt{N}} \sum_{y \in [N]} \omega_N^{p'y} |y\rangle$. Thus, the action on the register $\mathcal{F}_x$ is:

- If $f(x) = \bot$ then $|f(x)\rangle_{\mathcal{F}_x} \overset{\mathcal{S}}{\longmapsto} \frac{1}{\sqrt{N}} \sum\limits_{y \in [N]} |y\rangle \overset{\mathcal{O}}{\longmapsto} \frac{1}{\sqrt{N}} \sum\limits_{y \in [N]} \omega_N^{py} |y\rangle \overset{\mathcal{S}}{\longmapsto} \frac{1}{\sqrt{N}} \sum\limits_{y \in [N]} \omega_N^{py} |y\rangle$.
- If $f(x) \in [N]$ then $|f(x)\rangle_{\mathcal{F}_x} = \frac{1}{\sqrt{N}} \sum\limits_{p' \in [N]} \omega_N^{-p' f(x)} |\widehat{p'}\rangle \overset{\mathcal{S}}{\longmapsto} \frac{1}{\sqrt{N}} |\bot\rangle + \frac{1}{\sqrt{N}} \sum\limits_{p' \in [N] \setminus \{0\}} \omega_N^{-p' f(x)}$ $|\widehat{p'}\rangle \overset{\mathcal{O}}{\longmapsto} \frac{1}{\sqrt{N}} |\bot\rangle + \frac{1}{\sqrt{N}} \sum\limits_{p' \in [N] \setminus \{0\}} \omega_N^{-p' f(x)} |\widehat{p + p'}\rangle = \frac{1}{\sqrt{N}} |\bot\rangle + \frac{\omega_N^{pf(x)}}{\sqrt{N}} \sum\limits_{p' \in [N] \setminus \{p\}} \omega_N^{-p' f(x)} |\widehat{p'}\rangle$ $\overset{\mathcal{S}}{\longmapsto} \frac{1}{N} \sum\limits_{y \in [N]} |y\rangle + \frac{\omega_N^{pf(x)}}{\sqrt{N}} |\bot\rangle + \frac{\omega_N^{pf(x)}}{\sqrt{N}} \sum\limits_{p' \in [N] \setminus \{0,p\}} \omega_N^{-p' f(x)} |\widehat{p'}\rangle = \frac{\omega_N^{pf(x)}}{N} |\bot\rangle + \frac{1 + \omega_N^{pf(x)}(N-2)}{N}$ $|f(x)\rangle + \sum\limits_{y \in [N] \setminus \{f(x)\}} \frac{1 - \omega_N^{py} - \omega_N^{pf(x)}}{N} |y\rangle$. ◀

We note that the recording operator $\mathcal{R}$ is close to the mapping $|\bot\rangle_{\mathcal{F}_x} \mapsto \sum_{y \in [N]} \frac{\omega_N^{py}}{\sqrt{N}} |y\rangle$ and $|f(x)\rangle_{\mathcal{F}_x} \mapsto \omega_N^{pf(x)} |f(x)\rangle$ (if $f(x) \neq \bot$) up to lower-order terms of amplitude $O(1/N)$. This is analogous to a "lazy" classical oracle that would choose the value of $f(x)$ uniformly at random the first time it is queried.

4.2 Analysis of the recording progress

We define a measure of progress based on the number of disjoint collisions contained in the oracle's register of the recording model. We first give some projectors related to this quantity.

▶ **Definition 6.** *We define the following projectors by giving the basis states on which they project:*

- $\Pi_{\leq k}$, $\Pi_{=k}$ *and* $\Pi_{\geq k}$*: all basis states* $|x, p, w\rangle|f\rangle$ *such that* f *contains respectively* at most*,* exactly *or* at least k *disjoint collisions (the entries with* $\bot$ *are not considered as collisions).*
- $\Pi_{=k,\bot}$ *and* $\Pi_{=k,y}$ *for* $y \in [N]$*: all basis states* $|x, p, w\rangle|f\rangle$ *such that (1)* f *contains* exactly k *disjoint collisions, (2) the phase multiplier* p *is nonzero and (3)* $f(x) = \bot$ *or* $f(x) = y$ *respectively.*

We can now define the measure of progress $q_{t,k}$ for t queries and k collisions as

$$q_{t,k} = \|\Pi_{\geq k} |\phi_t\rangle\| \tag{1}$$

where $|\phi_t\rangle$ is the state after t queries in the recording model. The main result of this section is the following bound on the growth of $q_{t,k}$.

▶ **Proposition 7.** *For all* t *and* k*, we have that* $q_{t,k} \leq \binom{t}{k} \left(\frac{4\sqrt{t}}{\sqrt{N}}\right)^k$.

Proof. First, $q_{0,0} = 1$ and $q_{0,k} = 0$ for all $k \geq 1$ since the initial state is $|\phi_0\rangle = |0\rangle|\bot^M\rangle$. Then, we prove that $q_{t,k}$ satisfies the following recurrence relation

$$q_{t+1,k+1} \leq q_{t,k+1} + 4\sqrt{\frac{t}{N}} q_{t,k}. \tag{2}$$

From this result, it is trivial to conclude that $q_{t,k} \leq \binom{t}{k} \left(\frac{4\sqrt{t}}{\sqrt{N}}\right)^k$. In order to prove Equation (2), we first observe that $q_{t+1,k+1} = \|\Pi_{\geq k+1} U_{t+1} \mathcal{R} |\phi_t\rangle\| = \|\Pi_{\geq k+1} \mathcal{R} |\phi_t\rangle\|$ since the unitary U_{t+1}

applied by the algorithm at time $t+1$ does not modify the oracle's memory. Then, on any basis state $|x,p,w\rangle|f\rangle$, the recording query operator $\mathcal{R}$ acts as the identity on the registers $\mathcal{F}_{x'}$ for $x' \neq x$. Consequently, the basis states $|x,p,w\rangle|f\rangle$ in $|\phi_t\rangle$ that may contribute to $q_{t+1,k+1}$ must either already contain $k+1$ disjoint collisions in f, or exactly k disjoint collisions in f and $p \neq 0$. This implies that

$$q_{t+1,k+1} \leq q_{t,k+1} + \|\Pi_{\geq k+1}\mathcal{R}\Pi_{=k,\perp}|\phi_t\rangle\| + \sum_{y\in[N]} \|\Pi_{\geq k+1}\mathcal{R}\Pi_{=k,y}|\phi_t\rangle\|.$$

We first bound the term $\|\Pi_{\geq k+1}\mathcal{R}\Pi_{=k,\perp}|\phi_t\rangle\|$. Consider any basis state $|x,p,w\rangle|f\rangle$ in the support of $\Pi_{=k,\perp}$ and $|\phi_t\rangle$. The function f must contain at most t entries different from $\perp$ by Fact 2. By Lemma 5, we have $\mathcal{R}|x,p,w\rangle|f\rangle = \sum_{y\in[N]} \frac{\omega_N^{py}}{\sqrt{N}}|x,p,w\rangle|y\rangle_{\mathcal{F}_x} \otimes_{x'\neq x}|f(x')\rangle_{\mathcal{F}_{x'}}$. Since there are at most t entries in f that can collide with the value contained in the register $\mathcal{F}_x$, we have $\|\Pi_{\geq k+1}\mathcal{R}|x,p,w\rangle|f\rangle\| \leq \sqrt{t/N}$. Finally, since any two basis states in the support of $\Pi_{=k,\perp}$ remain orthogonal after $\Pi_{\geq k+1}\mathcal{R}$ is applied, we obtain that $\|\Pi_{\geq k+1}\mathcal{R}\Pi_{=k,\perp}|\phi_t\rangle\| \leq \sqrt{t/N}\|\Pi_{=k,\perp}|\phi_t\rangle\| \leq \sqrt{t/N}q_{t,k}$.

We now consider the term $\|\Pi_{\geq k+1}\mathcal{R}\Pi_{=k,y}|\phi_t\rangle\|$ for any $y \in [N]$. Again, we consider any basis state $|x,p,w\rangle|f\rangle$ in the support of $\Pi_{=k,y}$ where f has at most t entries different from $\perp$. According to Lemma 5, we have $\mathcal{R}|x,p,w\rangle|f\rangle = \frac{\omega_N^{pf(x)}}{N}|\perp\rangle + \frac{1+\omega_N^{pf(x)}(N-2)}{N}|f(x)\rangle + \sum_{y'\neq f(x)} \frac{1-\omega_N^{py'}-\omega_N^{pf(x)}}{N}|x,p,w\rangle|y'\rangle_{\mathcal{F}_x} \otimes_{x'\neq x}|f(x')\rangle_{\mathcal{F}_{x'}}$. As before, there are at most t terms in this sum that can be in the support of $\Pi_{\geq k+1}$. Consequently, $\|\Pi_{\geq k+1}\mathcal{R}|x,p,w\rangle|f\rangle\| \leq 3\sqrt{t}/N$ and $\|\Pi_{\geq k+1}\mathcal{R}\Pi_{=k,y}|\phi_t\rangle\| \leq 3\sqrt{t}/N\|\Pi_{=k,y}|\phi_t\rangle\|$.

We conclude that $q_{t+1,k+1} \leq q_{t,k+1} + \sqrt{t/N}q_{t,k} + \sum_{y\in[N]} 3\sqrt{t}/N\|\Pi_{=k,y}|\phi_t\rangle\| \leq q_{t,k+1} + \sqrt{t/N}q_{t,k} + 3\sqrt{t/N}\sqrt{\sum_{y\in[N]}\|\Pi_{=k,y}|\phi_t\rangle\|^2} \leq q_{t,k+1} + \sqrt{t/N}q_{t,k} + 3\sqrt{t/N}q_{t,k}$, where the second step is by Cauchy-Schwarz' inequality. ◀

4.3 From the recording progress to the success probability

We connect the success probability $\sigma = \|\Pi_{\mathrm{succ}}|\psi_T\rangle\|^2$ in the standard query model to the final progress $q_{T,k}$ in the recording model after T queries. We show that if the algorithm has made no significant progress for recording $k \geq K/2$ collisions then it needs to "guess" the positions of $K-k$ other collisions. Classically, the probability to find the values of $K-k$ collisions that have not been queried is at most $(1/N^2)^{K-k}$. Here, we show similarly that if a unit state contains at most k collisions in the recording model, then after mapping it to the standard query model (by applying the operator $\mathcal{T}$ of Theorem 3) the probability that the output register contains the correct positions of K collisions is at most $N^2(4K^2/N^2)^{K-k}$.

▶ **Proposition 8.** *For any state $|\phi\rangle$, we have $\|\Pi_{\mathrm{succ}}\mathcal{T}\Pi_{\leq k}|\phi\rangle\| \leq N\left(\frac{2K}{N}\right)^{K-k}\|\Pi_{\leq k}|\phi\rangle\|$.*

Proof. We assume that the output of the algorithm also contains the image of each collision pair under f. Namely, the output z is represented as a list of K triples $(x_1,x_2,y_1),\ldots,(x_{2K-1},x_{2K},y_K) \in [M]^2 \times ([N] \cup \{\perp\})$. It is correct if the input function $f:[M]\to[N]$ (in the standard query model) satisfies $f(x_{2i-1}) = f(x_{2i}) = y_i \neq \perp$ for all $1 \leq i \leq K$, and the values $x_1,x_2,\ldots,x_{2K}$ are all different. By definition, the support of Π_{succ} consists of all basis states $|x,p,w\rangle|f\rangle$ such that the output substring z of w satisfies these conditions.

We define a new family of projectors $\tilde{\Pi}_{a,b}$, where $0 \leq a+b \leq 2K$, whose supports consist of all basis states $|x,p,w\rangle|f\rangle$ satisfying the following conditions:

(A) The output substring z is made of K triples $(x_1,x_2,y_1),\ldots,(x_{2K-1},x_{2K},y_K)$ where the x_i are all different.

(B) There are exactly a indices $i \in [2K]$ such that $f(x_i) = \perp$.
(C) There are exactly b indices $i \in [2K]$ such that $f(x_i) \neq \perp$ and $f(x_i) \neq y_{\lceil i/2 \rceil}$.

For any state $|x,p,w\rangle|f\rangle$ in the support of $\tilde{\Pi}_{a,b}$, we claim that

$$\|\Pi_{\mathrm{succ}}\mathcal{T}|x,p,w\rangle|f\rangle\| \leq \left(\frac{1}{\sqrt{N}}\right)^a \left(\frac{1}{N}\right)^b. \tag{3}$$

Indeed, we have $\mathcal{T} = \otimes_{x' \in [M]} \mathcal{S}_{x'}$ and by Definition 4 the action of $\mathcal{S}_{x_i}$ on the register $|f(x_i)\rangle_{\mathcal{F}_{x_i}}$ is $|f(x_i)\rangle \mapsto \frac{1}{\sqrt{N}} \sum_{y \in [N]} |y\rangle$ if $f(x_i) = \perp$, and $|f(x_i)\rangle \mapsto \frac{1}{\sqrt{N}}|\perp\rangle + (1 - \frac{1}{N})|f(x_i)\rangle - \frac{1}{N} \sum_{y \in [N] \setminus f(x_i)} |y\rangle$ otherwise. The projector Π_{succ} only keeps the term $|y_{\lceil i/2 \rceil}\rangle$ in these sums, which implies Equation (3).

Let us now consider any linear combination $|\varphi\rangle = \sum_{x,p,w,f} \alpha_{x,p,w,f} |x,p,w\rangle|f\rangle$ of basis states that are in the support of $\tilde{\Pi}_{a,b}$. We claim that

$$\|\Pi_{\mathrm{succ}}\mathcal{T}|\varphi\rangle\| \leq \left(\sqrt{\frac{2K}{N}}\right)^{a+b} \||\varphi\rangle\|. \tag{4}$$

First, given two basis states $|x,p,w\rangle|f\rangle$ and $|\bar{x},\bar{p},\bar{w}\rangle|\bar{f}\rangle$ where $z = ((x_1,x_2,y_1),\ldots,(x_{2K-1}, x_{2K},y_K))$ is the output substring of w, if the tuples $\big(x,p,w,(f(x'))_{x' \notin \{x_1,\ldots,x_{2K}\}}\big)$ and $\big(\bar{x},\bar{p},\bar{w},(\bar{f}(x'))_{x' \notin \{x_1,\ldots,x_{2K}\}}\big)$ are different then $\Pi_{\mathrm{succ}}\mathcal{T}|x,p,w\rangle|f\rangle$ must be orthogonal to $\Pi_{\mathrm{succ}}\mathcal{T}|\bar{x},\bar{p},\bar{w}\rangle|\bar{f}\rangle$. Moreover, for any $z = ((x_1,x_2,y_1),\ldots,(x_{2K-1},x_{2K},y_K))$ that satisfies condition (A), there are $\binom{2K}{a}\binom{2K-a}{b}(N-1)^b \leq (2K)^{a+b}N^b$ different ways to choose $(f(x_i))_{i \in [2K]}$ that satisfy conditions (B) and (C). Let us write $w_{\vec{x}} = \{x_1,\ldots,x_{2K}\}$ when the output substring z of w contains $x_1,\ldots,x_{2K}$. Then, by using the Cauchy-Schwarz inequality and Equation (3), we get that

$$\begin{aligned}
\|\Pi_{\mathrm{succ}}\mathcal{T}|\varphi\rangle\|^2 &= \sum_{x,p,w,(f(x'))_{x' \notin w_{\vec{x}}}} \left\| \sum_{(f(x'))_{x' \in w_{\vec{x}}}} \alpha_{x,p,w,f} \Pi_{\mathrm{succ}}\mathcal{T}|x,p,w\rangle|f\rangle \right\|^2 \\
&\leq \sum_{x,p,w,(f(x'))_{x' \notin w_{\vec{x}}}} \left(\sum_{(f(x'))_{x' \in w_{\vec{x}}}} |\alpha_{x,p,w,f}|^2 \right) \left(\sum_{(f(x'))_{x' \in w_{\vec{x}}}} \|\Pi_{\mathrm{succ}}\mathcal{T}|x,p,w\rangle|f\rangle\|^2 \right) \\
&\leq \||\varphi\rangle\|^2 \cdot (2K)^{a+b} N^b \left(\frac{1}{N}\right)^a \left(\frac{1}{N^2}\right)^b \\
&= \left(\frac{2K}{N}\right)^{a+b} \||\varphi\rangle\|^2,
\end{aligned}$$

which proves Equation (4). The support of $\Pi_{\leq k}$ is contained into the union of the supports of $\tilde{\Pi}_{a,b}$ for $a + b \geq 2(K-k)$. Thus, by the triangle inequality, $\|\Pi_{\mathrm{succ}}\mathcal{T}\Pi_{\leq k}|\phi\rangle\| \leq \sum_{a+b \geq 2(K-k)} \|\Pi_{\mathrm{succ}}\mathcal{T}\tilde{\Pi}_{a,b}\Pi_{\leq k}|\phi\rangle\|$. This is at most $\sum_{a+b \geq 2(K-k)} \left(\sqrt{\frac{2K}{N}}\right)^{a+b} \|\tilde{\Pi}_{a,b}\Pi_{\leq k}|\phi\rangle\|$ by Equation (4). Finally, by the Cauchy-Schwarz inequality and the fact that the supports of the projectors $\tilde{\Pi}_{a,b}$ are disjoint, we obtain that $\|\Pi_{\mathrm{succ}}\mathcal{T}\Pi_{\leq k}|\phi\rangle\| \leq \sqrt{\sum_{a+b \geq 2(K-k)} \left(\frac{2K}{N}\right)^{a+b}} \sqrt{\sum_{a,b} \|\tilde{\Pi}_{a,b}\Pi_{\leq k}|\phi\rangle\|^2} \leq N\left(\frac{2K}{N}\right)^{K-k} \|\Pi_{\leq k}|\phi\rangle\|$. ◀

We can now conclude the proof of the main result of this section.

▶ **Theorem 9.** *The success probability of finding K disjoint collisions in a random function $f : [M] \to [N]$ is at most $O(T^3/(K^2N))^{K/2} + 2^{-K}$ for any algorithm making T quantum queries to f and any $1 \leq K \leq N/8$.*

Proof. Let $|\psi_T\rangle$ (resp. $|\phi_T\rangle$) denote the state of the algorithm after T queries in the standard (resp. recording) query model. We recall that $|\psi_T\rangle = \mathcal{T}|\phi_T\rangle$ (Theorem 3). Thus, by the triangle inequality, the success probability $\sigma = \|\Pi_{\mathrm{succ}}|\psi_T\rangle\|^2$ satisfies $\sqrt{\sigma} \leq \|\Pi_{\mathrm{succ}}\mathcal{T}\Pi_{\geq K/2}|\phi_T\rangle\| + \|\Pi_{\mathrm{succ}}\mathcal{T}\Pi_{\leq K/2}|\phi_T\rangle\| \leq \|\Pi_{\geq K/2}|\phi_T\rangle\| + \|\Pi_{\mathrm{succ}}\mathcal{T}\Pi_{\leq K/2}|\phi_T\rangle\|$. Using Proposition 7 and Proposition 8, we have that $\sqrt{\sigma} \leq \binom{T}{K/2}\left(4\sqrt{T/N}\right)^{K/2} + N(2K/N)^{K/2} \leq O(T^{3/2}/(K\sqrt{N}))^{K/2} + 2^{-K/2-1}$. Finally, the upper bound on σ is derived from the standard inequality $(u+v)^2 \leq 2u^2 + 2v^2$. ◀

5 Time-space tradeoff for Collision Pairs Finding

We use the time lower bound obtained in Section 4 to derive a new time-space tradeoff for the problem of finding K disjoint collisions in a random function $f : [M] \to [N]$. We recall that the output is produced in an online fashion (Section 2.2), meaning that a collision can be output as soon as it is discovered. The length of the output is not counted toward the space bound. We allow the same collision to be output several times, but it contributes only once to the total count.

▶ **Theorem 10.** *Any quantum algorithm for finding K disjoint collisions in a random function $f : [M] \to [N]$ with success probability $2/3$ must satisfy a time-space tradeoff of $T^3 S \geq \Omega(K^3 N)$, where $1 \leq K \leq N/8$.*

Proof. Our proof relies on the time-segmentation method for large-output problems used in [14, 29] for instance. Fix any quantum circuit $\mathcal{C}$ in the space-bounded model of Section 2.2 running in time T and using $S > \Omega(\log N)$ qubits of memory. The circuit $\mathcal{C}$ is partitioned into $L = T/T'$ consecutive sub-circuits $\mathcal{C}_1 \,\|\, \mathcal{C}_2 \,\|\, \cdots \,\|\, \mathcal{C}_L$ each running in time $T' = S^{2/3}N^{1/3}$, where $\mathcal{C}_j$ takes as input the output memory of $\mathcal{C}_{j-1}$ for each $j \in [L]$. Define X_j to be the random variable that counts the number of (mutually) disjoint collisions that $\mathcal{C}$ outputs between time $(j-1)T'$ and jT' (i.e. in the sub-circuit $\mathcal{C}_j$) when the input is a random function $f : [M] \to [N]$. The algorithm must satisfy $\sum_{j=1}^{L} \mathbb{E}[X_j] \geq \Omega(K)$ to be correct. We claim that the algorithm outputs at most $3S$ collisions in expectation in each segment of the computation. Assume by contradiction that $\mathbb{E}[X_j] \geq 3S$ for some j. Since X_j is bounded between 0 and N we have $\Pr[X_j > 2S] \geq S/N$. Consequently, by running $\mathcal{C}_j$ on the completely mixed state on S qubits we obtain $2S$ disjoint collisions with probability at least $S/N \cdot 2^{-S}$ in time T' (this is akin to a union-bound argument). However, by Theorem 9, no quantum algorithm can find more than $2S$ disjoint collisions in time $T' = S^{2/3}N^{1/3}$ with success probability larger than 4^{-S+1}. This contradiction implies that $\mathbb{E}[X_j] \leq 3S$ for all j. Consequently, there must be at least $L \geq \Omega(K/S)$ sub-circuits in order to have $\sum_{j=1}^{L} \mathbb{E}[X_j] \geq \Omega(K)$. Since each sub-circuit runs in time $S^{2/3}N^{1/3}$ the running time of $\mathcal{C}$ is $T \geq \Omega(L \cdot S^{2/3}N^{1/3}) \geq \Omega(KN^{1/3}/S^{1/3})$. ◀

As an illustration of the above result, we obtain that any quantum algorithm for finding $N/8$ disjoint collisions in a random function must satisfy a time-space tradeoff of $TS^{1/3} \geq \Omega(N^{4/3})$. We prove that any improvement to this lower bound would imply a breakthrough for the Element Distinctness problem.

▶ **Definition 11.** *The Element Distinctness problem* ED_N *on domain size N consists of finding a collision in a random function $f : [N] \to [N^2]$.*

It is well-known that the query complexity of Element Distinctness is $T = \Theta(N^{2/3})$ [2, 6]. However, it is a long-standing open problem to find any quantum time-space lower bound (even classically the question is not completely settled yet [39, 10]). Here, we show that *any* improvement to Theorem 10 would imply a non-trivial time-space tradeoff for Element Distinctness. This result relies on a reduction presented in Algorithm 1 and analyzed in Proposition 12 (the constants c_0, c_1, c_2 are chosen in the proof).

Algorithm 1 Finding collisions by using $\mathrm{ED}_{\sqrt{N}}$.

```
Input: a function $f : [N] \to [N]$ containing at least $c_0 N$ collisions.
Output: at least $c_1 N$ collisions in $f$ (not necessarily disjoint).

1. Repeat $c_2 N$ times:
   a. Sample a 4-wise independent hash function $h : [\sqrt{N}] \to [N]$ and store it in memory.
   b. Run an algorithm for $\mathrm{ED}_{\sqrt{N}}$ on input $f \circ h : [\sqrt{N}] \to [N]$. If it finds a collision
      $(f \circ h(i), f \circ h(j))$ check if $h(i) \neq h(j)$ and output the collision $(h(i), h(j))$ in this case.
```

▶ **Proposition 12.** *Let N be a square number. If there is an algorithm solving ED_N in time T_N and space S_N then Algorithm 1 runs in time $O(NT_{\sqrt{N}})$ and space $O\big(S_{\sqrt{N}}\big)$, and it finds $c_1 N$ collisions in any function $f : [N] \to [N]$ containing at least $c_0 N$ collisions.*

Proof. We choose $c_0 = 40$, $c_1 = 1/10^4$ and $c_2 = 8$. We study the probabilities of the following events to occur in a fixed round of Algorithm 1:

- **Event A:** The hash function h is collision free (i.e. $h(i) \neq h(j)$ for all $i \neq j$).
- **Event B:** None of the collisions output during the previous rounds is present in the image of h.
- **Event C:** The function $f \circ h : [\sqrt{N}] \to [N]$ contains a collision.
- **Event D:** The algorithm for $\mathrm{ED}_{\sqrt{N}}$ finds a collision at step 2.b.

Algorithm 1 succeeds if and only if the event $A \wedge B \wedge C \wedge D$ occurs during at least $c_1 N$ rounds. We now lower bound the probability of this event happening.

For **event A**, let us consider the random variable $X = \sum_{i \neq j \in [\sqrt{N}]} 1_{h(i)=h(j)}$. Using that h is pairwise independent, we have $\mathbb{E}[X] = \binom{\sqrt{N}}{2}\frac{1}{N} \leq \frac{1}{2}$. Thus, by Markov's inequality, $\Pr[A] = 1 - \Pr[X \geq 1] \geq \frac{1}{2}$.

For **event B**, let us assume that $k < c_1 N$ collisions $(x_1, x_2), \ldots, (x_{2k-1}, x_{2k})$ have been output so far. For any $i \in [k]$, the probability that both x_{2i-1} and x_{2i} belong to $\{h(1), \ldots, h(\sqrt{N})\}$ is at most $\binom{\sqrt{N}}{2}\frac{2}{N^2} \leq \frac{1}{N}$ since h is pairwise independent. By a union bound, $\Pr[B] \geq 1 - \frac{k}{N} \geq 1 - c_1$.

For **event C**, let us consider the binary random variables $Y_{i,j} = 1_{f \circ h(i) = f \circ h(j)}$ for $i \neq j \in [\sqrt{N}]$, and let $Y = \sum_{i \neq j} Y_{i,j}$ be twice the number of collisions in $f \circ h$. Note that we may have $Y_{i,j} = 1$ because $h(i) = h(j)$ (this is taken care of in event A). For each $y \in [N]$, let $N_y = |\{x : f(x) = y\}|$ denote the number of elements that are mapped to y by f. Using that h is 4-wise independent, for any $i \neq j \neq k \neq \ell$ we have,

$$\begin{cases} \Pr[Y_{i,j} = 1] = \dfrac{\sum_{y \in [N]} N_y^2}{N^2} \\[2ex] \Pr[Y_{i,j} = 1 \wedge Y_{i,k} = 1] = \dfrac{\sum_{y \in [N]} N_y^3}{N^3} \\[2ex] \Pr[Y_{i,j} = 1 \wedge Y_{k,\ell} = 1] = \Pr[Y_{i,j} = 1] \cdot \Pr[Y_{k,\ell} = 1]. \end{cases}$$

Consequently, $\mathbb{E}[Y] = \binom{\sqrt{N}}{2}\frac{\sum_{y\in[N]} N_y^2}{N^2}$ and

$$\begin{aligned}
\mathrm{Var}[Y] &= \sum_{\{i,j\}} \mathrm{Var}[Y_{i,j}] + \sum_{\{i,j\}\neq\{i,k\}} \mathrm{Cov}[Y_{i,j}, Y_{i,k}] + \sum_{\{i,j\}\cap\{k,\ell\}=\varnothing} \mathrm{Cov}[Y_{i,j}, Y_{k,\ell}] \\
&\leq \sum_{\{i,j\}} \mathbb{E}[Y_{i,j}^2] + \sum_{\{i,j\}\neq\{i,k\}} \mathbb{E}[Y_{i,j}Y_{i,k}] \\
&= \binom{\sqrt{N}}{2}\frac{\sum_{y\in[N]} N_y^2}{N^2} + 3\binom{\sqrt{N}}{3}\frac{\sum_{y\in[N]} N_y^3}{N^3}
\end{aligned}$$

where we have used that $Y_{i,j}$ and $Y_{k,\ell}$ are independent when $i \neq j \neq k \neq \ell$. The term $\sum_{y\in[N]} N_y^2$ is equal to the number of pairs $(x, x') \in [N]^2$ such that $f(x) = f(x')$. Each collision in f gives two such pairs, and we must also count the pairs (x, x). Thus, $\sum_{y\in[N]} N_y^2 \geq (1 + 2c_0)N$. Moreover, $\sum_{y\in[N]} N_y^3 \leq (\sum_{y\in[N]} N_y^2)^{3/2}$. Consequently,

$$\frac{\mathrm{Var}[Y]}{\mathbb{E}[Y]^2} \leq \frac{1 + \sqrt{N}\left(\frac{\sum_{y\in[N]} N_y^2}{N^2}\right)^{1/2}}{\binom{\sqrt{N}}{2}\frac{\sum_{y\in[N]} N_y^2}{N^2}} \leq \frac{4(1 + \sqrt{1 + 2c_0})}{1 + 2c_0}.$$

Finally, according to Chebyshev's inequality, $\Pr[Y = 0] \leq \Pr[|Y - \mathbb{E}[Y]| \geq \mathbb{E}[Y]] \leq \frac{\mathrm{Var}[Y]}{\mathbb{E}[Y]^2}$. Thus, $\Pr[C] = 1 - \Pr[Y = 0] \geq 1 - \frac{4(1+\sqrt{1+2c_0})}{1+2c_0}$.

For **event D**, we have $\Pr[D \mid A \wedge B \wedge C] \geq 2/3$ assuming the bounded-error algorithm for solving $\mathrm{ED}_{\sqrt{N}}$ succeeds with probability 2/3.

The probability of the four events happening together is $\Pr[A\wedge B\wedge C\wedge D] = \Pr[D|A\wedge B\wedge C]\cdot\Pr[A\wedge B\wedge C] \geq \Pr[D|A\wedge B\wedge C]\cdot(\Pr[A]+\Pr[B]+\Pr[C]-2) \geq \frac{2}{3}\cdot\left(\frac{1}{2} - c_1 - \frac{4(1+\sqrt{1+2c_0})}{1+2c_0}\right) \geq 1/250$. Let τ be the number of rounds after which c_1N collisions have been found (i.e. $A \wedge B \wedge C \wedge D$ has occurred c_1N times). We have $\mathbb{E}[\tau] \leq 8c_1N$, and by Markov's inequality $\Pr[\tau \geq c_2N] \leq 250c_1/c_2 \leq 1/3$. Thus, with probability at least 2/3, Algorithm 1 outputs at least c_1N collisions in f. ◀

We use the above reduction to transform any low-space algorithm for Element Distinctness into one for finding $\Omega(N/\log N)$ disjoint collisions in a random function. Observe that Algorithm 1 does not necessarily output collisions that are mutually disjoint. Nevertheless, there is a small probability that a random function $f : [M] \to [N]$ contains multi-collisions of size larger than $\log N$ when $M \approx N$ [24]. Thus, there is only a $\log N$ loss in the analysis.

▶ **Proposition 13.** *Suppose that there exists a bounded-error quantum algorithm for solving Element Distinctness on domain size N that satisfies a time-space tradeoff of $T^\alpha S^\beta \leq \widetilde{O}\big(N^{2(\gamma-\alpha)}\big)$ for some constants α, β, γ. Then, there exists a bounded-error quantum algorithm for finding $\Omega(N/\log N)$ disjoint collisions in a random function $f : [10N] \to [N]$ that satisfies a time-space tradeoff of $T^\alpha S^\beta \leq \widetilde{O}(N^\gamma)$.*

Proof. We use the constants c_0, c_1, c_2 specified in the proof of Proposition 12. First, we note that a random function $f : [10N] \to [N]$ contains c_0N collisions and no multi-collisions of size larger than $\log(N)$ with large probability [24]. Consequently, any set of c_1N collisions must contain at least $c_1N/\log N$ mutually disjoint collisions with large probability. Assume now that there exists an algorithm solving $\mathrm{ED}_{\sqrt{10N}}$ in time $T_{\sqrt{10N}}$ and space $S_{\sqrt{10N}}$ such that $\big(T_{\sqrt{10N}}\big)^\alpha S_{\sqrt{10N}}^\beta \leq \widetilde{\Omega}(N^{\gamma-\alpha})$. Then, by plugging it into Algorithm 1, one can find $c_1N/\log N$ disjoint collisions in a random function $f : [10N] \to [N]$ in time $T = O\big(NT_{\sqrt{10N}}\big)$ and space $S = O\big(S_{\sqrt{10N}}\big)$. We derive from the above tradeoff that $T^\alpha S^\beta \leq \widetilde{O}(N^\gamma)$. ◀

As an application of Proposition 13, we obtain the following result regarding the hardness of finding $\widetilde{\Omega}(N)$ collisions.

▶ **Corollary 14.** *Suppose that there exists $\epsilon > 0$ such that any quantum algorithm for finding $\widetilde{\Omega}(N)$ disjoint collisions in a random function $f : [10N] \to [N]$ must satisfy a time-space tradeoff of $TS^{1/3} \geq \widetilde{\Omega}(N^{4/3+\epsilon})$. Then, any quantum algorithm for solving Element Distinctness on domain size N must satisfy a time-space tradeoff of $TS^{1/3} \geq \widetilde{\Omega}(N^{2/3+2\epsilon})$.*

We conjecture that the optimal tradeoff for finding K collisions is $T^2S = \Theta(K^2N)$, which would imply an optimal time-space tradeoff of $T^2S \geq \widetilde{\Omega}(N^2)$ for Element Distinctness.

▶ **Conjecture 15.** *Any quantum algorithm for finding K disjoint collisions in a random function $f : [M] \to [N]$ with success probability $2/3$ must satisfy a time-space tradeoff of $T^2S \geq \Omega(K^2N)$.*

▶ **Corollary 16.** *If Conjecture 15 is true, then any quantum algorithm for solving the Element Distinctness problem with success probability $2/3$ must satisfy a time-space tradeoff of $T^2S \geq \widetilde{\Omega}(N^2)$.*

We describe a quantum algorithm that achieves the tradeoff of $T^2S \leq \widetilde{O}(K^2N)$. In order to simplify the analysis, we do not require the collisions to be disjoint.

Algorithm 2 Finding K collision pairs in $f : [N] \to [N]$ using a memory of size S.

```
1. Repeat $\widetilde{O}(K/S)$ times:
   a. Sample a subset $G \subset [N]$ of size $S$ uniformly at random.
   b. Construct a table containing all pairs $(x, f(x))$ for $x \in G$. Sort the table
      according to the second entry of each pair.
   c. Define the function $g : [N] \setminus G \to \{0,1\}$ where $g(x) = 1$ iff there exists $x' \in G$
      such that $f(x) = f(x')$. Run the Grover search algorithm [15] on $g$, by using
      the table computed at step 1.b, to find all pairs $(x, x') \in G \times ([N] \setminus G)$ such
      that $f(x) = f(x')$. Output all of these pairs.
```

▶ **Proposition 17.** *For any $1 \leq K \leq O(N)$ and $\widetilde{\Omega}(\log N) \leq S \leq \widetilde{O}(K^{2/3}N^{1/3})$, there exists a bounded-error quantum algorithm that can find K collisions in a random function $f : [N] \to [N]$ by making $T = \widetilde{O}(K\sqrt{N/S})$ queries and using S qubits of memory.*

Proof. We prove that Algorithm 2 satisfies the statement of the proposition. For simplicity, we do not try to tune the hidden factors in the big O notations.

The probability that a fixed pair (x, x') satisfies $(x, x') \in G \times ([N] \setminus G)$ for at least one iteration of step 1 is $\Omega(K/S \cdot S/N \cdot (1 - S/N)) = \Omega(K/N)$. Since a random function $f : [N] \to [N]$ contains $\Omega(N)$ collisions with high probability, the algorithm encounters $\Omega(K)$ collisions in total. Thus, if the Grover search algorithm never fails we obtain the desired number of collisions.

The expected number of pre-images of 1 under g is $O(S)$. Consequently, the complexity of Grover's search at step 1.c is $O(\sqrt{SN})$. The overall query complexity is $T = \widetilde{O}(K/S \cdot \sqrt{SN}) = \widetilde{O}(K\sqrt{N/S})$, and the space complexity is $\widetilde{O}(S)$. ◀

References

1 S. Aaronson. Limitations of quantum advice and one-way communication. *Theory of Computing*, 1(1):1–28, 2005.

2 S. Aaronson and Y. Shi. Quantum lower bounds for the collision and the element distinctness problems. *Journal of the ACM*, 51(4):595–605, 2004.

3 K. Abrahamson. A time-space tradeoff for boolean matrix multiplication. In *Proceedings of the 31st Symposium on Foundations of Computer Science (FOCS)*, pages 412–419, 1990.

4 G. Adj, D. Cervantes-Vázquez, J.-J. Chi-Domínguez, A. Menezes, and F. Rodríguez-Henríquez. On the cost of computing isogenies between supersingular elliptic curves. In *Proceedings of the 25th Conference on Selected Areas in Cryptography (SAC)*, pages 322–343, 2018.

5 A. Ambainis. Quantum lower bounds by quantum arguments. *Journal of Computer and System Sciences*, 64(4):750–767, 2002.

6 A. Ambainis. Quantum walk algorithm for element distinctness. *SIAM Journal on Computing*, 37(1):210–239, 2007.

7 A. Ambainis. A new quantum lower bound method, with an application to a strong direct product theorem for quantum search. *Theory of Computing*, 6(1):1–25, 2010.

8 A. Ambainis, R. Špalek, and R. de Wolf. A new quantum lower bound method, with applications to direct product theorems and time-space tradeoffs. *Algorithmica*, 55(3):422–461, 2009.

9 P. Beame. A general sequential time-space tradeoff for finding unique elements. *SIAM Journal on Computing*, 20(2):270–277, 1991.

10 P. Beame, M. Saks, X. Sun, and E. Vee. Time-space trade-off lower bounds for randomized computation of decision problems. *Journal of the ACM*, 50(2):154–195, 2003.

11 D. J. Bernstein. Understanding brute force, 2005. ECRYPT STVL Workshop on Symmetric Key Encryption.

12 D. J. Bernstein. Cost analysis of hash collisions: Will quantum computers make SHARCS obsolete? In *Proceedings of the 4th Workshop on Special-purpose Hardware for Attacking Cryptograhic Systems (SHARCS)*, pages 105–116, 2009.

13 A. Borodin, F. E. Fich, F. Meyer auf der Heide, E. Upfal, and A. Wigderson. A time-space tradeoff for element distinctness. *SIAM Journal on Computing*, 16(1):97–99, 1987.

14 A. Borodin, M. J. Fischer, D. G. Kirkpatrick, N. A. Lynch, and M. Tompa. A time-space tradeoff for sorting on non-oblivious machines. *Journal of Computer and System Sciences*, 22(3):351–364, 1981.

15 M. Boyer, G. Brassard, P. Høyer, and A. Tapp. Tight bounds on quantum searching. *Fortschritte der Physik*, 46(4-5):493–505, 1998.

16 G. Brassard, P. Høyer, and A. Tapp. Quantum cryptanalysis of hash and claw-free functions. In *Proceedings of the 3rd Latin American Symposium on Theoretical Informatics (LATIN)*, pages 163–169, 1998.

17 H. Buhrman and R. de Wolf. Complexity measures and decision tree complexity: A survey. *Theoretical Computer Science*, 288(1):21–43, 2002.

18 A. Chakrabarti and Y. Chen. Time-space tradeoffs for the memory game, 2017. `arXiv:1712.01330 [cs.CC]`.

19 A. Chiesa, P. Manohar, and N. Spooner. Succinct arguments in the quantum random oracle model. In *Proceedings of the 17th Conference on Theory of Cryptography (TCC)*, pages 1–29, 2019.

20 J. Czajkowski, C. Majenz, C. Schaffner, and S. Zur. Quantum lazy sampling and game-playing proofs for quantum indifferentiability, 2019. `arXiv:1904.11477v1 [quant-ph]`.

21 C. Delaplace, A. Esser, and A. May. Improved low-memory subset sum and LPN algorithms via multiple collisions. In *Proceedings of the 17th IMA International Conference on Cryptography and Coding (IMACC)*, pages 178–199, 2019.

22 I. Dinur. Tight time-space lower bounds for finding multiple collision pairs and their applications. In *Proceedings of the 39th International Conference on the Theory and Applications of Cryptographic Techniques (EUROCRYPT)*, pages 405–434, 2020.

23 I. Dinur, O. Dunkelman, N. Keller, and A. Shamir. Efficient dissection of composite problems, with applications to cryptanalysis, knapsacks, and combinatorial search problems. In *Proceedings of the 32th International Cryptology Conference (CRYPTO)*, pages 719–740, 2012.

24 P. Flajolet and A. M. Odlyzko. Random mapping statistics. In *Proceedings of the 7th Workshop on the Theory and Application of Cryptographic Techniques (EUROCRYPT)*, pages 329–354, 1989.

25 A. Ghoshal, J. Jaeger, and S. Tessaro. The memory-tightness of authenticated encryption. In *Proceedings of the 40th International Cryptology Conference (CRYPTO)*, pages 127–156, 2020.

26 A. Hosoyamada and T. Iwata. 4-round Luby-Rackoff construction is a qPRP. In *Proceedings of the 25th International Conference on the Theory and Applications of Cryptology and Information Security (ASIACRYPT)*, pages 145–174, 2019.

27 J. Jaeger and S. Tessaro. Tight time-memory trade-offs for symmetric encryption. In *Proceedings of the 38th International Conference on the Theory and Applications of Cryptographic Techniques (EUROCRYPT)*, pages 467–497, 2019.

28 A. Joux and S. Lucks. Improved generic algorithms for 3-collisions. In *Proceedings of the 15th International Conference on the Theory and Applications of Cryptology and Information Security (ASIACRYPT)*, pages 347–363, 2009.

29 H. Klauck, R. Špalek, and R. de Wolf. Quantum and classical strong direct product theorems and optimal time-space tradeoffs. *SIAM Journal on Computing*, 36(5):1472–1493, 2007.

30 Q. Liu and M. Zhandry. On finding quantum multi-collisions. In *Proceedings of the 38th International Conference on the Theory and Applications of Cryptographic Techniques (EUROCRYPT)*, pages 189–218, 2019.

31 Y. Mansour, N. Nisan, and P. Tiwari. The computational complexity of universal hashing. *Theoretical Computer Science*, 107(1):121–133, 1993.

32 P. C. van Oorschot and M. J. Wiener. Parallel collision search with cryptanalytic applications. *Journal of Cryptology*, 12(1):1–28, 1999.

33 J. M. Pollard. A Monte Carlo method for factorization. *BIT Numerical Mathematics*, 15(3):331–334, 1975.

34 R. Špalek. The multiplicative quantum adversary. In *Proceedings of the 23rd Computational Complexity Conference (CCC)*, pages 237–248, 2008.

35 S. Tessaro and A. Thiruvengadam. Provable time-memory trade-offs: Symmetric cryptography against memory-bounded adversaries. In *Proceedings of the 16th Conference on Theory of Cryptography (TCC)*, pages 3–32, 2018.

36 C. van Vredendaal. Reduced memory meet-in-the-middle attack against the NTRU private key. *LMS Journal of Computation and Mathematics*, 19(A):43–57, 2016.

37 D. Wagner. A generalized birthday problem. In *Proceedings of the 22nd International Cryptology Conference (CRYPTO)*, pages 288–304, 2002.

38 M. J. Wiener. The full cost of cryptanalytic attacks. *Journal of Cryptology*, 17(2):105–124, 2004.

39 A. C.-C. Yao. Near-optimal time-space tradeoff for element distinctness. *SIAM Journal on Computing*, 23(5):966–975, 1994.

40 M. Zhandry. A note on the quantum collision and set equality problems. *Quantum Information & Computation*, 15(7&8):557–567, 2015.

41 M. Zhandry. How to record quantum queries, and applications to quantum indifferentiability. In *Proceedings of the 39th International Cryptology Conference (CRYPTO)*, pages 239–268, 2019.

A Time lower bound for K-Search

In this section, we illustrate the use of the recording model to upper bound the success probability of a query-bounded algorithm on a *non-uniform* input distribution.

▶ **Theorem 18.** *The success probability of finding $K \leq N/8$ preimages of 1 in a random function $f : [M] \to \{0,1\}$ where $f(x) = 1$ with probability K/N for each $x \in [M]$ is at most $O(T^2/(KN))^{K/2} + 2^{-K}$ for any algorithm using T quantum queries to f.*

We show that, similarly to the classical setting where a query can reveal a 1 with probability K/N, the *amplitude* of the basis states that record a new 1 increases by a factor of $\sqrt{K/N}$ after each query (Proposition 22). Thus, the amplitude of the basis states that have recorded at least $K/2$ ones after T queries is at most $O(T/\sqrt{KN})^{K/2}$. This implies that any algorithm with $T < O(\sqrt{KN})$ queries is likely to output at least $K/2$ ones at positions that have not been recorded. These outputs can only be correct with probability $O(K/N)^{K/2}$ (Proposition 23).

A.1 Recording query operator

We describe a recording operator that encodes the distribution that gives $f : [M] \to [N]$ where $f(x) = 1$ with probability K/N independently for each $x \in [M]$. In the standard query model, the oracle's initial state is $|\mathrm{init}\rangle = \otimes_{x\in[M]}\big(\sqrt{1-K/N}|0\rangle_{\mathcal{F}_x} + \sqrt{K/N}|1\rangle_{\mathcal{F}_x}\big)$ for this distribution. Consequently, we instantiate the recording model as follows.

▶ **Definition 19.** *For any $x \in [M]$, define the unitary $\mathcal{S}_x$ acting on the register $\mathcal{F}_x$ to be*

$$\mathcal{S}_x|\bot\rangle_{\mathcal{F}_x} = |+\rangle_{\mathcal{F}_x}, \qquad \mathcal{S}_x|+\rangle_{\mathcal{F}_x} = |\bot\rangle_{\mathcal{F}_x}, \qquad \mathcal{S}_x|-\rangle_{\mathcal{F}_x} = |-\rangle_{\mathcal{F}_x}$$

where $\alpha = \sqrt{1-K/N}$, $\beta = \sqrt{K/N}$ and $|+\rangle_{\mathcal{F}_x} = \alpha|0\rangle_{\mathcal{F}_x} + \beta|1\rangle_{\mathcal{F}_x}$, $|-\rangle_{\mathcal{F}_x} = \beta|0\rangle_{\mathcal{F}_x} - \alpha|1\rangle_{\mathcal{F}_x}$.

We have $\mathcal{T}|\bot^M\rangle = |\mathrm{init}\rangle$ when $\mathcal{T} = \otimes_{x\in[M]}\mathcal{S}_x$ as required by Theorem 3. The recording query operator is $\mathcal{R} = \mathcal{S}\mathcal{O}\mathcal{S}$ since $\mathcal{S}^\dagger = \mathcal{S}$, and it satisfies the next equations.

▶ **Lemma 20.** *If the recording query operator $\mathcal{R}$ associated with Definition 19 is applied to a basis state $|x,p,w\rangle|f\rangle$ where $p = 1$ then the register $|f(x)\rangle_{\mathcal{F}_x}$ is mapped to*

$$\begin{cases} (1-2\beta^2)|\bot\rangle + 2\alpha\beta^2|0\rangle - 2\alpha^2\beta|1\rangle & \text{if } f(x) = \bot \\ 2\alpha\beta^2|\bot\rangle + (1-2\alpha^2\beta^2)|0\rangle + 2\alpha^3\beta|1\rangle & \text{if } f(x) = 0 \\ -2\alpha^2\beta|\bot\rangle + 2\alpha^3\beta|0\rangle + (1-2\alpha^4)|1\rangle & \text{if } f(x) = 1 \end{cases}$$

and the other registers are unchanged. If $p = 0$ then none of the registers are changed.

Proof. By definition, the unitary $\mathcal{S}_x$ maps $|\bot\rangle_{\mathcal{F}_x} \mapsto |+\rangle$, $|0\rangle_{\mathcal{F}_x} \mapsto \alpha|\bot\rangle + \beta|-\rangle$, $|1\rangle_{\mathcal{F}_x} \mapsto \beta|\bot\rangle - \alpha|-\rangle$. Thus, the action on the register $\mathcal{F}_x$ is

- If $f(x) = \bot$ then $|f(x)\rangle_{\mathcal{F}_x} \overset{\mathcal{S}}{\mapsto} |+\rangle \overset{\mathcal{O}}{\mapsto} \alpha|0\rangle - \beta|1\rangle \overset{\mathcal{S}}{\mapsto} (\alpha^2-\beta^2)|\bot\rangle + 2\alpha\beta|-\rangle$.
- If $f(x) = 0$ then $|f(x)\rangle_{\mathcal{F}_x} \overset{\mathcal{S}}{\mapsto} \alpha|\bot\rangle + \beta|-\rangle \overset{\mathcal{O}}{\mapsto} \alpha|\bot\rangle + \beta(\beta|0\rangle + \alpha|1\rangle) \overset{\mathcal{S}}{\mapsto} 2\alpha\beta^2|\bot\rangle + (1 - 2\alpha^2\beta^2)|0\rangle + 2\alpha^3\beta|1\rangle$.
- If $f(x) = 1$ then $|f(x)\rangle_{\mathcal{F}_x} \overset{\mathcal{S}}{\mapsto} \beta|\bot\rangle - \alpha|-\rangle \overset{\mathcal{O}}{\mapsto} \beta|\bot\rangle - \beta(\beta|0\rangle + \alpha|1\rangle) \overset{\mathcal{S}}{\mapsto} -2\alpha^2\beta|\bot\rangle + 2\alpha^3\beta|0\rangle + (1-2\alpha^4)|1\rangle$.

◀

If $\alpha \gg \beta$, the above lemma shows that $\mathcal{R}$ is close to the mapping $|\bot\rangle_{\mathcal{F}_x} \mapsto |\bot\rangle - 2\beta|1\rangle$, $|0\rangle_{\mathcal{F}_x} \mapsto |0\rangle + 2\beta|1\rangle$, $|1\rangle_{\mathcal{F}_x} \mapsto -|1\rangle + 2\beta(|0\rangle - |\bot\rangle)$ up to lower order terms of amplitude $O(\beta^2)$.

A.2 Analysis of the recording progress

The measure of progress is based on the number of ones contained in the oracle's register. We first give some projectors related to this quantity.

▶ **Definition 21.** *We define the following projectors by giving the basis states on which they project:*

- $\Pi_{\leq k}$, $\Pi_{=k}$ *and* $\Pi_{\geq k}$*: all basis states* $|x,p,w\rangle|f\rangle$ *such that* f *contains respectively* at most, exactly *or* at least k *coordinates equal to* 1*.*
- $\Pi_{=k,\perp}$ *and* $\Pi_{=k,0}$*: all basis states* $|x,p,w\rangle|f\rangle$ *such that (1)* f *contains* exactly k *coordinates equal to* 1*, (2) the phase multiplier is* $p=1$ *and (3)* $f(x)=\perp$ *or* $f(x)=0$ *respectively.*

We can now define the measure of progress $q_{t,k}$ for t queries and k ones as

$$q_{t,k} = \|\Pi_{\geq k}|\phi_t\rangle\| \tag{5}$$

where $|\phi_t\rangle$ is the state after t queries in the recording model. The main result of this section is the following bound on the growth of $q_{t,k}$.

▶ **Proposition 22.** *For all* t *and* k*, we have that* $q_{t,k} \leq \binom{t}{k}\left(\frac{4\sqrt{K}}{\sqrt{N}}\right)^k$*.*

Proof. First, $q_{0,0}=1$ and $q_{0,k}=0$ for all $k\geq 1$ since the initial state is $|\phi_0\rangle = |0\rangle|\perp^M\rangle$. Then, we prove that $q_{t,k}$ satisfies the following recurrence relation

$$q_{t+1,k+1} \leq q_{t,k+1} + 4\sqrt{\frac{K}{N}}q_{t,k}. \tag{6}$$

From this result, it is trivial to conclude that $q_{t,k} \leq \binom{t}{k}\left(\frac{4\sqrt{K}}{\sqrt{N}}\right)^k$. In order to prove Equation (6), we first observe that $q_{t+1,k+1} = \|\Pi_{\geq k+1}U_{t+1}\mathcal{R}|\phi_t\rangle\| = \|\Pi_{\geq k+1}\mathcal{R}|\phi_t\rangle\|$ where U_{t+1} is the unitary applied by the algorithm at time $t+1$. Then, on a basis state $|x,p,w\rangle|f\rangle$, the recording query operator $\mathcal{R}$ acts as the identity on the registers $\mathcal{F}_{x'}$ for $x'\neq x$. Consequently, the basis states $|x,p,w\rangle|f\rangle$ in $|\phi_t\rangle$ that may contribute to $q_{t+1,k+1}$ must either already contain $k+1$ ones in f, or exactly k ones in f and $f(x)\neq 1$, $p=1$. This implies that

$$q_{t+1,k+1} \leq q_{t,k+1} + \|\Pi_{\geq k+1}\mathcal{R}\Pi_{=k,\perp}|\phi_t\rangle\| + \|\Pi_{\geq k+1}\mathcal{R}\Pi_{=k,0}|\phi_t\rangle\|.$$

We first bound the term $\|\Pi_{\geq k+1}\mathcal{R}\Pi_{=k,\perp}|\phi_t\rangle\|$. Consider any state $|x,p,w\rangle|f\rangle$ in the support of $\Pi_{=k,\perp}$. By Lemma 20, we have $\Pi_{\geq k+1}\mathcal{R}|x,p,w\rangle|f\rangle = -2\alpha^2\beta|x,p,w\rangle|1\rangle_{\mathcal{F}_x} \otimes_{x'\neq x}|f(x')\rangle_{\mathcal{F}_{x'}}$. Since any two basis states in the support of $\Pi_{=k,\perp}$ remain orthogonal after $\Pi_{\geq k+1}\mathcal{R}$ is applied, we obtain that $\|\Pi_{\geq k+1}\mathcal{R}\Pi_{=k,\perp}|\phi_t\rangle\| = 2\alpha^2\beta\|\Pi_{=k,\perp}|\phi_t\rangle\| \leq 2\sqrt{K/N}(1-K/N)q_{t,k}$.

Similarly, for $|x,p,w\rangle|f\rangle$ in the support of $\Pi_{=k,0}$ we have $\|\Pi_{\geq k+1}\mathcal{R}|x,p,w\rangle|f\rangle\| = 2\alpha^3\beta$ by Lemma 20. Consequently, $\|\Pi_{\geq k+1}\mathcal{R}\Pi_{=k,0}|\phi_t\rangle\| = 2\alpha^3\beta\|\Pi_{=k,0}|\phi_t\rangle\| \leq 2\sqrt{K/N}(1-K/N)^{3/2}q_{t,k}$. We can now conclude the proof,

$$q_{t+1,k+1} \leq q_{t,k+1} + 2\sqrt{\frac{K}{N}}\left(1-\frac{K}{N}\right)q_{t,k} + 2\sqrt{\frac{K}{N}}\left(1-\frac{K}{N}\right)^{3/2}q_{t,k} \leq q_{t,k+1} + 4\sqrt{\frac{K}{N}}q_{t,k}.$$

◀

A.3 From the recording progress to the success probability

We connect the success probability $\sigma = \|\Pi_{\mathrm{succ}}|\psi_T\rangle\|^2$ in the standard query model to the final progress $q_{T,k}$ in the recording model after T queries. We show that if the algorithm has made no significant progress for $k \geq K/2$ then it needs to "guess" that $f(x) = 1$ for about $K - k$ positions where the $\mathcal{F}_x$ register does not contain 1. Classically, the probability to find $K - k$ preimages of 1 at positions that have not been queried would be $(K/N)^{K-k}$. Here, we show similarly that if a unit state contains at most k ones in the quantum recording model, then after mapping it to the standard query model (by applying the operator $\mathcal{T}$ of Theorem 3) the probability that the output register contains the correct positions of K preimages of 1 is at most $3^K\left(\frac{K}{N}\right)^{K-k}$.

▶ **Proposition 23.** *For any $|\phi\rangle$, we have $\|\Pi_{\mathrm{succ}}\mathcal{T}\Pi_{\leq k}|\phi\rangle\| \leq 3^{K/2}\left(\sqrt{\frac{K}{N}}\right)^{K-k}\|\Pi_{\leq k}|\phi\rangle\|$.*

Proof. Let $|x,p,w\rangle|f\rangle$ be any basis state in the support of $\Pi_{\leq k}$. The output value z is a substring of w made of K distinct values $x_1,\ldots,x_K \in [M]$ indicating positions where the input f is supposed to contain ones. By definition of $\Pi_{\leq k}$, we have $f(x_i) \neq 1$ for at least $K-k$ indices $i \in [K]$. For each such index i, after applying $\mathcal{T} = \otimes_{x' \in [M]}\mathcal{S}_{x'}$, the amplitude of $|1\rangle_{\mathcal{F}_{x_i}}$ is $\sqrt{\frac{K}{N}}$ (if $f(x_i) = \perp$) or $\sqrt{\frac{K}{N}\left(1-\frac{K}{N}\right)}$ (if $f(x_i) = 0$) by Definition 19. Consequently,

$$\|\Pi_{\mathrm{succ}}\mathcal{T}|x,p,w\rangle|f\rangle\| \leq \left(\sqrt{\frac{K}{N}}\right)^{K-k}. \tag{7}$$

Fix any state $|\phi\rangle$ and denote $|\varphi\rangle = \Pi_{\leq k}|\phi\rangle = \sum_{x,p,w,f}\alpha_{x,p,w,f}|x,p,w\rangle|f\rangle$. Let us write $w_{\vec{x}} = \{x_1,\ldots,x_K\}$ when the output substring z of w contains $x_1,\ldots,x_K$. For any two basis states $|x,p,w\rangle|f\rangle$ and $|\bar{x},\bar{p},\bar{w}\rangle|\bar{f}\rangle$, if $\left(x,p,w,(f(x'))_{x'\notin w_{\vec{x}}}\right) \neq \left(\bar{x},\bar{p},\bar{w},(\bar{f}(x'))_{x'\notin w_{\vec{x}}}\right)$ then $\Pi_{\mathrm{succ}}\mathcal{T}|x,p,w\rangle|f\rangle$ is orthogonal to $\Pi_{\mathrm{succ}}\mathcal{T}|\bar{x},\bar{p},\bar{w}\rangle|\bar{f}\rangle$. There are 3^K choices for $|x,p,w\rangle|f\rangle$ once we set the value of $(x,p,w,(f(x'))_{x'\notin w_{\vec{x}}})$ since it remains to choose $f(x') \in \{\perp,0,1\}$ for $x' \in w_{\vec{x}}$. By using the Cauchy–Schwarz inequality and Equation (7), we get that

$$\begin{aligned}
\|\Pi_{\mathrm{succ}}\mathcal{T}|\varphi\rangle\|^2 &= \sum_{x,p,w,(f(x'))_{x'\notin w_{\vec{x}}}}\left\|\sum_{(f(x'))_{x'\in w_{\vec{x}}}}\alpha_{x,p,w,f}\Pi_{\mathrm{succ}}\mathcal{T}|x,p,w\rangle|f\rangle\right\|^2 \\
&\leq \sum_{x,p,w,(f(x'))_{x'\notin w_{\vec{x}}}}\left(\sum_{(f(x'))_{x'\in w_{\vec{x}}}}|\alpha_{x,p,w,f}|^2\right)\left(\sum_{(f(x'))_{x'\in w_{\vec{x}}}}\|\Pi_{\mathrm{succ}}\mathcal{T}|x,p,w\rangle|f\rangle\|^2\right) \\
&\leq \||\varphi\rangle\|^2\cdot 3^K\left(\frac{K}{N}\right)^{K-k}.
\end{aligned}$$

◀

We can now conclude the proof of the main result.

Proof of Theorem 18. Let $|\psi_T\rangle$ (resp. $|\phi_T\rangle$) denote the state of the algorithm after T queries in the standard (resp. recording) query model. According to Theorem 3, we have $|\psi_T\rangle = \mathcal{T}|\phi_T\rangle$. Thus, by the triangle inequality, the success probability $\sigma = \|\Pi_{\mathrm{succ}}|\psi_T\rangle\|^2$ satisfies $\sqrt{\sigma} \leq \|\Pi_{\mathrm{succ}}\mathcal{T}\Pi_{\geq K/2}|\phi_T\rangle\| + \|\Pi_{\mathrm{succ}}\mathcal{T}\Pi_{\leq K/2}|\phi_T\rangle\| \leq \|\Pi_{\geq K/2}|\phi_T\rangle\| + \|\Pi_{\mathrm{succ}}\mathcal{T}\Pi_{\leq K/2}|\phi_T\rangle\|$. Using Propositions 22 and 23, we have that $\sqrt{\sigma} \leq \binom{T}{K/2}\left(4\sqrt{K/N}\right)^{K/2} + 3^{K/2}\left(\sqrt{K/N}\right)^{K/2} \leq O(T/\sqrt{KN})^{K/2} + 2^{-K/2-1}$. Finally, the upper bound on σ is derived from the standard inequality $(u+v)^2 \leq 2u^2 + 2v^2$. ◀

B Time-space tradeoff for Sorting

We use the time lower bound obtained in Appendix A to reprove the time-space tradeoff for the Sorting problem described in [29, Theorem 21]. The input to the Sorting problem is represented as a function $f : [N] \to \{0,1,2\}$ (we do not need to consider a larger range for the proof). A quantum algorithm for the Sorting problem must output in order a sequence $x_1, \ldots, x_N \in [N]$ of distinct integers such that $f(x_1) \geq f(x_2) \geq \cdots \geq f(x_N)$ with probability at least $2/3$.

▶ **Theorem 24.** *Any quantum algorithm for sorting a function $f : [N] \to \{0,1,2\}$ with success probability $2/3$ must satisfy a time-space tradeoff of $T^2 S \geq \Omega(N^3)$.*

Proof. The proof is a modified version of [29, Theorem 21] adapted to our version of the K-Search problem. Given a circuit $\mathcal{C}$ that runs in time T and space $\Omega(\log N) \leq S \leq N/64$, we partition it into $L = T/T'$ consecutive sub-circuits $\mathcal{C}_1 \parallel \mathcal{C}_2 \parallel \cdots \parallel \mathcal{C}_L$ each running in time $T' = \sqrt{SN}/4$. Assume by contradiction that a circuit $\mathcal{C}_j$ outputs the elements of ranks $r, r+1, \ldots, r+2S-1$ for some $r \leq N/2$. We use $\mathcal{C}_j$ to solve the K-search problem for $K = 2S$ as follows. Given an input $g : [N/2] \to \{0,1\}$ to the K-search problem where $g(x) = 1$ with probability $\frac{K}{N/4}$ for each x, define the function $f : [N] \to \{0,1,2\}$ where

$$f(x) = \begin{cases} 2 & \text{if } x < r, \\ g(x - r + 1) & \text{if } r \leq x < r + N/2, \\ 0 & \text{if } x \geq r + N/2. \end{cases}$$

Note that the function g contains at least $2S$ preimages of 1 with probability at least $2S/N$. Thus, if the circuit $\mathcal{C}$ is run on the input f, then the indices output by the sub-circuit $\mathcal{C}_j$ must contain the position of $2S$ preimages of 1 with probability at least $2/3 \cdot 2S/N$. Consequently, by running $\mathcal{C}_j$ on the completely mixed state on S qubits we can find $2S$ preimages of 1 under g with probability at least $2/3 \cdot 2S/N \cdot 2^{-S}$ in time T'. However, by Theorem 18, any such algorithm must succeed with probability at most 4^{-S+1}. This contradiction implies that there must be at least $L \geq \Omega(N/S)$ sub-circuits in $\mathcal{C}$. Thus, the running time of $\mathcal{C}$ is $T \geq \Omega(L \cdot \sqrt{SN}) \geq \Omega(N^{3/2}/\sqrt{S})$. ◀

The time-space tradeoffs for the Boolean matrix-vector product [29, Theorem 23] and the Boolean matrix product [29, Theorem 25] problems can be reproved in a similar way.

Quantum Pseudorandomness and Classical Complexity

William Kretschmer
University of Texas at Austin, TX, USA

Abstract

We construct a quantum oracle relative to which $\mathsf{BQP} = \mathsf{QMA}$ but cryptographic pseudorandom quantum states and pseudorandom unitary transformations exist, a counterintuitive result in light of the fact that pseudorandom states can be "broken" by quantum Merlin-Arthur adversaries. We explain how this nuance arises as the result of a distinction between algorithms that operate on quantum and classical inputs. On the other hand, we show that *some* computational complexity assumption is needed to construct pseudorandom states, by proving that pseudorandom states do not exist if $\mathsf{BQP} = \mathsf{PP}$. We discuss implications of these results for cryptography, complexity theory, and quantum tomography.

2012 ACM Subject Classification Theory of computation → Quantum complexity theory

Keywords and phrases pseudorandom quantum states, quantum Merlin-Arthur

Digital Object Identifier 10.4230/LIPIcs.TQC.2021.2

Related Version *Previous Version*: https://arxiv.org/abs/2103.09320

Funding *William Kretschmer*: Supported by a National Defense Science and Engineering Graduate (NDSEG) Fellowship from the US Department of Defense.

Acknowledgements Thanks to Scott Aaronson for suggestions on the writing, Adam Bouland for insightful discussions, and Qipeng Liu for clarifying some questions about [16].

1 Introduction

Pseudorandomness is a key concept in complexity theory and cryptography, capturing the notion of objects that appear random to computationally-bounded adversaries. Recent works have extended the theory of computational pseudorandomness to quantum objects, with a particular focus on quantum states and unitary transformations that resemble the Haar measure [19, 13, 12].

Ji, Liu, and Song [19] define a *pseudorandom state* (PRS) ensemble as a keyed family of quantum states $\{|\varphi_k\rangle\}_{k\in\mathcal{K}}$ such that states from the ensemble can be generated in polynomial time, and such that no polynomial-time quantum adversary can distinguish polynomially many copies of a random $|\varphi_k\rangle$ from polynomially many copies of a Haar-random state. They also define an ensemble of *pseudorandom unitary transformations* (PRUs) analogously as a set of efficiently implementable unitary transformations that are computationally indistinguishable from the Haar measure. These definitions can be viewed as quantum analogues of pseudorandom generators (PRGs) and pseudorandom functions (PRFs), respectively. The authors then present a construction of PRSs assuming the existence of quantum-secure one-way functions, and also give a candidate construction of PRUs that they conjecture is secure.

Several applications of PRSs and PRUs are known. PRSs and PRUs are potentially useful in quantum algorithms: in computational applications that require approximations to the Haar measure, PRSs and PRUs can be much more efficient than t-designs, which are information-theoretic approximations to the Haar measure that are analogous to t-

16th Conference on the Theory of Quantum Computation, Communication and Cryptography (TQC 2021).
Editor: Min-Hsiu Hsieh; Article No. 2; pp. 2:1–2:20
Leibniz International Proceedings in Informatics
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

wise independent functions.[1] Cryptographic applications are possible, with [19] giving a construction of a private-key quantum money scheme based on PRSs. Recent work by Bouland, Fefferman, and Vazirani [12] has also established a fundamental connection between PRSs and any possible resolution to the so-called "wormhole growth paradox" in the AdS/CFT correspondence.

1.1 Main Results

Given the importance of PRSs and PRUs across quantum complexity theory, in this work we seek to better understand the theoretical basis for the existence of these primitives. We start with a very basic question: what hardness assumptions are necessary for the existence of PRSs,[2] and which unlikely complexity collapses (such as $\mathsf{P} = \mathsf{PSPACE}$ or $\mathsf{BQP} = \mathsf{QMA}$) would invalidate the security of PRSs? Viewed another way, we ask: what computational power suffices to distinguish PRSs from Haar-random states?

At first glance, it appears that an "obvious" upper bound on the power needed to break PRSs is QMA, the quantum analogue of NP consisting of problems decidable by a polynomial-time quantum Merlin-Arthur protocol (or even QCMA, where the witness is restricted to be classical). If Arthur holds many copies of a pure quantum state $|\psi\rangle$ that can be prepared by some polynomial-size quantum circuit C, then Merlin can send Arthur a classical description of C, and Arthur can verify via the swap test that the output of C approximates $|\psi\rangle$. By contrast, most Haar-random states cannot even be approximated by small quantum circuits. So, in some sense, PRSs can be "distinguished" from Haar-random by quantum Merlin-Arthur adversaries.

There is a subtle problem here, though: QMA is defined as a set of decision problems where the inputs are *classical* bit strings, whereas an adversary against a PRS ensemble inherently operates on a *quantum* input. As a result, it is unclear whether the hardness of breaking PRSs can be related to the hardness of QMA, or any other standard complexity class. Even if we had a proof that $\mathsf{BQP} = \mathsf{QMA}$, this might not give rise to an efficient algorithm for breaking the security of PRSs.

One way to tackle this is to consider quantum adversaries that can query a classical oracle. If we can show that PRSs can be broken by a polynomial-time quantum algorithm with oracle access to some language $\mathcal{L} \subseteq \{0,1\}^*$, we conclude that if PRSs exist, then $\mathcal{L} \notin \mathsf{BQP}$. A priori, it is not immediately obvious whether oracle access to *any* language $\mathcal{L}$ suffices for a polynomial-time quantum adversary to break PRSs. For our first result, we show that a PP-complete language works. Hence, if $\mathsf{BQP} = \mathsf{PP}$, then PRSs do not exist.

▶ **Theorem 1** (Informal version of Theorem 15). *There exists a polynomial-time quantum algorithm augmented with a* PP *oracle that can distinguish PRSs from Haar-random states.*

This raises the natural question of whether the PP oracle in the above theorem can be made weaker. For instance, can we break PRSs with a QCMA or QMA oracle, coinciding with our intuition that the task is solvable by a quantum Merlin-Arthur protocol? In our second result, we show that this intuition is perhaps misguided, as we construct a quantum oracle relative to which such a QMA reduction is impossible.

[1] t-designs are also sometimes called "pseudorandom" in the literature, e.g. [27, 14]. We emphasize that t-designs and PRSs/PRUs are fundamentally different notions and that they are generally incomparable: a t-design need not be a PRS/PRU ensemble, or vice-versa.

[2] Note that PRUs imply PRSs, so we focus only on PRSs for this part.

▶ **Theorem 2** (Informal version of Theorem 18 and Theorem 21). *There exists a quantum oracle $\mathcal{O}$ such that:*

(1) $\mathsf{BQP}^{\mathcal{O}} = \mathsf{QMA}^{\mathcal{O}}$*, and*

(2) *PRUs (and hence PRSs) exist relative to $\mathcal{O}$.*

Let us remark how bizarre this theorem appears from a cryptographer's point of view. If $\mathsf{BQP} = \mathsf{QMA}$, then *no* quantum-secure classical cryptographic primitives exist, because such primitives can be broken in NP. So, our construction is a black-box separation between PRUs and *all* quantum-secure classical cryptography – a relativized world in which any computationally-secure cryptography must use quantum communication. Theorem 2 thus provides a negative answer (in the quantum black box setting) to a question of Ji, Liu, and Song [19] that asks if quantum-secure one-way functions are necessary for PRSs. One could even view our result as evidence that it might be possible to base the existence of PRSs and PRUs on weaker assumptions than those usually used for classical cryptography.

1.2 Application: Hyperefficient Shadow Tomography

An immediate corollary of our results is a new impossibility result for shadow tomography. Aaronson [2] defined the shadow tomography problem as the following estimation task: given copies of an n-qubit mixed state ρ and a list of two-outcome measurements $O_1, \ldots, O_M$, estimate $\mathrm{Tr}(O_i \rho)$ for each i up to additive error ε. Aaronson showed that, remarkably, this is possible using very few copies of ρ: just $\mathrm{poly}(n, \log M, \frac{1}{\varepsilon})$ copies suffice, which is polylogarithmic in both the dimension of ρ and the number of quantities to be estimated.

Aaronson then asked in what cases shadow tomography can be made *computationally* efficient with respect to n and $\log M$. Of course, just writing down the input to the problem would take $\Omega(4^n M)$ time if the measurements are given explicitly as Hermitian matrices, and listing the outputs would also take $\Omega(M)$ time. But perhaps one could hope for an algorithm that only operates *implicitly* on both the inputs and outputs. For example, suppose we stipulate the existence of a quantum algorithm that performs the measurement O_i given input $i \in [M]$, and that this algorithm runs in time $\mathrm{poly}(n, \log M)$. Consider a shadow tomography procedure that takes a description of such an algorithm as input, and that outputs a quantum circuit C such that $|C(i) - \mathrm{Tr}(O_i \rho)| \le \varepsilon$ for each $i \in [M]$.[3] Aaronson calls this a "hyperefficient" shadow tomography protocol if it additionally runs in time $\mathrm{poly}(n, \log M, \frac{1}{\varepsilon})$.

Aaronson gave some evidence that hyperefficient shadow tomography is unlikely to exist, by observing that if hyperefficient shadow tomography is possible, then quantum advice can always be efficiently replaced by classical advice – in other words, $\mathsf{BQP/qpoly} = \mathsf{BQP/poly}$. However, Aaronson and Kuperberg [4] showed a quantum oracle $\mathcal{U}$ relative to which $\mathsf{BQP}^{\mathcal{U}}/\mathsf{qpoly} \neq \mathsf{BQP}^{\mathcal{U}}/\mathsf{poly}$, which implies that hyperefficient shadow tomography is impossible if the observables are merely given as a black box that implements the measurement. The proof of this oracle separation amounts to showing that if the oracle $\mathcal{U}$ either (1) implements a reflection about a Haar-random n-qubit state, or (2) acts as the identity, then no $\mathrm{poly}(n)$-query algorithm can distinguish these two cases, even given a classical witness of size $\mathrm{poly}(n)$.

[3] Note the slight abuse of notation here, as the shadow tomography procedure can err with some small probability, and C itself might be a probabilistic quantum circuit. For simplicity, we assume that the shadow tomography procedure always succeeds and that C is deterministic in this exposition.

One can consider stronger forms of query access to the observables. For instance, in the common scenario where each observable measures fidelity with a pure state, meaning it has the form $O_i = |\psi_i\rangle\langle\psi_i|$, then in addition to the ability to measure overlap with $|\psi_i\rangle$, one might also have the power to produce copies of $|\psi_i\rangle$. Note that the ability to prepare $|\psi_i\rangle$ is generally much more powerful than the ability to recognize $|\psi_i\rangle$, the latter of which is equivalent to oracle access to the reflection $\mathbb{1} - 2|\psi_i\rangle\langle\psi_i|$. For example, Aaronson and Kuperberg's oracle separation of QCMA and QMA [4] amounts to building an oracle relative to which certain quantum states can be recognized efficiently but cannot be approximately prepared by small quantum circuits. Other black-box separations of state preparation and state reflection are known, e.g. [9], so one might hope that this type of query access could be substantially more powerful for shadow tomography as well.

Nevertheless, our results imply that black-box hyperefficient shadow tomography is impossible even in this setting where we have state preparation access to the observables. This follows from the simple observation that hyperefficient shadow tomography of this form would suffice to break PRS ensembles with a QCMA oracle.

▶ **Theorem 3.** *If a hyperefficient shadow tomography procedure exists that works for any list of observables of the form $|\psi_1\rangle\langle\psi_1|, \ldots, |\psi_M\rangle\langle\psi_M|$ given state preparation access to $|\psi_1\rangle, \ldots, |\psi_M\rangle$, then all PRS ensembles can be broken by polynomial-time quantum adversaries with oracle access to* QCMA.

Proof sketch. For a given PRS ensemble $\{|\varphi_k\rangle\}_{k\in\mathcal{K}}$, we have state preparation access to the observable list $\{|\varphi_k\rangle\langle\varphi_k|\}_{k\in\mathcal{K}}$ by way of the generating algorithm of the PRS. Hence, we can run hyperefficient shadow tomography using this observable list on copies of some unknown state $|\psi\rangle$. Suppose that this produces a quantum circuit C such that $|C(k) - \mathrm{Tr}(|\varphi_k\rangle\langle\varphi_k|\psi\rangle\langle\psi|)| \leq \frac{1}{10}$ for each $k \in \mathcal{K}$. Observe that the problem of deciding whether there exists some k such that $C(k) \geq \frac{9}{10}$ is in QCMA. If $|\psi\rangle$ is pseudorandom, then such a k always exists (whichever k satisfies $|\psi\rangle = |\varphi_k\rangle$), whereas if $|\psi\rangle$ is Haar-random, such a k exists with negligible probability over the choice of $|\psi\rangle$. Hence, these two ensembles can be distinguished by feeding C into this QCMA language. ◀

The above theorem also relativizes, in the sense that if the shadow tomography procedure only accesses the state preparation algorithm via a black box $\mathcal{O}$, then hyperefficient shadow tomography lets us break PRSs in polynomial time with oracle access to $\mathcal{O}$ and $\mathsf{QCMA}^{\mathcal{O}}$. Since Theorem 2 gives an oracle relative to which $\mathsf{BQP}^{\mathcal{O}} = \mathsf{QCMA}^{\mathcal{O}} = \mathsf{QMA}^{\mathcal{O}}$ and PRSs exist, we conclude that hyperefficient shadow tomography is impossible with only black-box state preparation access to the observables.

1.3 Our Techniques

The starting point for the proof of Theorem 1, which gives an upper bound of PP on the power needed to break pseudorandom states, is a theorem of Huang, Kueng, and Preskill [17] that gives a simple procedure for shadow tomography.

▶ **Theorem 4** ([17]). *Fix M different observables $O_1, O_2, \ldots, O_M$ and an unknown n-qubit mixed state ρ. Then there exists a quantum algorithm that performs $T = O(\log(M/\delta)/\varepsilon^2 \cdot \max_i \mathrm{Tr}(O_i^2))$ single-copy measurements in random Clifford bases of ρ, and uses the measurement results to estimate the quantities $\mathrm{Tr}(O_1\rho), \mathrm{Tr}(O_2\rho), \ldots, \mathrm{Tr}(O_M\rho)$, such that with probability at least $1 - \delta$, all of the M quantities are correct up to additive error ε.*

If $\{|\varphi_k\rangle_{k\in\mathcal{K}}\}$ is a family of PRSs, then by choosing $O_k = |\varphi_k\rangle\langle\varphi_k|$ for each $k \in \mathcal{K}$ to be the list of observables, we can use the above algorithm to determine whether ρ is close to one of the states in the PRS ensemble. A Haar-random state will be far from *all* of the pseudorandom states with overwhelming probability. Hence, Theorem 4 implies the existence of an algorithm that distinguishes the pseudorandom and Haar-random ensembles, by performing a polynomial number of random Clifford measurements and analyzing the results. The key observation is that the Clifford measurements can be performed efficiently, even though the resulting analysis (which operates on purely classical information) might be computationally expensive.

Next, one could try to argue that the computationally difficult steps in the above algorithm can be made efficient with a PP oracle. However, we take a different approach. We adopt a Bayesian perspective: suppose that with 50% probability we are given copies of a Haar-random state, and otherwise with 50% probability we are given copies of a randomly chosen state from the pseudorandom ensemble. We wish to distinguish these two cases using only the results of the random Clifford measurements as observed data. One way to do this is via the Bayes decision rule: we compute the posterior probability of being Haar-random or pseudorandom given the measurements, and then guess the more likely result. In fact, the Bayes decision rule is well-known to be the *optimal* decision rule in general, in the sense that any decision rule errs at least as often as the Bayes decision rule (see e.g. [11, Chapter 4.4.1]). Hence, because the algorithm of Huang, Kueng, and Preskill (Theorem 4) distinguishes the Haar-random and pseudorandom ensembles with good probability, the Bayes decision rule conditioned on the random Clifford measurements must work *at least* as well at the same distinguishing task.

Finally, we observe that using a quantum algorithm with postselection, we can approximate the relevant posterior probabilities needed for the Bayes decision rule. This allows us to appeal to the equivalence PostBQP = PP [1] to simulate this postselection with a PP oracle.

Technically, one challenge is that the postselected quantum algorithm requires the ability to prepare copies of a Haar-random state, even though a polynomial-time quantum algorithm cannot even approximately prepare most Haar-random states. The solution is to replace the Haar ensemble by an approximate quantum design, which we argue does not substantially change the success probability of the algorithm.

For our second result (Theorem 2), the oracle construction we use is simple to describe. The oracle $\mathcal{O}$ consists of two parts: a quantum oracle $\mathcal{U} = \{\mathcal{U}_n\}_{n\in\mathbb{N}}$, where each $\mathcal{U}_n$ consists of 2^n different Haar-random n-qubit unitary matrices, and a classical oracle $\mathcal{P}$ that is an arbitrary PSPACE-complete language. We prove that Theorem 2 holds with probability 1 over the choice of $\mathcal{U}$.

Showing that PRUs exist relative to $(\mathcal{U}, \mathcal{P})$ is reasonably straightforward. The proof uses the BBBV theorem (i.e. the optimality of Grover's algorithm) [10], and is analogous to showing that one-way functions or pseudorandom generators exist relative to a random *classical* oracle, as was shown by Impagliazzo and Rudich [18]. We only rigorously prove security against adversaries with classical advice, though we believe that the recently introduced framework of Chung, Guo, Liu, and Qian [16] should yield a security proof against adversaries with quantum advice.

Slightly more technically involved is proving that $\mathsf{BQP}^{\mathcal{U},\mathcal{P}} = \mathsf{QMA}^{\mathcal{U},\mathcal{P}}$. To do so, we argue that a QMA verifier is not substantially more powerful than a BQP machine at learning nontrivial properties of $\mathcal{U}$. More precisely, we argue that if a QMA verifier $\mathcal{V}$ makes T queries to $\mathcal{U}_n$ for some $n \in \mathbb{N}$, then either (1) $n = O(\log T)$ is sufficiently small that poly(T) queries to $\mathcal{U}_n$ actually suffice to learn $\mathcal{U}_n$ to inverse-polynomial precision, or else (2) $n = \omega(\log t)$ is sufficiently large that with high probability, the maximum acceptance probability of $\mathcal{V}$ (over the choice of Merlin's witness) is close to the average maximum acceptance probability of $\mathcal{V}$ when $\mathcal{U}_n$ is replaced by a random set of matrices sampled from the Haar measure. We prove this as a consequence of the extremely strong concentration of measure properties exhibited by the Haar measure [22].

This allows a $\mathsf{BQP}^{\mathcal{U},\mathcal{P}}$ machine to approximate the maximum acceptance probability of $\mathcal{V}^{\mathcal{U},\mathcal{P}}$ as follows. In case (1), the $\mathsf{BQP}^{\mathcal{U},\mathcal{P}}$ machine first queries $\mathcal{U}_n$ enough times to learn a unitary transformation $\widetilde{\mathcal{U}}_n$ that is close to $\mathcal{U}_n$, and then hard codes $\widetilde{\mathcal{U}}_n$ into a new $\mathsf{QMA}^{\mathcal{P}}$ verifier $\widetilde{\mathcal{V}}$ that simulates $\mathcal{V}$ by replacing queries to $\mathcal{U}_n$ with calls to $\widetilde{\mathcal{U}_n}$. In case (2), the $\mathsf{BQP}^{\mathcal{U},\mathcal{P}}$ machine similarly constructs a new $\mathsf{QMA}^{\mathcal{P}}$ verifier $\widetilde{\mathcal{V}}$, instead simulating $\mathcal{V}$ by replacing queries to $\mathcal{U}_n$ with unitaries chosen from an approximate polynomial design.[4] In both cases, $\widetilde{\mathcal{V}}$ defines a $\mathsf{QMA}^{\mathcal{P}}$ problem. Because $\mathcal{P}$ is PSPACE-complete, $\mathsf{BQP}^{\mathcal{P}} = \mathsf{QMA}^{\mathcal{P}} = \mathsf{PSPACE}$, and therefore this problem can be decided with a single query to $\mathcal{P}$.

The astute reader may notice that this proof works for more general choices of $\mathcal{P}$: it shows that for any oracle $\mathcal{P}$, if $\mathsf{BQP}^{\mathcal{P}} = \mathsf{QMA}^{\mathcal{P}}$, then $\mathsf{BQP}^{\mathcal{U},\mathcal{P}} = \mathsf{QMA}^{\mathcal{U},\mathcal{P}}$ with probability 1 over the choice of $\mathcal{U}$. An interesting consequence is the special case when $\mathcal{P}$ is trivial.

► **Corollary 5.** *If* $\mathsf{BQP} = \mathsf{QMA}$*, then* $\mathsf{BQP}^{\mathcal{U}} = \mathsf{QMA}^{\mathcal{U}}$ *with probability* 1 *over the choice of* $\mathcal{U}$*.*

In words, if $\mathsf{BQP} = \mathsf{QMA}$ in the unrelativized world, then the complexity classes also coincide relative to a collection $\mathcal{U}$ of Haar-random oracles. Or, viewed another way, separating BQP from QMA relative to $\mathcal{U}$ requires separating them in the unrelativized world. This is in stark contrast to the case of random *classical* oracles, where we can prove unconditionally that for a uniformly random oracle $\mathcal{O}$, $\mathsf{BQP}^{\mathcal{O}} \neq \mathsf{QMA}^{\mathcal{O}}$ (and indeed, $\mathsf{NP}^{\mathcal{O}} \not\subset \mathsf{BQP}^{\mathcal{O}}$) with probability 1 over $\mathcal{O}$ [10].

1.4 Open Problems

Can we prove a similar result to Theorem 2 using a *classical* oracle, for either PRUs or PRSs? Attempting to resolve this question seems to run into many of the same difficulties that arise in constructing a classical oracle separation between QCMA and QMA, which also remains an open problem [4]. For one, as pointed out in [4], we do not even know whether every n-qubit unitary transformation can be approximately implemented in poly(n) time relative to some classical oracle. Even if one could resolve this, it is not clear whether the resulting PRUs or PRSs would be secure against adversaries with the power of QMA. For instance, we show in Appendix C that an existing construction of PRSs, whose security is provable in the random oracle model [13], can be broken with an NP oracle.

What else can be said about the hardness of learning quantum states and unitary transformations, either in the worst case or on average? A related question is to explore the hardness of problems involving quantum *meta-complexity*: that is, problems that themselves encode computational complexity or difficulty. Consider, for example, a version of the minimum circuit size problem (MCSP) for quantum states: given copies of a pure quantum state $|\psi\rangle$, determine the size of the smallest quantum circuit that approximately outputs $|\psi\rangle$. If PRSs exist, then this task should be hard, but placing an upper bound on the complexity of this task might be difficult in light of our results. We view this problem as particularly intriguing because it does not appear to have an obvious classical analogue, and also because of

[4] Technically, this requires choosing a random element of the polynomial design for each $x \in \{0,1\}^n$ by means of a random oracle, so we use Zhandry's strategy [28] to simulate T quantum queries to a random oracle using a $2T$-wise independent function.

its relevance to the wormhole growth paradox and Susskind's Complexity=Volume conjecture in AdS/CFT [12, 25, 24]. A number of recent breakthroughs in complexity theory have involved ideas from meta-complexity (see surveys by Allender [6, 7]), and it would be interesting to see which of these techniques could be ported to the quantum setting.

What other complexity-theoretic evidence can be given for the existence of PRSs and PRUs? Can we give candidate constructions of PRSs or PRUs that do not rely on the assumption $\mathsf{BQP} \neq \mathsf{QMA}$? To give a specific example, an interesting question is whether polynomial-size quantum circuits with random local gates form PRUs. Random circuits are known to information-theoretically approximate the Haar measure in the sense that they form approximate unitary designs [15], and it seems conceivable that they could also be computationally pseudorandom.

2 Preliminaries

2.1 Notation

Throughout, $[n]$ denotes the set of integers $\{1, 2, \ldots, n\}$, and $[n, m]$ denotes the set of integers $\{n, n+1, n+2, \ldots, m\}$. If $x \in \{0,1\}^n$ is a binary string, then $|x|$ denotes the length of x. For X a finite set, we let $|X|$ denote the size of X. If X is a probability distribution, then we use $x \leftarrow X$ to denote a random variable x sampled according to X. When X is a finite set, we also use $x \leftarrow X$ to indicate a random variable x drawn uniformly from X. A function $f(n)$ is *negligible* if for every constant $c > 0$, $f(n) \leq \frac{1}{n^c}$ for all sufficiently large n. We use $\operatorname{negl}(n)$ to denote an arbitrary negligible function, and $\operatorname{poly}(n)$ to denote an arbitrary polynomially-bounded function.

We use $||M||_F = \sqrt{\operatorname{Tr}(M^\dagger M)}$ to denote the Frobenius norm of a matrix M. We denote by $||A||_\diamond$ the diamond norm of a superoperator A acting on density matrices (see [5] for a definition). For a unitary matrix U, we use $U \cdot U^\dagger$ to denote the superoperator that maps a density matrix ρ to $U \rho U^\dagger$.

We use $\mathbb{S}(N)$ to denote the set of N-dimensional pure quantum states, and $\mathbb{U}(N)$ to denote the group of $N \times N$ unitary matrices. When $N = 2^n$, we identify these with n-qubit states and unitary transformations, respectively. We use σ_N to denote the Haar measure on $\mathbb{S}(N)$, and we let μ_N denote the Haar measure over $\mathbb{U}(N)$. We write $\mathbb{U}(N)^M$ for the space of $MN \times MN$ block-diagonal unitary matrices, where each block has size $N \times N$, and we also identify $\mathbb{U}(N)^M$ with M-tuples of $N \times N$ unitary matrices. We use μ_N^M to denote the product measure $\mu_N^M(U_1, U_2, \ldots, U_M) = \mu_N(U_1) \cdot \mu_N(U_2) \cdots \mu_N(U_M)$ on $\mathbb{U}(N)^M$.

We assume familiarity with standard complexity classes such as BQP and PP, including relativized versions of these classes that can query a quantum or classical oracle. For completeness, we define some of the relevant complexity classes and related notions in Appendix B.

We use superscript notation for algorithms that query oracles. For instance, $\mathcal{A}^{\mathcal{U}}(x, |\psi\rangle)$ denotes a quantum algorithm $\mathcal{A}$ that queries an oracle $\mathcal{U}$ and receives a classical input x and a quantum input $|\psi\rangle$.

2.2 Quantum Information

We require the following well-known fact, which bounds the distance in the diamond norm between two unitary superoperators in terms of the Frobenius norm of the difference of the two matrices. We provide a proof in Appendix A.

▶ **Lemma 6.** *Let $U, V \in \mathbb{U}(N)$. Then $||U \cdot U^\dagger - V \cdot V^\dagger||_\diamond \leq 2||U - V||_F$.*

We use the notion of an ε-approximate quantum (state) t-design, which is a distribution over quantum states that information-theoretically approximates the Haar measure over states.

▶ **Definition 7** (Approximate quantum design [8]). *A probability distribution S over $\mathbb{S}(N)$ is an* ε-approximate quantum t-design *if:*

$$(1-\varepsilon) \underset{|\psi\rangle \leftarrow \sigma_N}{\mathbb{E}} |\psi\rangle\langle\psi|^{\otimes t} \leq \underset{|\psi\rangle \leftarrow S}{\mathbb{E}} |\psi\rangle\langle\psi|^{\otimes t} \leq (1+\varepsilon) \underset{|\psi\rangle \leftarrow \sigma_N}{\mathbb{E}} |\psi\rangle\langle\psi|^{\otimes t}$$

and:

$$\underset{|\psi\rangle \leftarrow S}{\mathbb{E}} |\psi\rangle\langle\psi| = \underset{|\psi\rangle \leftarrow \sigma_N}{\mathbb{E}} |\psi\rangle\langle\psi| .$$

Similarly, we require ε-approximate *unitary* t-designs, which are approximations to the Haar measure over unitary matrices. The definition of ε-approximate unitary t-designs is more technical, so we point to [15, Definition 2] for a formal definition. While there are several definitions of approximate t-designs used in the literature, for this work it is crucial that we use *multiplicative* approximate designs for both states and unitaries, meaning that the designs approximate the first t moments of the Haar measure to within a multiplicative $1 \pm \varepsilon$ error (as opposed to additive error).

Efficient constructions of approximate unitary t-designs over qubits are known, as below.

▶ **Lemma 8.** *Fix $\varepsilon > 0$. For each $n, t \in \mathbb{N}$, there exists $m(n) \leq \mathrm{poly}(n)$ and a $\mathrm{poly}(n,t)$-time classical algorithm $\mathcal{S}$ that takes as input a random string $x \leftarrow \{0,1\}^m$ and outputs a description of a quantum circuit on n qubits such that the circuits sampled from $\mathcal{S}$ form an ε-approximate unitary t-design over $\mathbb{U}(2^n)$.*

Proof sketch. Fix an arbitrary universal quantum gate set G with algebraic entries that is closed under taking inverses (e.g. $G = \{\mathrm{CNOT}, H, T, T^\dagger\}$). Brandão, Harrow, and Horodecki [15, Corollary 7] show that n-qubit quantum circuits consisting of $\mathrm{poly}(n,t)$ random gates sampled from G, applied to random pairs of qubits, form ε-approximate unitary t-designs. So, $\mathcal{S}$ just has to sample from this distribution, which can be done with $\mathrm{poly}(n,t)$ bits of randomness. ◀

Note that this also implies an efficient construction of ε-approximate quantum (state) t-designs, as if S is an ε-approximate unitary t-design over $\mathbb{U}(N)$ then $S|\psi\rangle$ is an ε-approximate quantum t-design for any fixed $|\psi\rangle$ (e.g. $|0^n\rangle$).

Essentially the only property we need of approximate t-designs is that they can be used in place of the Haar measure in any quantum algorithm that uses t copies of a Haar-random state (or t queries to a Haar-random unitary), and the measurement probabilities of the algorithm will change by only a small multiplicative factor.

▶ **Fact 9.** *Let S be an ε-approximate quantum t-design over $\mathbb{S}(N)$, and let $\mathcal{A}$ be an arbitrary quantum measurement. Then:*

$$(1-\varepsilon) \underset{|\psi\rangle \leftarrow \sigma_N}{\Pr} \left[\mathcal{A}\left(|\psi\rangle^{\otimes t}\right) = 1\right] \leq \underset{|\psi\rangle \leftarrow S}{\Pr} \left[\mathcal{A}\left(|\psi\rangle^{\otimes t}\right) = 1\right] \leq (1+\varepsilon) \underset{|\psi\rangle \leftarrow \sigma_N}{\Pr} \left[\mathcal{A}\left(|\psi\rangle^{\otimes t}\right) = 1\right].$$

▶ **Fact 10** ([15]). *Let S be an ε-approximate unitary t-design over $\mathbb{U}(N)$, and let $\mathcal{A}^U$ be an arbitrary quantum algorithm that makes t queries to some $U \in \mathbb{U}(N)$. Then:*

$$(1-\varepsilon) \underset{U \leftarrow \mu_N}{\Pr} \left[\mathcal{A}^U = 1\right] \leq \underset{U \leftarrow S}{\Pr} \left[\mathcal{A}^U = 1\right] \leq (1+\varepsilon) \underset{U \leftarrow \mu_N}{\Pr} \left[\mathcal{A} = 1\right].$$

We require the following concentration inequality on the Haar measure, which is stated in terms of Lipschitz continuous functions. For a metric space M with metric d, a function $f : M \to \mathbb{R}$ is *L-Lipschitz* if for all $x, y \in M$, $|f(x) - f(y)| \le L \cdot d(x, y)$.

▶ **Theorem 11** ([22, Theorem 5.17]). *Given $N_1, \ldots, N_k \in \mathbb{N}$, let $X = \mathbb{U}(N_1) \oplus \cdots \oplus \mathbb{U}(N_k)$ be the space of block-diagonal unitary matrices with blocks of size $N_1, \ldots, N_k$. Let $\mu = \mu_{N_1} \times \cdots \times \mu_{N_k}$ be the product of Haar measures on X. Suppose that $f : X \to \mathbb{R}$ is L-Lipschitz in the Frobenius norm. Then for every $t > 0$:*

$$\Pr_{U \leftarrow \mu}\left[f(U) \ge \mathop{\mathbb{E}}_{V \leftarrow \mu}[f(V)] + t\right] \le \exp\left(-\frac{(N-2)t^2}{24L^2}\right),$$

where $N = \min\{N_1, \ldots, N_k\}$.

2.3 Cryptography

A family of functions $\{f_k\}_{k \in \mathcal{K}}$ where $f_k : \{0,1\}^n \to \{0,1\}^m$ is called *t-wise independent* if for every distinct $x_1, x_2, \ldots, x_t \in \{0,1\}^n$ and every (not necessarily distinct) $y_1, y_2, \ldots, y_t \in \{0,1\}^m$:

$$\Pr_{k \leftarrow \mathcal{K}}[f_k(x_i) = y_i \ \forall i \in [t]] = 2^{-mt}.$$

Efficient constructions of t-wise independent functions are known, in the sense that one can sample a random f_k from a t-wise independent function family and make queries to f_k in $\mathrm{poly}(t, n, m)$ time [28]. Our primary use of t-wise independent functions is in simulating random oracles: $2t$-wise independent functions can be used in place of a uniformly random function $\{0,1\}^n \to \{0,1\}^m$ in any quantum algorithm that makes at most t queries to the random function; see Zhandry [28, Theorem 6.1] for more details.

We use the following definitions of pseudorandom quantum states (PRSs) and pseudorandom unitaries (PRUs), which were introduced by Ji, Liu, and Song [19].

▶ **Definition 12** (Pseudorandom quantum states [19]). *Let $\kappa \in \mathbb{N}$ be the security parameter. Let D be the dimension of a quantum system and let $\mathcal{K}$ be the key set, both parameterized by κ. A keyed family of quantum states $\{|\varphi_k\rangle\}_{k \in \mathcal{K}} \subset \mathbb{S}(D)$ is* pseudorandom *if the following two conditions hold:*

(1) *(Efficient generation) There is a polynomial-time quantum algorithm G that generates $|\varphi_k\rangle$ on input k, meaning $G(k) = |\varphi_k\rangle$.*

(2) *(Computationally indistinguishable) For any polynomial-time quantum algorithm $\mathcal{A}$ and $T = \mathrm{poly}(\kappa)$:*

$$\left|\Pr_{k \leftarrow \mathcal{K}}\left[\mathcal{A}\left(|\varphi\rangle^{\otimes T}\right) = 1\right] - \Pr_{|\psi\rangle \leftarrow \sigma_D}\left[\mathcal{A}\left(|\psi\rangle^{\otimes T}\right) = 1\right]\right| = \mathrm{negl}(\kappa).$$

▶ **Definition 13** (Pseudorandom unitary transformations [19]). *Let $\kappa \in \mathbb{N}$ be the security parameter. Let D be the dimension of a quantum system and let $\mathcal{K}$ be the key set, both parameterized by κ. A keyed family of unitary transformations $\{U_k\}_{k \in \mathcal{K}} \subset \mathbb{U}(D)$ is* pseudorandom *if the following two conditions hold:*

(1) *(Efficient computation) There is a polynomial-time quantum algorithm G that implements U_k on input k, meaning that for any $|\psi\rangle \in \mathbb{S}(D)$, $G(k, |\psi\rangle) = U_k |\psi\rangle$.*

(2) *(Computationally indistinguishable) For any polynomial-time quantum algorithm $\mathcal{A}^U$ that queries $U \in \mathbb{U}(D)$:*

$$\left|\Pr_{k \leftarrow \mathcal{K}}\left[\mathcal{A}^{U_k}(1^\kappa) = 1\right] - \Pr_{U \leftarrow \mu_D}\left[\mathcal{A}^U(1^\kappa) = 1\right]\right| = \mathrm{negl}(\kappa).$$

We generally take the key set $\mathcal{K} = \{0,1\}^\kappa$ and choose $D = 2^n$ for some $n = \mathrm{poly}(\kappa)$ in the above definitions. We sometimes call the negligible quantities in the above definitions the *advantage* of the quantum adversary $\mathcal{A}$.

In this work, we consider security against non-uniform quantum algorithms with classical advice, which means that the adversary is allowed to be a different polynomial-time quantum algorithm for each setting of the security parameter $\kappa \in \mathbb{N}$. Without loss of generality, such an adversary can always be assumed to take the form of a *uniform* $\mathrm{poly}(\kappa)$-time quantum algorithm $\mathcal{A}(1^\kappa, x)$, where $x \in \{0,1\}^{\mathrm{poly}(\kappa)}$ is an advice string that depends only on κ.

3 Breaking Pseudorandomness with a Classical Oracle

In this section, we prove that a polynomial-time quantum algorithm with a PP oracle can distinguish a PRS from a Haar-random state. First, we need a lemma about the overlap between a fixed state $|\varphi\rangle$ and a Haar-random state $|\psi\rangle$.

▶ **Lemma 14.** *Let $|\varphi\rangle \in \mathbb{S}(N)$, and let $\varepsilon > 0$. Then:*

$$\Pr_{|\psi\rangle \leftarrow \sigma_N}\left[|\langle\psi|\varphi\rangle|^2 \geq \varepsilon\right] \leq e^{-\varepsilon N}.$$

Proof. This follows from standard concentration inequalities, or even an explicit computation, using the fact that $|\langle\psi|\varphi\rangle|^2$ is roughly exponentially distributed for a random state $|\psi\rangle$. See e.g. [15, Equation (14)] ◀

The formal statement of our result is below.

▶ **Theorem 15.** *For any PRS ensemble $\{|\varphi_k\rangle\}_{k\in\mathcal{K}}$ of n-qubit states with security parameter κ, there exists a PP language $\mathcal{L}$, a $\mathrm{poly}(\kappa)$-time quantum algorithm $\mathcal{A}^{\mathcal{L}}$, and $T = \mathrm{poly}(\kappa)$ such that the following holds. Let $X \leftarrow \{0,1\}$ be a uniform random bit. Let $|\psi\rangle$ be sampled uniformly from the PRS ensemble if $X = 0$, and otherwise let $|\psi\rangle$ be sampled from the Haar measure σ_{2^n} if $X = 1$. Then we have:*

$$\Pr_{X,|\psi\rangle}\left[\mathcal{A}^{\mathcal{L}}\left(|\psi\rangle^{\otimes T}\right) = X\right] \geq 0.995.$$

Proof. We first describe $\mathcal{A}$. For some T to be chosen later, on input $|\psi\rangle^{\otimes T}$, $\mathcal{A}$ measures each copy of $|\psi\rangle$ in a different randomly chosen Clifford basis. Call the list of measurement bases $b = (b_1, b_2, \ldots, b_T)$ and the measurement results $c = (c_1, c_2, \ldots, c_T)$. $\mathcal{A}$ then feeds (b, c) into a single query to $\mathcal{L}$, and outputs the result of the query. This takes polynomial time because there exists an $O(n^3)$-time algorithm to sample a random n-qubit Clifford unitary, and this algorithm also produces an implementation of the unitary with $O(n^2/\log n)$ gates [20, 3].

The PP language $\mathcal{L}$ we choose for the oracle is most easily described in terms of a PostBQP algorithm $\mathcal{B}(b, c)$ (i.e. a postselected polynomial-time quantum algorithm, as in Definition 23), by the equivalence PostBQP = PP [1].[5] Let S be a $\frac{1}{17}$-approximate n-qubit quantum T-design (Definition 7) such that a state can be drawn from S in $\mathrm{poly}(\kappa)$ time (because $n, T \leq \mathrm{poly}(\kappa)$, the existence of such a design follows from Lemma 8). $\mathcal{B}$ begins by initializing the state:

$$\hat{\rho} = \frac{1}{2}|0\rangle\langle 0| \otimes \mathop{\mathbb{E}}_{k\leftarrow\mathcal{K}}\left[|\varphi_k\rangle\langle\varphi_k|^{\otimes T}\right] + \frac{1}{2}|1\rangle\langle 1| \otimes \mathop{\mathbb{E}}_{|\psi\rangle\leftarrow S}\left[|\psi\rangle\langle\psi|^{\otimes T}\right].$$

[5] Note that any promise problem in PostBQP is also in PP [1], and any promise problem in PP can be extended to a language in PP because PP is a syntactic class. Hence, we might as well take a language in PP instead of a promise problem.

$\mathcal{B}$ measures all but the leftmost qubit of $\hat{\rho}$ in the basis given by b, and postselects on observing c (i.e. $\mathcal{B}$ outputs $*$ if the measurements are not equal to c). Finally, conditioned on postselection succeeding, $\mathcal{B}$ measures and outputs the result of the leftmost qubit that was not measured.

It remains to show that $\mathcal{A}$ distinguishes the pseudorandom and Haar-random state ensembles. For the purpose of this analysis, it will be convenient to view $\hat{\rho}$ as an approximation to the state:

$$\rho = \frac{1}{2} |0\rangle \langle 0| \otimes \mathop{\mathbb{E}}_{k \leftarrow \mathcal{K}} \left[|\varphi_k\rangle \langle \varphi_k|^{\otimes T} \right] + \frac{1}{2} |1\rangle \langle 1| \otimes \mathop{\mathbb{E}}_{|\psi\rangle \leftarrow \sigma_{2^n}} \left[|\psi\rangle \langle \psi|^{\otimes T} \right],$$

where the ε-approximate T-design S is replaced by the Haar measure σ_{2^n}. Indeed, we will essentially argue the algorithm's correctness if the state $\hat{\rho}$ is replaced by ρ, and then argue that this implies the correctness of the actual algorithm.

For each $k \in \mathcal{K}$, define $O_k = |\varphi_k\rangle \langle \varphi_k|$. Note that if $X = 0$ (i.e. $|\psi\rangle$ is pseudorandom), there always exists a k such that $\mathrm{Tr}(O_k |\psi\rangle \langle \psi|) = 1$, namely whichever k satisfies $|\psi\rangle = |\varphi_k\rangle$. On the other hand, by Lemma 14 and a union bound, if $X = 1$ (i.e. $|\psi\rangle$ is Haar-random), $\mathrm{Tr}(O_k |\psi\rangle \langle \psi|) < \frac{1}{3}$ for every $k \in \mathcal{K}$, except with probability at most $|\mathcal{K}| \cdot e^{-2^n/3} \leq \mathrm{negl}(\kappa)$ over $|\psi\rangle$.

If we choose $M = |\mathcal{K}|$, $\varepsilon = \frac{1}{3}$, and $\delta = 0.001 - |\mathcal{K}| \cdot e^{-2^n/3}$, then by Theorem 4 there exists a quantum algorithm that takes as input the results (b, c) of $T = O(\log |\mathcal{K}|) = O(\kappa)$ single-copy random Clifford measurements, uses the measurement results to estimate $\mathrm{Tr}(O_k |\psi\rangle \langle \psi|)$ for each k up to additive error $\frac{1}{3}$, and is correct with probability at least $0.999 + |\mathcal{K}| \cdot e^{-2^n/3}$. In particular, this algorithm can distinguish the pseudorandom ensemble from the Haar-random ensemble, by checking if there exists a k such that the estimate for $\mathrm{Tr}(O_k |\psi\rangle \langle \psi|)$ is at least $\frac{2}{3}$. Call this algorithm $\mathcal{C}$, so that $\Pr[\mathcal{C}(b, c) = X] \geq 0.999$.

We will not actually use $\mathcal{C}$, but only its existence. By the optimality of the Bayes decision rule [11, Chapter 4.4.1], because $\mathcal{C}$ uses (b, c) to identify a state $|\psi\rangle$ as either Haar-random or pseudorandom with probability 0.999, an algorithm that computes the maximum a posteriori estimate of X also succeeds with probability at least 0.999. In symbols, let $p_i = \Pr[X = i \mid b, c]$, which we view as a random variable (depending on b and c) for each $i \in \{0, 1\}$. Then $\Pr\left[\arg\max_i p_i = X\right] \geq 0.999$.

Next, observe that $\Pr\left[\arg\max_i p_i = X\right] = \mathbb{E}\left[\Pr\left[\arg\max_i p_i = X | b, c\right]\right] = \mathbb{E}\left[\max_i p_i\right]$, by the law of total expectation. Hence, by Markov's inequality (and the fact that $\frac{1}{2} \leq \max_i p_i \leq 1$), we know that $\Pr\left[\max_i p_i \geq \frac{3}{4}\right] \geq 0.996$. In other words, the Bayes decision rule is usually confident in its predictions, so to speak.

Notice that p_i equals the probability (conditioned on postselection succeeding) that $\mathcal{B}$ outputs i if it starts with ρ in place of $\hat{\rho}$. For $i \in \{0, 1\}$, define $\hat{p}_i$ analogously as the postselcted output probabilities of $\mathcal{B}$ itself: $\hat{p}_i = \Pr\left[\mathcal{B}(b, c) = i \mid \mathcal{B}(b, c) \in \{0, 1\}\right]$. To argue that $\mathcal{A}$ is correct with 0.995 probability, it suffices to show that $\Pr\left[\max_i \hat{p}_i \geq \frac{2}{3} \wedge \arg\max_i \hat{p}_i = X\right] \geq 0.995$, as in this case the $\mathsf{PostBQP}$ promise is satisfied and the output of $\mathcal{L}$ agrees with X. We have that:

$$\begin{aligned}
\Pr\left[\max_i \hat{p}_i \geq \frac{2}{3} \wedge \arg\max_i \hat{p}_i = X\right] &\geq \Pr\left[\max_i p_i \geq \frac{3}{4} \wedge \arg\max_i p_i = X\right] \\
&\geq 1 - \Pr\left[\max_i p_i < \frac{3}{4}\right] - \Pr\left[\arg\max_i p_i \neq X\right] \\
&\geq 0.996 - \Pr\left[\arg\max_i p_i \neq X\right] \\
&\geq 0.995
\end{aligned}$$

Above, the first inequality follows from the assumption that S is a $\frac{1}{17}$-approximate T-design, because the acceptance probability of a postselected quantum algorithm can be viewed as the ratio of two probabilities:

$$\hat{p_i} = \frac{\Pr[\mathcal{B}(b,c) = i]}{\Pr[\mathcal{B}(b,c) \in \{0,1\}]}.$$

Fact 9 implies that both the numerator and denominator change by at most a multiplicative factor of $1 \pm \frac{1}{17}$ when switching between ρ and $\hat{\rho}$. So, if $p_i \geq \frac{3}{4}$, then $\hat{p}_i \geq \frac{3}{4} \cdot \frac{1-\frac{1}{17}}{1+\frac{1}{17}} = \frac{2}{3}$. The second inequality follows by a union bound, and the remaining inequalities were established above. ◀

We remark that the above theorem also holds relative to all oracles, in the sense that if the state generation algorithm G in the definition of the PRS (Definition 12) queries a classical or quantum oracle $\mathcal{U}$, then the corresponding ensemble of states can be distinguished from Haar-random by a polynomial-time quantum algorithm with a $\mathsf{PostBQP}^{\mathcal{U}}$ oracle.

4 Pseudorandomness from a Quantum Oracle

In this section, we construct a quantum oracle $(\mathcal{U}, \mathcal{P})$ relative to which $\mathsf{BQP} = \mathsf{QMA}$ and PRUs exist.

4.1 $\mathsf{BQP} = \mathsf{QMA}$ Relative to $(\mathcal{U}, \mathcal{P})$

We start with a lemma showing that the acceptance probability of a quantum query algorithm, viewed as a function of the unitary transformation used in the query, is Lipschitz.

▶ **Lemma 16.** *Let $\mathcal{A}^U$ be quantum algorithm that makes T queries to $U \in \mathbb{U}(D)$, and define $f(U) = \Pr\left[\mathcal{A}^U = 1\right]$. Then f is $2T$-Lipschitz in the Frobenius norm.*

Proof. Suppose that $||U - V||_F \leq d$. By Lemma 6, this implies that the distance between U and V in the diamond norm is at most $2d$. The sub-additivity of the diamond norm under composition implies that as superoperators, $||\mathcal{A}^U - \mathcal{A}^V||_\diamond \leq 2Td$. By the definition of the diamond norm, it must be the case that $|f(U) - f(V)| \leq 2Td$. ◀

The next lemma extends Lemma 16 to QMA verifiers: we should think of $\mathcal{V}$ as a QMA verifier that receives a witness $|\psi\rangle$, in which case this lemma states that the maximum acceptance probability of $\mathcal{V}$ is Lipschitz with respect to the queried unitary.

▶ **Lemma 17.** *Let $\mathcal{V}^U(|\psi\rangle)$ be quantum algorithm that makes T queries to $U \in \mathbb{U}(D)$ and takes as input a quantum state $|\psi\rangle$ on some fixed (but arbitrary) number of qubits. Define $f(U) = \max_{|\psi\rangle} \Pr\left[\mathcal{V}^U(|\psi\rangle) = 1\right]$. Then f is $2T$-Lipschitz in the Frobenius norm.*

Proof. Note that f is well-defined because of the extreme value theorem. Define $f_\psi : \mathbb{U}(D) \to \mathbb{R}$ by:

$$f_\psi(U) = \Pr\left[\mathcal{V}^U(|\psi\rangle) = 1\right],$$

so that $f(U) = \max_{|\psi\rangle} f_\psi(U)$. Lemma 16 implies that f_ψ is $2T$-Lipschitz for every $|\psi\rangle$. Let $U, V \in \mathbb{U}(D)$, and suppose that $|\psi\rangle$ and $|\varphi\rangle$ are such that $f(U) = f_\psi(U)$ and $f(V) = f_\varphi(V)$.

Then:

$$\begin{aligned}|f(U)-f(V)| &= |f_\psi(U)-f_\varphi(V)|\\ &= \max\{f_\psi(U)-f_\varphi(V), f_\varphi(V)-f_\psi(U)\}\\ &\le \max\{f_\psi(U)-f_\psi(V), f_\varphi(V)-f_\varphi(U)\}\\ &\le 2T||U-V||_F,\end{aligned}$$

where the third line uses the fact that $f_\psi(V) \le f_\varphi(V)$ and $f_\varphi(U) \le f_\psi(U)$, and the last line uses the fact that f_ψ and f_φ are $2T$-Lipschitz. ◀

We are now ready to prove the first main result of this section, that $\mathsf{BQP}^{\mathcal{U},\mathcal{P}} = \mathsf{QMA}^{\mathcal{U},\mathcal{P}}$.

▶ **Theorem 18.** *Let $\mathcal{U} = \{\mathcal{U}_n\}_{n\in\mathbb{N}}$ be a quantum oracle where each $\mathcal{U}_n$ is chosen randomly from $\mu_{2^n}^{2^n}$. Let $\mathcal{P}$ be an arbitrary* PSPACE*-complete language. Then with probability* 1 *over $\mathcal{U}$,* $\mathsf{BQP}^{\mathcal{U},\mathcal{P}} = \mathsf{QMA}^{\mathcal{U},\mathcal{P}}$.

Proof. First, some notation. We view each $\mathcal{U}_n$ alternatively as either a unitary transformation on $2n$ qubits, or as a list of 2^n different n-qubit unitary transformations $\mathcal{U}_n = \{\mathcal{U}_{nm}\}_{m\in\{0,1\}^n}$ indexed by n-bit strings.

Let $\mathcal{L} \in \mathsf{QMA}^{\mathcal{U},\mathcal{P}}$, which means that there exists a polynomial-time $\mathsf{QMA}^{\mathcal{U},\mathcal{P}}$ verifier $\mathcal{V}^{\mathcal{U},\mathcal{P}}(x, |\psi\rangle)$ with completeness $\frac{2}{3}$ and soundness $\frac{1}{3}$. Without loss of generality, we can amplify the completeness and soundness probabilities of $\mathcal{V}$ to $\frac{11}{12}$ and $\frac{1}{12}$, respectively. Let $p(n)$ be a polynomial upper bound on the running time of $\mathcal{V}$ on inputs of length n.

We now describe a $\mathsf{BQP}^{\mathcal{U},\mathcal{P}}$ machine $\mathcal{A}^{\mathcal{U},\mathcal{P}}(x)$ such that, with probability 1 over $\mathcal{U}$, $\mathcal{A}$ computes $\mathcal{L}$ on all but finitely many inputs $x \in \{0,1\}^*$. Let $d = \lfloor \log_2\left(13824|x|p(|x|)^2+2\right)\rfloor$. For each $n \in [d]$, $\mathcal{A}$ performs process tomography on each $\mathcal{U}_n$, producing estimates $\widetilde{\mathcal{U}}_n$ such that $||\widetilde{\mathcal{U}}_n \cdot \widetilde{\mathcal{U}}_n^\dagger - \mathcal{U}_n \cdot \mathcal{U}_n^\dagger||_\diamond \le \frac{1}{12p(|x|)}$ for every n, with probability at least $\frac{2}{3}$ over the randomness of $\mathcal{A}$.[6] Let $\mathcal{S}$ be the algorithm from Lemma 8 that samples from a $\frac{1}{12}$-approximate unitary $p(|x|)$-design on n qubits, given as input a random seed $r \leftarrow \{0,1\}^{k_n}$ where $k_n = \mathrm{poly}(n, |x|)$.

Consider a $\mathsf{QMA}^{\mathcal{P}}$ verifier $\widetilde{\mathcal{V}}^{\mathcal{P}}(x, |\psi\rangle)$ that simulates $\mathcal{V}^{\mathcal{U},\mathcal{P}}(x, |\psi\rangle)$ by replacing queries to $\mathcal{U}$ as follows. For each $n \in [d+1, p(|x|)]$, $\widetilde{\mathcal{V}}$ samples a function $f_n : \{0,1\}^n \to \{0,1\}^{k_n}$ from a $2p(|x|)$-wise independent function family. Then, for $n \in [d]$, queries to $\mathcal{U}_n$ are replaced by queries to $\widetilde{\mathcal{U}}_n$. For $n \in [d+1, p(|x|)]$ and $m \in \{0,1\}^n$, queries to $\mathcal{U}_{nm}$ are replaced by queries to $\mathcal{S}(f_n(m))$ (i.e. the mth unitary in $\mathcal{U}_n$ is replaced by an element of the $p(|x|)$-design, selected by $f_n(m)$). Consider the $\mathsf{QMA}^{\mathcal{P}}$ promise problem $\widetilde{\mathcal{L}}$ corresponding to $\widetilde{V}$ with completeness $\frac{2}{3}$ and soundness $\frac{1}{3}$. Since $\mathsf{QMA}^A \subseteq \mathsf{PSPACE}^A$ for all classical oracles A, $\widetilde{\mathcal{L}} \in \mathsf{PSPACE}^{\mathcal{P}} = \mathsf{PSPACE}$, so $\mathcal{A}$ can decide $\widetilde{\mathcal{L}}(x)$ with a single query to $\mathcal{P}$. $\mathcal{A}$ does this, and outputs $\widetilde{\mathcal{L}}(x)$.

We now argue that for any x, with high probability over $\mathcal{U}$, $\Pr\left[\mathcal{A}^{\mathcal{U},\mathcal{P}}(x) = \mathcal{L}(x)\right] \ge \frac{2}{3}$. It will be convenient to define several hybrid verifiers:

- $V_1 = \mathcal{V}$.
- V_2: For each $n \in [d+1, p(|x|)]$, chooses a matrix $U_n \leftarrow \mu_{2^n}^{2^n}$. Simulates V_1, replacing queries to $\mathcal{U}_n$ by U_n for $n \in [d+1, p(|x|)]$.

[6] There are many ways to accomplish this. For instance, one can use the Choi-Jamiołkowski isomorphism and quantum state tomography [23] to estimate the Choi state of $\mathcal{U}_n$ to inverse polynomial (in 2^n) error in trace distance. The estimated unitary transformation $\widetilde{\mathcal{U}}_n$ can then be compiled to a circuit using $2^{O(n)}$ 1- and 2-qubit gates [26]. Since $n \le d = O(\log|x|)$, this can be done in polynomial time.

- V_3: For each $n \in [d+1, p(|x|)]$, samples a function $g_n : \{0,1\}^n \to \{0,1\}^{k_n}$ uniformly at random. Simulates V_2, replacing queries to U_{nm} by by queries to $\mathcal{S}(g_n(m))$ for $n \in [d+1, p(|x|)]$ and $m \in \{0,1\}^n$.
- V_4: For each $n \in [d+1, p(|x|)]$, samples a function $f_n : \{0,1\}^n \to \{0,1\}^{k_n}$ from a $2p(|x|)$-wise independent function family. Simulates V_3, replacing queries to g_n by f_n.
- V_5: Simulates V_4, replacing queries to $\mathcal{U}_n$ by queries to $\widetilde{\mathcal{U}_n}$ for $n \in [d]$. Note that V_5 and $\widetilde{\mathcal{V}}$ are equivalent, and that of these hybrids, V_5 is the only one whose output depends on the randomness of $\mathcal{A}$ (by way of $\widetilde{\mathcal{U}_n}$).

For $i \in [5]$ and a fixed choice of $\mathcal{U}$, define:

$$\mathrm{acc}(V_i) = \max_{|\psi\rangle} \Pr\left[V_i(x, |\psi\rangle) = 1\right],$$

which is well defined by the extreme value theorem. We now bound $|\mathrm{acc}(V_i) - \mathrm{acc}(V_{i-1})|$ for various i:

- By Lemma 17, because $\mathcal{V}$ makes at most $p(|x|)$ queries, we know that $\mathrm{acc}(V_1)$, viewed as a function of $(\mathcal{U}_{d+1}, \mathcal{U}_{d+2}, \ldots, \mathcal{U}_{p(|x|)})$, is $2p(|x|)$-Lipschitz in the Frobenius norm. Hence, by Theorem 11 with $N = 13824|x|p(|x|)^2 + 2$, $L = 2p(|x|)$, and $t = \frac{1}{12}$, we have that:

$$\begin{aligned}\Pr_{\mathcal{U}}\left[|\mathrm{acc}(V_1) - \mathrm{acc}(V_2)| \geq \frac{1}{12}\right] &\leq 2\exp\left(-\frac{(N-2)t^2}{24L^2}\right)\\ &= 2\exp\left(-\frac{13824|x|p(|x|)^2 \cdot \frac{1}{144}}{96p(|x|)^2}\right)\\ &= 2e^{-|x|}.\end{aligned}$$

  The factor of 2 appears because Theorem 11 applies to one-sided error, but the absolute value forces us to consider two-sided error.
- Fact 10 and the assumption that $\mathcal{S}$ samples from a $\frac{1}{12}$-approximate unitary $p(|x|)$-design implies that for any fixed $|\psi\rangle$, $|\Pr[V_2(x, |\psi\rangle) = 1] - \Pr[V_3(x, |\psi\rangle) = 1]| \leq \frac{1}{12}$. This in turn implies that $|\mathrm{acc}(V_2) - \mathrm{acc}(V_3)| \leq \frac{1}{12}$.
- Zhandry [28, Theorem 6.1] shows that a quantum algorithm that makes T queries to a random function can be exactly simulated by the same algorithm with T queries to a $2T$-wise independent function, so $\mathrm{acc}(V_3) = \mathrm{acc}(V_4)$.
- Because $||\widetilde{\mathcal{U}}_n \cdot \widetilde{\mathcal{U}}_n^\dagger - \mathcal{U}_n \cdot \mathcal{U}_n^\dagger||_\diamond \leq \frac{1}{12p(|x|)}$ for each $n \in [d]$ with probability at least $\frac{2}{3}$ over $\mathcal{A}$, from the definition of the diamond norm [5] and because $\mathcal{V}$ makes at most $p(|x|)$ queries, it holds that $\Pr_{\mathcal{A}}\left[|\mathrm{acc}(V_4) - \mathrm{acc}(V_5)| \geq \frac{1}{12}\right] \leq \frac{1}{3}$.

Putting these bounds together, we have that, except with probability $2e^{-|x|}$ over $\mathcal{U}$:

$$\begin{aligned}\mathcal{L}(x) = 1 \quad &\Longrightarrow \quad \Pr_{\mathcal{A}}\left[\max_{|\psi\rangle} \Pr\left[\widetilde{\mathcal{V}}(x, |\psi\rangle) = 1\right] \geq \frac{2}{3}\right] \geq \frac{2}{3}\\ \mathcal{L}(x) = 0 \quad &\Longrightarrow \quad \Pr_{\mathcal{A}}\left[\max_{|\psi\rangle} \Pr\left[\widetilde{\mathcal{V}}(x, |\psi\rangle) = 1\right] \leq \frac{1}{3}\right] \geq \frac{2}{3}.\end{aligned}$$

This is to say that $\mathcal{A}$ correctly decides $\mathcal{L}(x)$, expect with probability at most $2e^{-|x|}$ over $\mathcal{U}$. By the Borel-Cantelli Lemma, because $\sum_{i=1}^{\infty} 2^i \cdot 2e^{-i} = \frac{4}{e-2} < \infty$, $\mathcal{A}$ correctly decides $\mathcal{L}(x)$ for all but finitely many $x \in \{0,1\}^*$, with probability 1 over $\mathcal{U}$. Hence, with probability 1 over $\mathcal{U}$, $\mathcal{A}$ can be modified into an algorithm $\mathcal{A}'$ that agrees with $\mathcal{L}$ on every $x \in \{0,1\}^*$, by simply hard-coding those x on which $\mathcal{A}$ and $\mathcal{L}$ disagree.

Because there are only countably many $\mathsf{QMA}^{\mathcal{U},\mathcal{P}}$ machines, we can union bound over all $\mathcal{L} \in \mathsf{QMA}^{\mathcal{U},\mathcal{P}}$ to conclude that $\mathsf{QMA}^{\mathcal{U},\mathcal{P}} \subseteq \mathsf{BQP}^{\mathcal{U},\mathcal{P}}$ with probability 1 over $\mathcal{U}$. ◀

4.2 PRUs Relative to $(\mathcal{U}, \mathcal{P})$

We proceed to the second part of the oracle construction, showing that PRUs exist relative to $(\mathcal{U}, \mathcal{P})$. We begin with a lemma establishing that the average advantage of a polynomial-time adversary is small against our PRU construction. Here, we should think of $\{U_k\}_{k\in[N]}$ as the PRU ensemble.

▶ **Lemma 19.** *Consider a quantum algorithm $\mathcal{A}^{O,U}$ that makes T queries to $O \in \mathbb{U}(D)$ and $U = (U_1, \ldots, U_N) \in \mathbb{U}(D)^N$. For fixed U, define:*

$$\operatorname{adv}(\mathcal{A}^U) := \Pr_{k\leftarrow[N]}\left[\mathcal{A}^{U_k,U} = 1\right] - \Pr_{O\leftarrow\mu_D}\left[\mathcal{A}^{O,U} = 1\right].$$

Then there exists a universal constant $c > 0$ such that:

$$\mathop{\mathbb{E}}_{U\leftarrow\mu_D^N}\left[\operatorname{adv}(\mathcal{A}^U)\right] \leq \frac{cT^2}{N}.$$

Proof. Our strategy is to reduce to the quantum query lower bound for unstructured search. Intuitively, if $\mathcal{A}$ could identify whether $O \in \{U_1, \ldots, U_N\}$ or not, then $\mathcal{A}$ could be modified into a quantum algorithm $\mathcal{B}$ that finds a single marked item from a list of size N. Then the BBBV theorem [10] forces T to be $\Omega\left(\sqrt{N}\right)$.

More formally, we construct an algorithm $\mathcal{B}(x)$ that queries a string $x \in \{0,1\}^N$ as follows. $\mathcal{B}$ draws a unitary $V = (V_0, V_1, \ldots, V_N) \in \mathbb{U}(D)^{N+1}$ from μ_D^{N+1}. Then, $\mathcal{B}$ runs $\mathcal{A}$, replacing queries to O by queries to V_0, and replacing queries to $U_k \in U$ by V_0 if $x_k = 1$ and by V_k if $x_k = 0$.

Let $e_k \in \{0,1\}^N$ be the string with 1 in the kth position and 0s everywhere else. We have that:

$$\begin{aligned}\mathop{\mathbb{E}}_{U\leftarrow\mu_D^N}\left[\operatorname{adv}(\mathcal{A}^U)\right] &= \mathop{\mathbb{E}}_{U\leftarrow\mu_D^N}\left[\Pr_{k\leftarrow[N]}\left[\mathcal{A}^{U_k,U} = 1\right]\right] - \mathop{\mathbb{E}}_{U\leftarrow\mu_D^N}\left[\Pr_{O\leftarrow\mu_D}\left[\mathcal{A}^{O,U} = 1\right]\right]\\ &= \Pr_{k\leftarrow[N]}\left[\mathcal{B}\left(e_k\right) = 1\right] - \Pr\left[\mathcal{B}\left(0^N\right) = 1\right]\\ &\leq \frac{cT^2}{N}.\end{aligned}$$

Above, the first line applies linearity of expectation, the second line holds by definition of $\mathcal{B}$, and the third line holds for some universal c by the BBBV theorem [10]. ◀

The next lemma uses Lemma 19 to show that the advantage of $\mathcal{A}$ is small with extremely high probability, which follows from the strong concentration properties of the Haar measure (Theorem 11).

▶ **Lemma 20.** *Consider a quantum algorithm $\mathcal{A}^{O,U}$ that makes T queries to $O \in \mathbb{U}(D)$ and $U = (U_1, \ldots, U_N) \in \mathbb{U}(D)^N$. Let $\operatorname{adv}(\mathcal{A}^U)$ be defined as in Lemma 20. Then there exists a universal constant $c > 0$ such that for any p,*

$$\Pr_{U\leftarrow\mu_D^N}\left[\left|\operatorname{adv}(\mathcal{A}^U)\right| \geq p\right] \leq 2\exp\left(-\frac{(D-2)\left(p - cT^2/N\right)^2}{384T^2}\right).$$

Proof. By Lemma 16, $\operatorname{adv}(\mathcal{A}^U)$ is $4T$-Lipschitz as a function of U, because $\operatorname{adv}(\mathcal{A}^U)$ can be expressed as the the difference between the acceptance probabilities of two algorithms that each make T queries to U. Combining Lemma 19 and Theorem 11, we obtain:

$$\Pr_{U\leftarrow\mu_D^N}\left[\operatorname{adv}(\mathcal{A}^U) \geq p\right] \leq \exp\left(-\frac{(D-2)\left(p - cT^2/N\right)^2}{384T^2}\right).$$

Similar reasoning yields the same upper bound on $\Pr_{U \leftarrow \mu_D^N}\left[\text{adv}(\mathcal{A}^U) \leq -p\right]$, so we get the final bound (with an additional factor of 2) by a union bound. ◀

Completing the security proof of the PRU construction amounts to combining Lemma 20 with a union bound over all possible polynomial-time adversaries.

▶ **Theorem 21.** *Let $\mathcal{U} = \{\mathcal{U}_n\}_{n\in\mathbb{N}}$ be a quantum oracle where each $\mathcal{U}_n$ is chosen randomly from $\mu_{2^n}^{2^n}$. Let $\mathcal{P}$ be an arbitrary* PSPACE*-complete language. Then with probability* 1 *over $\mathcal{U}$, there exists a family of PRUs relative to $(\mathcal{U}, \mathcal{P})$.*

Proof. Fix an input length $n \in \mathbb{N}$. We take the key set $\mathcal{K} = [2^n]$ and take the PRU family to be $\{U_k\}_{k\in\mathcal{K}}$, where $\mathcal{U}_n = (U_1, U_2, \ldots, U_{2^n}) \in \mathbb{U}(2^n)^{2^n}$. In words, the family consists of the 2^n different Haar-random n-qubit unitaries specified by $\mathcal{U}_n$.

Without loss of generality, assume the adversary is a uniform polynomial-time quantum algorithm $\mathcal{A}^{O,\mathcal{U},\mathcal{P}}(1^n, x)$, where $x \in \{0,1\}^{\text{poly}(n)}$ is the advice and $O \in \mathbb{U}(2^n)$ is the oracle that the adversary seeks to distinguish as pseudorandom or Haar-random.

By Lemma 20 with $N = D = 2^n$ and $T = \text{poly}(n)$, for any fixed $x \in \{0,1\}^{\text{poly}(n)}$, $\mathcal{A}^{O,\mathcal{U},\mathcal{P}}(1^n, x)$ achieves non-negligible advantage with extremely low probability over $\mathcal{U}$. This is to say that for any $p = \frac{1}{\text{poly}(n)}$:

$$\Pr_{\mathcal{U}_n \leftarrow \mu_{2^n}^{2^n}}\left[\left|\Pr_{k\leftarrow[2^n]}\left[\mathcal{A}^{U_k,\mathcal{U},\mathcal{P}}(1^n,x)=1\right] - \Pr_{O\leftarrow\mu_{2^n}}\left[\mathcal{A}^{O,\mathcal{U},\mathcal{P}}(1^n,x)=1\right]\right| \geq p\right] \leq \exp\left(-\frac{2^n}{\text{poly}(n)}\right).$$

By a union bound over all $x \in \{0,1\}^{\text{poly}(n)}$, $\mathcal{A}^{O,\mathcal{U},\mathcal{P}}(1^n, x)$ achieves advantage larger than p for *any* $x \in \{0,1\}^{\text{poly}(n)}$ with probability at most $2^{\text{poly}(n)} \cdot \exp\left(-\frac{2^n}{\text{poly}(n)}\right) \leq \text{negl}(n)$. Hence, by the Borel-Cantelli lemma, $\mathcal{A}$ achieves negligible advantage for all but finitely many input lengths $n \in \mathbb{N}$ with probability 1 over $\mathcal{U}$, as $\sum_{n=1}^{\infty} \text{negl}(n) < \infty$. This is to say that $\{U_k\}_{k\in\mathcal{K}}$ defines a PRU ensemble. ◀

We expect that using the techniques of Chung, Guo, Liu, and Qian [16], one can extend Theorem 21 to a security proof against adversaries with quantum advice. Some version of [16, Theorem 5.14] likely suffices. The idea is that breaking the PRU should remain hard even if $\mathcal{A}$ could query an explicit description of O and explicit descriptions of U_k for $k \in [2^n]$, which is a strictly more powerful model. But then this corresponds to the security game defined in [16, Definition 5.12], except that the range of the random oracle is $\mathbb{U}(D)$ rather than the finite set $[M]$.

References

1 Scott Aaronson. Quantum computing, postselection, and probabilistic polynomial-time. *Proceedings of the Royal Society A*, 461:3473–3482, 2005.

2 Scott Aaronson. Shadow tomography of quantum states. In *Proceedings of the 50th Annual ACM SIGACT Symposium on Theory of Computing*, STOC 2018, pages 325–338, New York, NY, USA, 2018. Association for Computing Machinery.

3 Scott Aaronson and Daniel Gottesman. Improved simulation of stabilizer circuits. *Phys. Rev. A*, 70:052328, November 2004.

4 Scott Aaronson and Greg Kuperberg. Quantum versus classical proofs and advice. *Theory of Computing*, 3(7):129–157, 2007.

5 Dorit Aharonov, Alexei Kitaev, and Noam Nisan. Quantum circuits with mixed states. In *Proceedings of the Thirtieth Annual ACM Symposium on Theory of Computing*, STOC '98, pages 20–30, New York, NY, USA, 1998. Association for Computing Machinery.

6 Eric Allender. *The Complexity of Complexity*, pages 79–94. Springer International Publishing, Cham, 2017.

7 Eric Allender. The new complexity landscape around circuit minimization. In Alberto Leporati, Carlos Martín-Vide, Dana Shapira, and Claudio Zandron, editors, *Language and Automata Theory and Applications*, pages 3–16, Cham, 2020. Springer International Publishing.

8 Andris Ambainis and Joseph Emerson. Quantum t-designs: T-wise independence in the quantum world. In *Proceedings of the Twenty-Second Annual IEEE Conference on Computational Complexity*, CCC '07, pages 129–140, USA, 2007. IEEE Computer Society.

9 Aleksandrs Belovs and Ansis Rosmanis. Tight quantum lower bound for approximate counting with quantum states, 2020.

10 Charles H. Bennett, Ethan Bernstein, Gilles Brassard, and Umesh Vazirani. Strengths and weaknesses of quantum computing. *SIAM Journal on Computing*, 26(5):1510–1523, 1997.

11 James O. Berger. *Statistical Decision Theory and Bayesian Analysis.* Springer Series in Statistics. Springer New York, 2013.

12 Adam Bouland, Bill Fefferman, and Umesh Vazirani. Computational Pseudorandomness, the Wormhole Growth Paradox, and Constraints on the AdS/CFT Duality (Abstract). In Thomas Vidick, editor, *11th Innovations in Theoretical Computer Science Conference (ITCS 2020)*, volume 151 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 63:1–63:2, Dagstuhl, Germany, 2020. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik.

13 Zvika Brakerski and Omri Shmueli. (Pseudo) random quantum states with binary phase. In Dennis Hofheinz and Alon Rosen, editors, *Theory of Cryptography*, pages 229–250, Cham, 2019. Springer International Publishing.

14 Fernando G. S. L. Brandão, Aram W. Harrow, and Michał Horodecki. Efficient quantum pseudorandomness. *Phys. Rev. Lett.*, 116:170502, April 2016.

15 Fernando G. S. L. Brandão, Aram W. Harrow, and Michał Horodecki. Local random quantum circuits are approximate polynomial-designs. *Communications in Mathematical Physics*, 346(2):397–434, 2016.

16 Kai-Min Chung, Siyao Guo, Qipeng Liu, and Luowen Qian. Tight quantum time-space tradeoffs for function inversion. In *2020 IEEE 61st Annual Symposium on Foundations of Computer Science (FOCS)*, pages 673–684, 2020.

17 Hsin-Yuan Huang, Richard Kueng, and John Preskill. Predicting many properties of a quantum system from very few measurements. *Nature Physics*, 2020.

18 Russell Impagliazzo and Steven Rudich. Limits on the provable consequences of one-way permutations. In *Proceedings of the Twenty-First Annual ACM Symposium on Theory of Computing*, STOC '89, pages 44–61, New York, NY, USA, 1989. Association for Computing Machinery.

19 Zhengfeng Ji, Yi-Kai Liu, and Fang Song. Pseudorandom quantum states. In Hovav Shacham and Alexandra Boldyreva, editors, *Advances in Cryptology – CRYPTO 2018*, pages 126–152, Cham, 2018. Springer International Publishing.

20 Robert Koenig and John A. Smolin. How to efficiently select an arbitrary Clifford group element. *Journal of Mathematical Physics*, 55(12):122202, 2014.

21 Greg Kuperberg. How hard is it to approximate the Jones polynomial? *Theory of Computing*, 11(6):183–219, 2015.

22 Elizabeth S. Meckes. *The Random Matrix Theory of the Classical Compact Groups.* Cambridge Tracts in Mathematics. Cambridge University Press, 2019.

23 Ryan O'Donnell and John Wright. Efficient quantum tomography. In *Proceedings of the Forty-Eighth Annual ACM Symposium on Theory of Computing*, STOC '16, pages 899–912, New York, NY, USA, 2016. Association for Computing Machinery.

24 Leonard Susskind. Addendum to computational complexity and black hole horizons. *Fortschritte der Physik*, 64(1):44–48, 2016.

25 Leonard Susskind. Computational complexity and black hole horizons. *Fortschritte der Physik*, 64(1):24–43, 2016.

26 Juha J. Vartiainen, Mikko Möttönen, and Martti M. Salomaa. Efficient decomposition of quantum gates. *Phys. Rev. Lett.*, 92:177902, April 2004.

27 Yaakov S. Weinstein, Winton G. Brown, and Lorenza Viola. Parameters of pseudorandom quantum circuits. *Phys. Rev. A*, 78:052332, November 2008.

28 Mark Zhandry. Secure identity-based encryption in the quantum random oracle model. In Reihaneh Safavi-Naini and Ran Canetti, editors, *Advances in Cryptology – CRYPTO 2012*, pages 758–775, Berlin, Heidelberg, 2012. Springer Berlin Heidelberg.

A Proof of Lemma 6

Proof. Let $\{\lambda_i : i \in [N]\}$ denote the eigenvalues of $UV^\dagger$. Then we have:

$$\begin{aligned}
||U - V||_F^2 &= \mathrm{Tr}\left((U-V)(U-V)^\dagger\right) \\
&= \mathrm{Tr}(2I - UV^\dagger - VU^\dagger) \\
&= 2N - \sum_{i=1}^{N}(\lambda_i + \lambda_i^*) \\
&= \sum_{i=1}^{N}(2 - 2\mathrm{Re}(\lambda_i)) \\
&\geq \max_i (2 - 2\mathrm{Re}(\lambda_i)), \qquad (1)
\end{aligned}$$

where $\mathrm{Re}(\lambda_i)$ denotes the real part of λ_i. The last line holds because the eigenvalues of a unitary matrix have absolute value 1. Aharonov, Kitaev, and Nisan [5] show that $||U \cdot U^\dagger - V \cdot V^\dagger||_\diamond = 2\sqrt{1-d^2}$, where d is the distance in the complex plane between 0 and the polygon whose vertices are $\lambda_1, \ldots, \lambda_N$. From this we may conclude:

$$\begin{aligned}
||U \cdot U^\dagger - V \cdot V^\dagger||_\diamond &\leq \max_i 2\sqrt{1 - \max\{\mathrm{Re}(\lambda_i), 0\}^2} \\
&\leq \max_i 2\sqrt{2 - 2\mathrm{Re}(\lambda_i)} \\
&\leq 2||U - V||_F,
\end{aligned}$$

where the first inequality uses the fact that either all of the eigenvalues have positive real components and therefore $d \geq \min_i \mathrm{Re}(\lambda_i)$, or else $d \geq 0$; the second inequality substitutes $1 - \max\{x, 0\}^2 \leq 2 - 2x$ which holds for all $x \in \mathbb{R}$; and the third inequality substitutes (1). ◀

B Complexity Classes

▶ **Definition 22.** *A promise problem* $\mathcal{L} = (\mathcal{L}_{\text{yes}}, \mathcal{L}_{\text{no}})$ *is in* QMA *if there exists a polynomial-time quantum algorithm* $\mathcal{V}(x, |\psi\rangle)$ *called a* QMA verifier *and a polynomial* p *such that:*

1. *(Completeness) If* $x \in \mathcal{L}_{\text{yes}}$*, then there exists a state* $|\psi\rangle$ *(called a* witness *or* proof*) on* $p(|x|)$ *qubits such that* $\Pr[\mathcal{V}(x, |\psi\rangle) = 1] \geq \frac{2}{3}$.
2. *(Soundness) If* $x \in \mathcal{L}_{\text{no}}$*, then for every state* $|\psi\rangle$ *on* $p(|x|)$ *qubits,* $\Pr[\mathcal{V}(x, |\psi\rangle) = 1] \leq \frac{1}{3}$.

Aaronson [1] defined PostBQP as follows, and showed that PostBQP = PP.

▶ **Definition 23.** *A promise problem* $\mathcal{L} = (\mathcal{L}_{\text{yes}}, \mathcal{L}_{\text{no}})$ *is in* PostBQP *if there exists a polynomial-time quantum algorithm* $\mathcal{A}(x)$ *that outputs a trit* $\{0, 1, *\}$ *such that:*

1. *If* $x \in \mathcal{L}_{\text{yes}} \cup \mathcal{L}_{\text{no}}$*, then* $\Pr[\mathcal{A}(x) \in \{0,1\}] > 0$*. When* $\mathcal{A}(x) \in \{0,1\}$*, we say that* postselection succeeds*.*
2. *If* $x \in \mathcal{L}_{\text{yes}}$*, then* $\Pr[\mathcal{A}(x) = 1 \mid \mathcal{A}(x) \in \{0,1\}] \geq \frac{2}{3}$*. In other words, conditioned on postselection succeeding,* $\mathcal{A}$ *outputs* 1 *with at least* $\frac{2}{3}$ *probability.*
3. *If* $x \in \mathcal{L}_{\text{no}}$*, then* $\Pr[\mathcal{A}(x) = 1 \mid \mathcal{A}(x) \in \{0,1\}] \leq \frac{1}{3}$*. In other words, conditioned on postselection succeeding,* $\mathcal{A}$ *outputs* 1 *with at most* $\frac{1}{3}$ *probability.*

Technically, the definition of PostBQP is sensitive to the choice of universal gate set used to specify quantum algorithms, as was observed by Kuperberg [21]. However, for most "reasonable" gate sets, such as unitary gates with algebraic entries [21], the choice of gate set is irrelevant. We assume such a gate set, e.g. $\{\text{CNOT}, H, T\}$.

We consider versions of BQP, QMA, and PostBQP augmented with quantum oracles, where the algorithm (or in the case of QMA, the verifier) can apply unitary transformations from an infinite sequence $\mathcal{U} = \{\mathcal{U}_n\}_{n \in \mathbb{N}}$. We denote the respective complexity classes by $\mathsf{BQP}^{\mathcal{U}}$, $\mathsf{QMA}^{\mathcal{U}}$, and $\mathsf{PostBQP}^{\mathcal{U}}$. We assume the algorithm incurs a cost of n to query $\mathcal{U}_n$ so that a polynomial-time algorithm on input x can query $\mathcal{U}_n$ for any $n \leq \text{poly}(|x|)$. In this model, a query to $\mathcal{U}_n$ consists of a single application of either $\mathcal{U}_n$, $\mathcal{U}_n^\dagger$, or controlled versions of $\mathcal{U}_n$ or $\mathcal{U}_n^\dagger$.

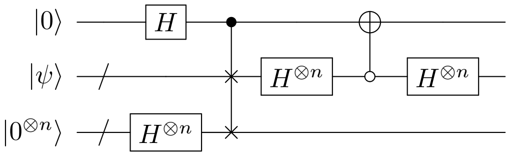
The quantum oracle model includes classical oracles as a special case. For a language $\mathcal{L}$, a query to $\mathcal{L}$ is implemented via the unitary transformation $\mathcal{U}$ that acts as $\mathcal{U}\,|x\rangle\,|b\rangle = |x\rangle\,|b \oplus \mathcal{L}(x)\rangle$.

C PRSs with Binary Phases

In this section, we sketch a proof that a PRS construction proposed by Ji, Liu, and Song [19] and shown secure by Brakerski and Shmueli [13] can be broken efficiently with an NP oracle. The PRS family is based on pseudorandom functions (PRFs). Let $\{f_k\}_{k \in \mathcal{K}}$ be a PRF family of functions $f_k : \{0,1\}^n \to \{0,1\}$ keyed by $\mathcal{K}$. The corresponding PRS family is the set of states $\{|\varphi_k\rangle\}_{k \in \mathcal{K}}$ given by:

$$|\varphi_k\rangle = \frac{1}{2^{n/2}} \sum_{x \in \{0,1\}^n} (-1)^{f_k(x)}\,|x\rangle\,.$$

For simplicity, suppose that each f_k is *balanced*, meaning that $|f_k^{-1}(0)| = |f_k^{-1}(1)| = 2^{n-1}$. Consider the quantum circuit below:



Observe that if $|\psi\rangle = |\varphi_k\rangle$, then this circuit produces the state $|0\rangle \frac{|\varphi_k\rangle|+\rangle^{\otimes n} + |+\rangle^{\otimes n}|\varphi_k\rangle}{\sqrt{2}}$ from a single copy of $|\varphi_k\rangle$. Notice that if we measure the resulting state in the computational basis, then we observe $|0\rangle\,|x\rangle\,|y\rangle$ with nonzero probability for $x, y \in \{0,1\}^n$ if and only if $f_k(x) = f_k(y)$. This is because the amplitude on this basis state is given by:

$$\langle x|\,\langle y|\,\frac{|\varphi_k\rangle\,|+\rangle^{\otimes n} + |+\rangle^{\otimes n}\,|\varphi_k\rangle}{\sqrt{2}} = \frac{(-1)^{f_k(x)} + (-1)^{f_k(y)}}{2^n\sqrt{2}}.$$

Furthermore, this shows that we in fact sample a uniformly random pair (x, y) such that $f_k(x) = f_k(y)$.

Suppose that given a state $|\psi\rangle$ which is either pseudorandom or Haar-random, we repeat this procedure $\mathrm{poly}(n)$ times to obtain a list of pairs $\{(x_i, y_i)\}$. It is an NP problem to decide whether there exists a k such that $f_k(x_i) = f_k(y_i)$ for all i. If $|\psi\rangle = |\varphi_k\rangle$ for some k then this NP language always returns true, while if $|\psi\rangle$ is Haar-random, this NP language returns true with negligible probability, so long as we take sufficiently many samples (x_i, y_i).

In the case where f_k is not perfectly balanced, we simply observe that the above procedure still works with good probability so long as f_k is *close* to a balanced function. But PRFs must be close to balanced functions, in the sense that for most $k \in \mathcal{K}$, it must be possible to change a $\mathrm{negl}(n)$ fraction of the outputs of f_k to turn it into a balanced function. Otherwise, the PRF family could be distinguished efficiently from random functions, which are $\mathrm{negl}(n)$-close to balanced with high probability.

Sample Efficient Algorithms for Learning Quantum Channels in PAC Model and the Approximate State Discrimination Problem

Kai-Min Chung
Institute of Information Science, Academia Sinica, Taipei, Taiwan

Han-Hsuan Lin
Department of Computer Science, National Tsing Hua University, Hsinchu, Taiwan

Abstract

The probably approximately correct (PAC) model [30] is a well studied model in classical learning theory. Here, we generalize the PAC model from concepts of Boolean functions to quantum channels, introducing *PAC model for learning quantum channels*, and give two sample efficient algorithms that are analogous to the classical "Occam's razor" result [12]. The classical Occam's razor algorithm is done trivially by excluding any concepts not compatible with the input-output pairs one gets, but such an approach is not immediately possible with a concept class of quantum channels, because the outputs are unknown quantum states from the quantum channel.

To study the quantum state learning problem associated with PAC learning quantum channels, we focus on the special case where the channels all have constant output. In this special case, learning the channels reduce to a problem of learning quantum states that is similar to the well known quantum state discrimination problem [8], but with the extra twist that we allow ϵ-trace-distance-error in the output. We call this problem *Approximate State Discrimination*, which we believe is a natural problem that is of independent interest.

We give two algorithms for learning quantum channels in PAC model. The first algorithm has sample complexity

$$O\left(\frac{\log|C| + \log(1/\delta)}{\epsilon^2}\right),$$

but only works when the outputs are pure states, where C is the concept class, ϵ is the error of the output, and δ is the probability of failure of the algorithm. The second algorithm has sample complexity

$$O\left(\frac{\log^3|C|(\log|C| + \log(1/\delta))}{\epsilon^2}\right),$$

and work for mixed state outputs. Some implications of our results are that we can PAC-learn a polynomial sized quantum circuit in polynomial samples, and approximate state discrimination can be solved in polynomial samples even when the size of the input set is exponential in the number of qubits, exponentially better than a naive state tomography.

2012 ACM Subject Classification Theory of computation → Quantum information theory; Theory of computation → Quantum complexity theory

Keywords and phrases PAC learning, Quantum PAC learning, Sample Complexity, Approximate State Discrimination, Quantum information

Digital Object Identifier 10.4230/LIPIcs.TQC.2021.3

Related Version *Full Version*: https://arxiv.org/abs/1810.10938

Funding *Kai-Min Chung*: This research is partially supported by the 2016 Academia Sinica Career Development Award under Grant no. 23-17.
Han-Hsuan Lin: This material is based on research/work supported by the Singapore National Research Foundation under NRF RF Award No. NRF-NRFF2013-13.

16th Conference on the Theory of Quantum Computation, Communication and Cryptography (TQC 2021).
Editor: Min-Hsiu Hsieh; Article No. 3; pp. 3:1–3:22
Leibniz International Proceedings in Informatics
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

1 Introduction

In computational learning theory, the Probably Approximately Correct (PAC) model of Valiant [30] gives a complexity-theoretic foundation of what it means for a concept class to be (efficiently) learnable. In the most basic setting of PAC learning model, we want to learn a set of Boolean functions, $C = \{c : \{0,1\}^n \to \{0,1\}\}$, called the concept class. The goal of a learning algorithm A is to guess the identity of an unknown target concept $c^* \in C$ from samples $\{(x_1, c^*(x_1)), (x_2, c^*(x_2)), \dots\}$, where $\{x_1, x_2, \dots\}$ are inputs randomly drawn from a distribution D that is unknown to A. Specifically, with error parameters ϵ and δ, for all concept $c^* \in C$ and probability distribution D, A is required to, given access to the samples $\{(x_1, c^*(x_1)), (x_2, c^*(x_2)), \dots\}$, with probability $1-\delta$, come up with a hypothesis $h \in C$ that is ϵ-close to c^*, i.e. $\Pr_{x \leftarrow D}[c(x) \neq h(x)] \leq \epsilon$. Such a learning algorithm is called a proper[1] (ϵ, δ)-PAC learner for the concept class C. Of course, we would like the learner A to be as efficient as possible in terms of both sample complexity (i.e., the number of samples A needs to access) and time complexity, and ideally, polynomial in the input length n and the error parameters ϵ^{-1} and $\log(1/\delta)$. Since its introduction in the 80's by Valiant, PAC learning theory has been deeply studied to characterize when efficient learning is or is not possible.

Following Valient's PAC learning model on Boolean functions, generalization to different kinds of concept classes has been proposed, including Boolean functions on continuous spaces [13], probabilistic Boolean functions [20, 2], functions with $\{0, \dots, n\}$ outputs [26, 11], and real valued functions [10].

With quantum computers coming closer and closer into reality, it is natural to generalize the PAC learning model to quantum channels, capturing the learnability of quantum circuits or devices that we might build in the near future. Note that quantum states has an inherent "unlearnability", as manifested by the no-cloning theorem and uncertainty principle. Therefore this study of learnability of quantum channels has an interesting interaction between classical learning theory and quantum information theory.

Formally, we define the *PAC learning model for quantum channels* as follows: Let the *concept class* C be a finite set of known d_1 to d_2 dimensional quantum channels. We are trying to learn an unknown quantum channel, the *target concept* $c^* \in C$. In order to do this, we are given *samples* $\{(x_1, c^*(x_1)), (x_2, c^*(x_2)), \dots\}$, where $\{x_1, x_2, \dots\}$ are classical descriptions of the input quantum states to the quantum channel c^* and $\{c^*(x_1), c^*(x_2), \dots\}$ are the corresponding quantum states outputted by c^*. The inputs are drawn from a distribution D unknown to the learner. Because of the no-cloning theorem, it is hard to justify holding both the inputs and outputs as unknown quantum states, so we assume that we have full classical description of the input state and keep the outputted states as unknown quantum states, meaning that we hold a copy of the quantum state $c^*(x_i)$ rather than the full classical description of it. A proper (ϵ, δ)-PAC learner for the concept class C of quantum channels is a quantum algorithm that for all concepts $c^* \in C$ and distribution D, takes the description of C and T samples $\{(x_1, c^*(x_1)), (x_2, c^*(x_2)), \dots (x_T, c^*(x_T))\}$ as input[2] and with probability $1-\delta$, outputs a *hypothesis* $h \in C$ that is ϵ-close to the target concept c^*, where the distance between two concepts h, c^* depends on the input distribution D and is defined as $\Delta(h, c^*) = \mathbb{E}_{x \in D}[\Delta_{tr}(h(x), c^*(x))]$, i.e. the expected trace distance between the outputs averaged over D.

[1] Proper means that the hypothesis h must be inside the concept class C, whereas an improper learner can output any h as the hypothesis. All learners in this paper are proper, and we sometimes omit the term "proper".

[2] Note that D is not part of the input and is unknown to the learner.

We gave two algorithms for learning quantum channels in PAC model that in a sense generalize the classical Occam's razor algorithm [12]. In particular, our algorithms have poly log sample complexity in the size of the concept class. The first algorithm has sample complexity

$$O\left(\frac{\log|C|+\log(1/\delta)}{\epsilon^2}\right),$$

but requires the outputs to be pure states. The second algorithm has sample complexity

$$O\left(\frac{\log^3|C|(\log|C|+\log(1/\delta))}{\epsilon^2}\right),$$

while outputs can be mixed.

The Occam's razor algorithm [12] is a classical PAC learner for any finite sized concept class C with sample complexity $O(\log|C|)$. The idea of the algorithm is simple: keep taking samples, check which concepts in the concept class do not agree with the samples and exclude them. One can show that every time a sample is taken, a constant fraction of the concepts that are ϵ-far away from the target concept will be excluded, so an ϵ-close hypothesis can be found in $O(\log|C|)$ samples.

Although the Occam's razor algorithm is simple, generalizing it to our PAC model for quantum channels is troublesome. The main difference is that when learning quantum channels, the outputs from the target concept are copies of unknown (possibly high dimensional) quantum states. By the nature of quantum mechanics, if we just have a few copies of a high dimensional quantum state, we can only learn a tiny fraction of information contained in the quantum state. Since we don't really know what the outputted state is, we cannot simply "exclude all channels that do not output this state." Instead, we need to carefully design the measurement we take on the outputted states, getting the information useful in distinguishing the quantum channels in our concept class. Note that the sample complexities of both of our algorithms do not depend on the dimension of the outputted states.

As a possible application of our result, our algorithms for learning quantum channels in PAC model can be viewed as a sample-efficient way to do quantum process tomography [23] when we know that the target quantum processes comes from a finite set and only care about being correct on average over an input distribution. For example, if we try to PAC-learn a polynomial sized quantum circuit of n-qubits, since there are only $2^{\mathrm{poly}(n)}$ possible polynomial sized circuits, our result shows that we can learn it in $\mathrm{poly}(n)$ samples, an exponential improvement over a naive process tomography that has no restriction on concept class size and inputs.

Note that this work studies the sample complexity instead of time complexity of learning. Just like various other cases in theoretical computer science where the oracle-based complexity does not match the time complexity of a problem, sample complexity and time complexity of learning quantum channels in PAC model is unlikely to match. In particular, Arunachalam et al. [5] showed that there is no polynomial time algorithm for learning TC^0 or AC^0 circuit even knowing D is uniform unless LWE can be solved in polynomial time by a quantum computer.

1.1 Approximate State Discrimination

As stated previously, the most challenging part of our algorithms is how to extract information from unknown outputted quantum states to distinguish the channels. We isolate and study this problem by focusing on the special case where the channels are "constant," i.e. every channel in the concept class outputs a fixed quantum state irrespective of the input[3]. Since the input does not matter, we don't need to write it down anymore, so the samples are just copies of the fixed unknown quantum state, and since a concept is fully specified by its unique output state, we might as well describe the concept class as a set of quantum states. In this special case, learning quantum channels in PAC model becomes an interesting hybrid of quantum state discrimination [6, 25, 24, 8, 29] and quantum state tomography [15, 27], and we named it the *approximate state discrimination* problem. The approximate state discrimination problem is formalized as follows: Let S be a known finite set of d-dimensional density matrices. We want to learn an unknown target state $\sigma \in S$ using as few identical copies of σ as possible. A quantum algorithm is an (ϵ, δ)-approximate discriminator of S if, for all $\sigma \in S$, it takes the description of S and T copies of σ as input and with probability $1-\delta$ outputs a state $\rho \in S$ with $\Delta_{tr}(\rho, \sigma) \leq \epsilon$. This problem is called approximate state discrimination because it is the same as the state discrimination problem except that ϵ-approximate answers are allowed.

Since approximate state discrimination is a special case of PAC learning quantum channels[4], it can also be solved with

$$O\left(\frac{\log|S| + \log(1/\delta)}{\epsilon^2}\right)$$

samples if S consists of pure states and

$$O\left(\frac{\log^3|S|(\log|S| + \log(1/\delta))}{\epsilon^2}\right)$$

samples if S consists of mixed states.

1.2 Related Works and Independent Work

There are several works in the literature that study the sample complexity of PAC learning with different ways of generalization to quantum information. Cheng, Hsieh, and Yeh [14] studies the sample complexity of PAC learning arbitrary two outcome measurements, where the inputs are quantum states, and the learner has complete classical description of them. They show an upper of sample complexity linear in the dimension of the Hilbert space. Note that one can trivially get a lower bound of similar order by noticing that Boolean functions is a subset of two outcome measurements. Arunachalam and de Wolf [4] studies the sample complexity of PAC learning classical functions with quantum samples and shows that there is no quantum speed up. See [3] for a survey of quantum learning theory.

1.2.0.1 Independent Work

Independent to our work, in [7], Bădescu and O'Donnell formulate the problem of *quantum hypothesis selection*. Quantum hypothesis selection can be viewed as a generalization of our approximate state discrimination problem where the unknown state σ might not be in the hypothesis set $|S|$, and the learner what to find the state in $|S|$ that is closest to the

[3] The outputs of different concepts are still different.

[4] We choose not to write up stand-alone algorithms for the approximate state discrimination problem as it will be very similar to that of PAC learning quantum channels. However, the reader can read the analysis of our algorithms with constant output assumptions to easily get the intuition behind them.

unknown state σ (see Theorem 1.5 of [7] for the formal definition). This is similar to the agnostic learning model [18, 21]. Let η be the minimum distance from the unknown state to something in $|S|$, Bădescu and O'Donnell give an algorithm that finds some $\rho \in S$ such that $\Delta_{tr}(\rho, \sigma) \leq 3.01\eta + \epsilon$ using $O\left(\frac{\log^3 |S| + \log(1/\delta)}{\epsilon^2}\right)$ samples. Since quantum hypothesis selection is a generalization of approximate state discrimination, Bădescu and O'Donnell's algorithm supersedes our algorithm for approximate state discrimination for the mixed state.

However, it is important to note that Bădescu and O'Donnell's algorithm requires many identical copies of the unknown state and thus does not generalize to our main result of PAC learning of quantum channels because every channel output might be a different state. On the other hand, as will shown in the following technical overview, our approach for approximate state discrimination involves a binary search through gap amplification and pretty good measurement and generalizes naturally to the PAC learning of quantum channels.

In [1], Aharonov, Cotler, and Qi introduced the notion of *quantum algorithmic measurement*, which broadly captures the query and computational complexity of quantum experiments, including those that generate unknown identical quantum states. In [19], Huang, Kueng, and Preskill compared the complexity of classically or quantumly training a machine learning model for predicting outcomes of physical experiments.

1.3 Technical Overview

The intuition behind both of our learning algorithms start with looking at the tensor product of all outputted states. The fidelity between such tensor produces decays exponentially in the number of samples drawn, so with enough samples , the tensor products from ϵ-far concepts will become almost orthogonal (see Lemma 4), so intuitively, we should be able to distinguish between them.

1.3.0.1 Pure State algorithm

In the case where the channels always output pure states, we have a rather simple algorithm. The key part is a theorem by Sen [28] on high dimensional random orthonormal measurements, which states that if we do a measurement of random orthonormal basis on two pure states, with high probability[5], the trace distance between the distribution of measurement outcome is lower bounded by a constant times the trace distance between those two states (see Theorem 11). This result might seem counter-intuitive, but remember that a random orthonormal measurement in d dimension has d possible outputs instead of 2. With this theorem in hand, the algorithm is rather easy: take enough samples to amplify the distance between outputted states and do a random orthonormal measurement on each sample. Choose the hypothesis as the channel that most likely to give the measurement result.

1.3.0.2 Mixed State algorithm

Our thought process on designing a learner for the channels that output mixed states is the following. In this case, Theorem 11 does not give us a useful result, so we need to find something else. Noticing the connection to the quantum state discrimination problem, we turned to pretty good measurement (PGM, Definition 3), a well studied tool for solving the quantum state discrimination problem. However, the lack of minimum distance between our outputted states is pretty pathological to PGM, so it was pretty easy to self-reject all our

[5] The probability goes to 1 as the dimension goes to infinity.

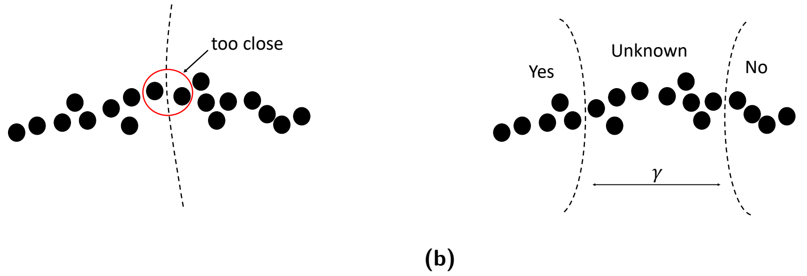


Figure 1 (a) Pathological case when trying to cut the concept class into two sets. (b) Cutting the concept class into three sets.

attempts. Following that, we sought guidance from the analysis of classical Occam's razor algorithm, where a constant fraction of concepts are ruled out by each sample. We tried to divide the concept class into two sets, then do a PGM to distinguish those two, so we can recurse this into a binary search. Cutting the concepts into two sets does not work either because there can be concepts really close to any cut, which again is pathological to PGM. At this point, we realized that we need to have some kind of minimum distance for our PGM, so we cut the concept class into *three* sets, S_{yes}, S_{no}, and $S_{unknown}$. We set a minimum distance γ between elements of S_{yes} and S_{no}, so those two sets can be distinguished. This is the idea that works out. See Figure 1 for a graphical representation.

To follow our intuition in the previous paragraph, we give a definition about the distance between two sets of quantum states. Actually fidelity is more useful than trace distance, so we give the following definition of fidelity between sets of quantum states, which is the maximum fidelity among all pairs:

$$F\left(S_{yes}, S_{no}\right) = \max\left\{F(\sigma,\rho) | \sigma \in S_{yes}, \rho \in S_{no}\right\}$$

1.3.0.3 Bichromatic State Discrimination Problem (BSD)

The key component our mixed state algorithm is solving what we called (η, N)-*Bichromatic State Discrimination Problem* (BSD). The (η, N)-Bichromatic State Discrimination Problem is defined as follows: given complete information of two sets of quantum states, S_{yes} and S_{no}, with fidelity $F(S_{yes}, S_{no}) \leq \eta$ and size $S_{yes} \leq N$, $S_{no} \leq N$, and one copy of an unknown quantum state σ, the goal is to decide whether $\sigma \in S_{yes}$ or $\sigma \in S_{no}$. A quantum algorithm solves (η, N)-BSD with error δ if for all S_{yes} and S_{no} such that $F(S_{yes}, S_{no}) \leq \eta$, $S_{yes} \leq N$, and $S_{no} \leq N$, given complete information about S_{yes} and S_{no} and one copy of an unknown quantum state σ as input to the algorithm, the algorithm output a *yes*/*no* answer satisfies the following two conditions[6]:

1. If $\sigma \in S_{yes}$, the learner outputs *yes* with probability $(1-\delta)$.
2. If $\sigma \in S_{no}$, the learner outputs *no* with probability $(1-\delta)$.

See Figure 2 for some graphical intuition of BSD.

Note that BSD only requires maximum fidelity *between* the two sets; two states from the same set can be arbitrarily close. This does not violate quantum state discrimination lower bounds because the solver only needs to discriminate between the two sets.

We are able to show that BSD can be solved with good enough parameter:

[6] The learner can output anything if σ does not come from either of the two sets.

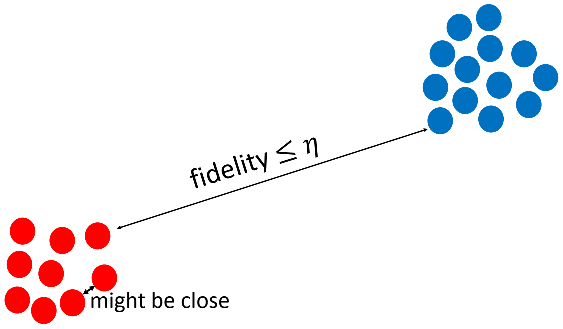


Figure 2 Bichromatic State Discrimination Problem.

▶ **Theorem 1.** *There exist an algorithm that solves* (η, N)*-BSD with error* $\delta = N^2\eta$*.*

The proof idea of Theorem 1 is trying to apply PGM on $S_{yes} \cup S_{no}$. We start with the observation that the result of [6] and [9], which gives an upper bound on PGM's error probability of mistaking one state as other states, can be generalized to an upper bound on PGM's error probability of mistaking one subset of states to its complement subset (See Appendix C). This almost gives us the required error bound for BSD, except that the PGM result is for the average case, where σ is drawn from some probability distribution, so we turned it into a worst case result with the minimax argument of [17].

1.3.0.4 Back to Learning Quantum Channels

With BSD solved, we can get an algorithm that recursively exclude a constant fraction of the concept class. In each recursion, the algorithm partition the remain concepts into three sets, $S_{yes}, S_{unknown}$, and S_{no}. Ideally, S_{yes} and S_{no} both occupy a constant fraction of the remaining concepts and have minimum distance $\gamma = \Omega(1/\operatorname{poly}\log|C|)$. Noticing that the fidelity between tensor products of outputs decays exponentially with number of samples by lemma 4, the BSD between $O(\log|C|/\gamma)$ samples of S_{yes} or S_{no} can be solved with high probability. If the target concept is in S_{yes}, the BSD solver will return *yes* with high probability, and if the target concept is in S_{no}, the BSD solver will return *no*. If the target concept is from $S_{unknown}$, the BSD solver might return anything, but what we can be sure is that, if the BSD solver returned *yes*, the target concept is not from S_{no}, and if the BSD solver return *no*, the target concept is not from S_{yes}. Therefore, we can always exclude either S_{yes} or S_{no} as possible target concept.

There is another complication in that the distance between the concepts depends on the unknown distribution D and thus cannot be calculated. In stead, we use the *empirical distance* between concepts, $\Delta_{emp}(c_1, c_2) = \frac{1}{T}\sum_{i=1}^{T}\left[\Delta_{tr}\left(c_1(x_i), c_2(x_i)\right)\right]$, where $\{x_i\}$ are the inputs points we drawn in each recursion. Our calculation shows that that the error incurred from this change of distance measure is negligible.

1.3.0.5 Partition Sub-algorithm

It is not always possible to have an ideal partition where S_{yes} and S_{no} are both constant-fraction sized[7] and separated by the gap γ. Therefore, we designed a classical partition sub-algorithm (Algorithm 1) to handle these exceptions.

[7] By "constant-fraction sized" we mean "occupies a constant fraction of the remaining concepts".

An example where the ideal partition is not possible is the extreme case where every concept in the concept class is literally identical to each other. Note that in this extreme case can be trivially solved by output anything in the concept class as the hypothesis because everything is ϵ-close to c^*.

Our partition sub algorithm builds on the intuition of what happened in the above extreme case. More specifically, our partition algorithm will not reserve a constant-fraction sized S_{no} if a significant fraction of C is clustered around a concept. In such case, we choose the cluster as S_{yes} with a γ-thick "shell" of $S_{unknown}$ around it. If we measured *no*, we can rule out S_{yes}, which is a constant fraction of $|C|$. If we measured yes, we can output the center of the cluster as the hypothesis, and we tune γ so that everything in either S_{yes} or $S_{unknown}$ is ϵ-close to the center. This completes our algorithm for mixed state outputs.

1.4 Lower Bounds and Agnostic Model

We complement our positive results on the sample complexity of PAC learning quantum channels with two simple lower bounds. First, by adapting a lower bound argument in [15], we prove that $\tilde{\Omega}((\log|C|)/\epsilon^2)$ samples are necessary to PAC learn quantum channels when the outputs are pure states, showing that our positive result is tight in the dependency on $|C|$ and ϵ. In particular, for the dependency on ϵ, this is in contrast with the classical results on the sample complexity for PAC learning concepts with Boolean outputs, where a tight $\Theta((\log|C|)/\epsilon)$ sample complexity is known [22, 16].[8]

Agnostic model is a learning model closely related to the PAC model, and the two models have similar sample complexity [18, 21]. In the agnostic model, the samples comes from a concept c_s that is not necessarily inside the concept class C. Accordingly, the goal of the learner is to find, with ϵ-distance error, the target concept $c^* \in C$ that is closest to c_s. We introduce the agnostic model for learning quantum channels, see section B.2 for details. Interestingly, in stark contrast to our algorithms that have dimension-independent sample complexity for learning quantum channels in PAC model, we found an $\Omega(\sqrt{d})$ lower bound on the sample complexity for learning quantum channels in agnostic model with output dimension d. Thus, in the agnostic model, learning quantum channels requires number of samples polynomial in the dimension, so it is not possible to efficiently learn quantum channels with large output dimension. Also, our negative example is in fact classical in nature, consisting of two concepts that output classical distributions, so learning classical distributions efficiently in agnostic model in large dimension is also impossible. However, since quantum pure states are not generalizations of classical distributions, the possibility of sample efficiently learn quantum channels with *pure state output* in agnostic is still open.

2 Preliminary

Throughout this paper, log is base 2 and ln is base e.

We use $\|\cdot\|_1$ to denote the trace norm $\|A\|_1 = \operatorname{tr}\sqrt{A^\dagger A}$. We use $\|\cdot\|_2$ or $\|\cdot\|_F$ to denote the Frobenius norm $\|A\|_2 = \sqrt{\operatorname{tr}(A^\dagger A)}$.

Denote the trace distance and fidelity between two distribution D_1, D_2 as $\Delta_{tr}(D_1, D_2)$ and $F(D_1, D_2)$, where the trace distance is equal to the total variation distance. Denote the trace distance and fidelity between two quantum states ρ_1, ρ_2 as $\Delta_{tr}(\rho_1, \rho_2) = \frac{1}{2}\|\rho_1 - \rho_2\|_1$

[8] The classical results show that the sample complexity is characterized by the VC dimension of the concept class C. In the case that C is finite, $\log|C|$ is a trivial upper bound on the VC dimension.

and $F(\rho_1, \rho_2) = \left\| \sqrt{\rho_1}\sqrt{\rho_2} \right\|_1$. For a quantum state σ and a quantum measurement M, denote $M(\sigma)$ as the output probability distribution when applying M on σ.

Note that fidelity and trace distance are related by

$$1 - F \leq \Delta_{tr} \leq \sqrt{1 - F^2}.$$

For two quantum channel concepts c_1, c_2, define the distance between them with respect to D as

$$\Delta(c_1, c_2) = \mathbb{E}_{x \in D}\left[\Delta_{tr}(c_1(x), c_2(x))\right].$$

We say that c_1, c_2 are ϵ-close if $\Delta(c_1, c_2) \leq \epsilon$ and ϵ-far if $\Delta(c_1, c_2) \geq \epsilon$. For two sets of concepts S_1 and S_2, define the distance between them as $\Delta\left(S_1, S_2\right) = \min\left\{\Delta(c_1, c_2) | c_1 \in S_1, c_2 \in S_2\right\}$.

2.1 Chernoff Bound

We use the following standard multiplicative version of Chernoff bound.

▶ **Theorem 2.** *Let $X_1, \ldots, X_T \in [0,1]$ be independent random variables with $\mathbb{E}[X_i] = \mu_i$. Let $X = (1/T)\sum_i X_i$, $\mu = (1/T)\sum_i \mu_i$ and $\alpha \in (0,1)$. We have*

$$\Pr[|X - \mu| \geq \alpha\mu] \leq 2^{-\Omega(\alpha^2 T \mu)}.$$

2.2 Pretty Good Measurement

The pretty good measurement (PGM) is defined as follows:

▶ **Definition 3** (pretty good measurement). *Let $\{\sigma_i\}$ be a set of density matrices and $\{p_i\}$ a probability distribution over $\{\sigma_i\}$. Define*

$$A_i = p_i \sigma_i,\ A = \sum_i A_i. \tag{1}$$

The PGM associated with $\{\sigma_i\}, \{p_i\}$ is the measurement $\{E_i\}$ with

$$E_i = A^{-1/2} A_i A^{-1/2}. \tag{2}$$

3 Problem Definitions

In this section we describe the PAC model of learning quantum channel and approximate state discrimination.

3.1 Classical PAC Learning Model

We start with a review of the classical PAC learning model.

In the classical probably approximately correct (PAC) learning model, a learner tries to learn a *target concept* $c^* \in C$ from a known *concept class* C, which is a set of Boolean functions $c : \{0,1\}^n \to \{0,1\}$, with respect to an *unknown* distribution D over the input domain $\{0,1\}^n$. Specifically, the learner is given access to a sample oracle $\mathcal{O}_{c^*,D}$, which generates i.i.d. samples $(x_i, c^*(x_i))$, where each $x_i \leftarrow D$ is drawn according to the distribution

D, and outputs a *hypothesis* $h \in C$.[9] The distance between two concepts c and h under the distribution D is defined as $\Delta_D(c,h) = \mathbb{E}_{x\sim D}\,|c(x) - h(x)|$. The goal of the learner is to find a hypothesis h with sufficiently small distance $\Delta_D(c^*,h)$ to c^*.

A learning algorithm A is a *proper* (ϵ,δ)*-PAC learner* for a concept class C if the following holds: For every $c^* \in C$ and distribution D, given oracle access to $\mathcal{O}_{c^*,D}$, $A^{\mathcal{O}_{c^*,D}}$ outputs an $h \in C$ such that $\Delta_D(c^*,h) \leq \epsilon$ with probability at least $1-\delta$. The sample complexity of A is the maximum number of samples T that A needs to query $\mathcal{O}_{c^*,D}$ to output h. The *proper* (ϵ,δ)*-PAC sample complexity* of a concept class C is the minimum sample complexity over all learners. A $\tilde{\Theta}((\log|C|)/\epsilon)$ sample complexity is known [22, 16].[10]

3.2 Learning Quantum Channels in PAC model

We now generalize classical PAC learning to the context of learning quantum channels. As above, we consider a learner trying to learn a target concept $c^* \in C$ from a known concept class C with respect to an unknown distribution D. Here, we consider the concept class C as a finite set of known d_1 to d_2 dimensional quantum channels, and D as a distribution over the Hilbert space of dimension d_1. Precisely, the learner is given access to a sample oracle $\mathcal{O}_{c^*,D}$ and outputs a hypothesis $h \in C$. The oracle $\mathcal{O}_{c^*,D}$ generates i.i.d. samples $(x_i, c^*(x_i))$, where each $x_i \leftarrow D$ is the classical description of a state drawn according to the distribution D, and $c^*(x_i)$ is the (potentially mixed) quantum state outputted by c^* on input x_i.

The distance between two concepts c and h under the distribution D is the expected trace distance $\Delta(c,h) = \mathbb{E}_{x\in D}\left[\Delta_{tr}(c(x),h(x))\right]$. The goal of the learner is to find a hypothesis $h \in C$ with sufficiently small $\Delta(c^*,h)$.

A quantum learning algorithm A is a *proper* (ϵ,δ)*-PAC learner* for C if the following holds: For every $c^* \in C$ and distribution D, given oracle access to $\mathcal{O}_{c^*,D}$, $A^{\mathcal{O}_{c^*,D}}$ outputs an $h \in C$ such that $\Delta_D(c^*,h) \leq \epsilon$ with probability at least $1-\delta$. The sample complexity of A is the maximum number of samples T that A needs to query $\mathcal{O}_{c^*,D}$ to output h. The *proper* (ϵ,δ)*-PAC sample complexity* of a concept class C is the minimum sample complexity over all learners.

3.3 Approximate State Discrimination

Let S be a finite set of d-dimensional density matrices. We want to learn a target state $\sigma \in S$ using as few identical copies of σ as possible. A quantum algorithm is an (ϵ,δ)-approximate discriminator of S if it takes the description of S and T copies of σ as input and with probability $1-\delta$ outputs a state $\rho \in S$ with $\Delta_{tr}(\rho,\sigma) \leq \epsilon$, for any $\sigma \in S$.

Note that approximate state discrimination can be viewed as a special case of PAC learning quantum channels with constant output, so the algorithms for PAC learning quantum channels in Section 4 and Section 5 trivially works for approximate state discrimination.

4 PAC Learning Quantum Channels with Pure State Output

See Appendix A

[9] The requirement that the hypothesis h is in the concept class C is referred to as proper learning. We focus on proper learning since our algorithms satisfy this property.

[10] We use $\tilde{\Theta}$ to denote Θ with log factors. The classical results show that the sample complexity is $\Theta\left((d + \log 1/\delta)/\epsilon\right)$, where d is the VC dimension of the concept class. In the case where $|C|$ is finite, $\log|C|$ is a trivial upper bound on d, and there are concept classes whose VC dimension d matches $\log|C|$.

5 PAC Learning Quantum Channels with Mixed State Output

The random orthonormal measurement approach in Section 4 does not work since two high dimensional mixed states with constant trace distance between them can have negligible Frobenius distance between them. Instead, We follow the intuitions detailed in Section 1.3. We define the bichromatic state discrimination problem (BSD), solve BSD with PGM techniques , and build our learner algorithm with the BSD solver and a partition sub-algorithm.

Before we show the algorithms for bicromatic state discrimination, let us first show that we can efficiently amplify the distance between concepts by taking samples.

▶ **Lemma 4** (concept distance amplification). *Let c be a quantum channel concept ϵ-far from the target concept c^*. Let $\{x_1, x_2, \ldots, x_T\}$ be T inputs drawn from the distribution D. With probability $1 - 2^{-\Omega(T\epsilon)}$ over $\{x_i\}$ drawn, we have*

$$F\left(\bigotimes_{i\in[T]} c(x_i), \bigotimes_{i\in[T]} c^*(x_i)\right) \leq 2^{-\Omega(T\epsilon^2)} \tag{3}$$

and

$$\Delta_{tr}\left(\bigotimes_{i\in[T]} c(x_i), \bigotimes_{i\in[T]} c^*(x_i)\right) \geq 1 - 2^{-\Omega(T\epsilon^2)}. \tag{4}$$

Proof. By Chernoff bound, with probability $1 - 2^{-\Omega(T\epsilon)}$,

$$\sum_i \Delta_{tr}\left(c(x_i), c^*(x_i)\right) \geq \frac{1}{2}T\epsilon. \tag{5}$$

Then by Cauchy-Schwarz Inequality,

$$\sum_i (\Delta_{tr}\left(c(x_i), c^*(x_i)\right))^2 \geq \frac{1}{4}T\epsilon^2. \tag{6}$$

Then the amplified fidelity is bounded by

$$\begin{aligned} F\left(\bigotimes_i c(x_i), \bigotimes_i c^*(x_i)\right) &= \Pi_i F\left(c(x_i), c^*(x_i)\right) \\ &\leq \Pi_i \sqrt{1 - (\Delta_{tr}\left(c(x_i), c^*(x_i)\right))^2} \\ &\leq \exp\left[-\frac{1}{2}\sum_i (\Delta_{tr}\left(c(x_i), c^*(x_i)\right))^2\right] = 2^{-\Omega(T\epsilon^2)}, \end{aligned} \tag{7}$$

where the last inequality is true because $1 - x \leq e^{-x}$. And the amplified trace distance is

$$\Delta_{tr}\left(\bigotimes_{i\in[T]} c(x_i), \bigotimes_{i\in[T]} c^*(x_i)\right) \geq 1 - F\left(\bigotimes_i c(x_i), \bigotimes_i c^*(x_i)\right) = 1 - 2^{-\Omega(T\epsilon^2)}. \tag{8}$$

◀

Lemma 4 means that we can amplify the distance between tensor products of samples from quantum channels as efficiently as we do on samples of fixed quantum states. This means that PAC learning quantum channels is really similar to approximate state discrimination even in the mixed state case.

Now back to BSD. The bichromatic state discrimination problem (BSD) is defined as follows:

▶ **Definition 5** (Bichromatic State Discrimination Problem (BSD)). *Given complete information of two sets of quantum states, S_{yes} and S_{no}, with fidelity $F(S_{yes}, S_{no}) \leq \eta$ and size $S_{yes} \leq N$, $S_{no} \leq N$, and one copy of an unknown quantum state σ, the goal is to decide whether $\sigma \in S_{yes}$ or $\sigma \in S_{no}$. We say a quantum algorithm solves (η, N)-BSD with error δ if for all S_{yes} and S_{no} such that $F(S_{yes}, S_{no}) \leq \eta$, $S_{yes} \leq N$, and $S_{no} \leq N$, given complete information about S_{yes} and S_{no} and one copy of an unknown quantum state σ as input to the algorithm, the algorithm output and yes/no answer satisfies the following two conditions:*

1. *If $\sigma \in S_{yes}$, the learner outputs yes with probability $(1-\delta)$.*
2. *If $\sigma \in S_{no}$, the learner outputs no with probability $(1-\delta)$.*

The learner can output anything if σ does not come from either of the two sets.

We show the existence of a BSD solver by first showing that PGM over $S_{yes} \cup S_{no}$ solves the "average case" BSD and then turn it into a "worst case" result by the minimax theorem.

First by slightly modifying a result of [9] and [6], We show that PGM can solve the "average case" BSD:

▶ **Lemma 6** (PGM for "average BSD"). *Let S_{yes}, S_{no} be two sets of density matrices and $\{p_i\}$ be a probability distribution over $S_{yes} \cup S_{no}$.* [11] *The PGM on $S_{yes} \cup S_{no}, \{p_i\}$ satisfies*

$$\sum_{i \in S_{yes}} \sum_{j \in S_{no}} [p_i \Pr(PGM(\sigma_i) = j) + p_j \Pr(PGM(\sigma_j) = i)] \leq \sum_{i \in S_{yes}} \sum_{j \in S_{no}} F(\sigma_i, \sigma_j). \quad (9)$$

Proof. See appendix C. ◀

We can group together the outputs of the PGM in Lemma 6 and define a binary measurement $\{E_{yes}, E_{no}\}$, where $E_{yes} = \sum_{i \in S_{yes}} E_i$, $E_{no} = \sum_{i \in S_{no}} E_i$, and $\{E_i\}$ is the PGM. By Lemma 6, the binary measurement solves "average BSD" with error probability at most $\sum_{i \in S_{yes}} \sum_{j \in S_{no}} F(\sigma_i, \sigma_j)$.[12]

Since the upper bound on error is independent of the distribution $\{p_i\}$, minimax theorem guarantees the existence of a measurement that distinguishes between S_{yes} and S_{no} for any distribution $\{p_i\}$ with error probability less than $\sum_{i \in S_{yes}} \sum_{j \in S_{no}} F(\sigma_i, \sigma_j)$[13]. In particular, if $p_i = 1$ for some $\sigma_i \in S_{yes}$, the probability of the minimax measurement mistaking σ_i as something in S_{no} is upper bounded by $\sum_{i \in S_{yes}} \sum_{j \in S_{no}} F(\sigma_i, \sigma_j)$, and vice versa. We formalize this discussion as the following Theorem.

▶ **Theorem 7** (solver for BSD, Theorem 1 restated). *There exist an algorithm that solves (η, N)-BSD with error $\delta = N^2\eta$*

Proof. Consider the zero sum game between two players where player1 choose a probability distribution $\{p_i\}$ over $S_{yes} \cup S_{no}$ and player2 choose a binary measurement strategy M. The score of player1 is given by the following error probability[14]:

$$P_{bi-error} = \sum_{i \in S_{yes}} [p_i \Pr(M(\sigma_i) = no)] + \sum_{j \in S_{no}} [p_j \Pr(M(\sigma_j) = yes)] \quad (10)$$

[11] We will slightly abuse the notation and write $i \in S_{yes}$ or $j \in S_{no}$ instead of $\sigma_i \in S_{yes}$ or $\sigma_j \in S_{no}$.

[12] A careful reader might notice that since we only want a binary answer, we are essentially distinguishing the states $A_{yes} = \sum_{i \in S_{yes}} p_i\sigma_i$ and $A_{no} = \sum_{j \in S_{no}} p_j\sigma_j$, and thus the optimal error probability is characterized by trace distance between A_{yes} and A_{no}. However, to our knowledge there is no inequality in the literature giving a *lower bound* on trace distance between on linear combinations of density matrices, so actually, the other direction of the trace-distance characterization is the relevant one: Lemma 6 gives a new lower bound on $\Delta_{tr}(A_{yes}, A_{no})$.

[13] This argument was used in [17]

[14] We will slightly abuse the notation and write $i \in S_{yes}$ or $j \in S_{no}$ instead of $\sigma_i \in S_{yes}$ or $\sigma_j \in S_{no}$.

It is easy to check that that strategies of both sides are linear, so we can apply the minimax theorem to get

$$\min_{M} \max_{\{p_i\}} P_{bi-error} = \max_{\{p_i\}} \min_{M} P_{bi-error} \leq \sum_{i \in S_{yes}} \sum_{j \in S_{no}} F(\sigma_i, \sigma_j) \leq N^2 \eta, \tag{11}$$

where the second inequality is from the promises of (η, N) BSD, and the first inequality is shown by considering the binary measurement $\{E_{yes}, E_{no}\}$, where $E_{yes} = \sum_{i \in S_{yes}} E_i$, $E_{no} = \sum_{i \in S_{no}} E_i$, and $\{E_i\}$ is the PGM of Lemma 6. This means that there is a measurement M whose error probability is less than $\sum_{i \in S_{yes}} \sum_{j \in S_{no}} F(\sigma_i, \sigma_j)$ for all probability distribution $\{p_i\}$. In particular, the error probability is at most $N^2\eta$ when player1 uses the deterministic strategy of always choosing some specific state $\sigma_i \in S_{yes} \cup S_{no}$. Therefore, algorithm of applying the measurement M solves (η, N)-BSD with error $N^2\delta$.

◀

Theorem 7 implies that if we amplify the maximum fidelity between S_{yes} and S_{no} by Lemma 4 to less than $O(1/|C|^2)$, we have a constant error probability in distinguishing whether a state is from S_{yes} or S_{no}. By lemma 4 this requires $\Theta\left(\log |C|/\gamma^2\right)$ samples if the distance between S_{yes} and S_{no} is γ.

Now we present the partition sub-algorithm. Let C_r be the set of remaining concepts that have not been cut off by the main algorithm. The sub-algorithm partitions the remaining concepts into three disjoint subsets: $(S_{yes}, S_{unknown}, S_{no})$, such that $|S_{yes}| \geq \frac{1}{9}|C_r|$[15], and $\Delta(S_{yes}, S_{no}) \geq \gamma = \Theta(\epsilon/\log|C_r|)$. The sub-algorithm might or might not found an extreme case. If no extreme case is found, $|S_{no}| \geq \frac{1}{9}|C_r|$. If an extreme case is found, more than $\frac{1}{3}|C_r|$ concepts are ϵ-close to some concept. The sub-algorithm initialized with every concept in S_{no}. It then repeatedly picks a concept c_c from S_{no} and adds concepts within the ball around c_c to S_{yes} and concepts in a γ-shell around the ball to $S_{unknown}$. The γ-shell of S_{no} ensures that $\Delta(S_{yes}, S_{no}) \geq \gamma$ and we choose the radius of the ball so that the number of concepts added to S_{yes} is greater than half the number of concepts added to $S_{unknown}$ to ensure that $|S_{yes}| > \frac{1}{2}|S_{unknown}|$ in the end. The sub-algorithm keeps adding concepts to S_{yes} and $S_{unknown}$ until $|S_{yes}| + |S_{unknown}| > \frac{1}{3}|C_r|$ or the loop is breaked by an extreme case. The sub-algorithm reports an extreme case if the number of concepts to be added to S_{yes} and $S_{unknown}$ in the current iteration is greater than $\frac{1}{3}|C_r|$. In this case we know that more than $\frac{1}{3}|C_r|$ concepts are around c_c. If no extreme case is found, since the loop stops when $|S_{yes}| + |S_{unknown}| > \frac{1}{3}|C_r|$ and the last iteration cannot add more than $\frac{1}{3}|C_r|$ concepts to S_{yes} or $S_{unknown}$, there are at least $(1 - \frac{1}{3} - \frac{1}{3})|C_r| > \frac{1}{9}|C_r|$ concepts left in S_{no}, and $|S_{yes}| > \frac{1}{3}(|S_{yes}| + |S_{unknown}|) > \frac{1}{9}|C_r|$.

There is another complication in that the distance between the concepts depends on the unknown distribution D and thus cannot be calculated. In stead, we calculate the *empirical distance* between concepts, $\Delta_{emp}(c_1, c_2) = \frac{1}{T}\sum_{i=1}^{T}\left[\Delta_{tr}\left(c_1(x_i), c_2(x_i)\right)\right]$, which depends on the input points drawn from D. We also tune ϵ into $\epsilon/2$ to accommodate for the extra error incurred.

The sub-algorithm is detailed in Algorithm 1.

▶ **Lemma 8.** *The output of Algorithm 1 satisfies the following conditions:*

$(S_{yes}, S_{unknown}, S_{no})$ is a partition of C_r. $\Delta_{emp}(S_{yes}, S_{no}) \geq \gamma = \epsilon/4\log|C_r|$. $|S_{yes}| \geq \frac{1}{9}|C_r|$. If flag_extreme $=$ false, $|S_{no}| \geq \frac{1}{9}|C_r|$. If flag_extreme $=$ true, $\Delta_{emp}(c, c_c) \leq \epsilon/2$, $\forall c \in (S_{yes} \cup S_{unknown})$.

[15] $\frac{1}{9}$ is an arbitrary constant and can be further optimized

Algorithm 1 partition sub-algorithm.

```
   Data: concepts class $C_r$, real number $\epsilon$.
   Result: Set of concepts $S_{yes}, S_{unknown}, S_{no}$, boolean variable $flag\_extreme$,
           concept $c_c$
1  $S_{no} \leftarrow C_r$, $S_{yes} \leftarrow \emptyset$, $S_{unknown} \leftarrow \emptyset$, $flag\_extreme \leftarrow false$, $\gamma \leftarrow \epsilon/(4\log|C_r|)$.
2  while $|S_{yes}| + |S_{unknown}| < \frac{1}{3}|C_r|^{16}$ do
3      $c_c \leftarrow$ a random concept in $S_{no}$;
4      Count the number of concept in $S_{no}$ whose distance to $c_c$ is in the interval
         $[(m-1)\gamma, m\gamma)$ for all $m \in [1/\gamma]$ and record the number as $b_m$. I.e.
         $b_m \leftarrow |\{c|\Delta(c, c_c) \in [(m-1)\gamma, m\gamma), c \in S_{no}\}|$;
5      Find the smallest $i^* \geq 2$ such that $b_{i^*} < 2\sum_{i\in[i^*-1]} b_i$;
6      if $\sum_{i\in[i^*-1]} b_i + b_{i^*} > \frac{1}{3}|C_r|$ then
7          $flag\_extreme \leftarrow true$;
8          move everything in $S_{yes}$ and $S_{unknown}$ back to $S_{no}$;
9          run line 12 once;
10         Terminate;
11     end
12     For the concepts in $S_{no}$, move the concepts within distance $(i^*-1)\gamma$ of $c_c$ to $S_{yes}$,
         and move the concepts whose distance to $c_c$ is in $[(i^*-1)\gamma, i^*\gamma)$ to $S_{unknown}$.
         I.e. move $\{c|\Delta(c, c_c) \in [0, (i^*-1)\gamma), c \in S_{no}\}$ to $S_{yes}$ and move
         $\{c|\Delta(c, c_c) \in [(i^*-1)\gamma, i^*\gamma), c \in S_{no}\}$ to $S_{unknown}$;
13 end
```

Proof. First note that in line 5, $\gamma = \epsilon/(4\log|C_r|)$ ensures that i^* exists and $i^* \leq \epsilon/(2\gamma)$. This can be proved by contradiction: if $b_i^* \geq 2\sum_{i\in[i^*-1]} b_i, \forall i^* \leq \epsilon/(2\gamma)$, then $b_i^* > 2b_{i^*-1}, \forall i^* \leq \epsilon/(2\gamma)$. Together with $b_1 \geq 1$ because $\Delta(c_c, c_c) = 0$, we have $b_{\lfloor\epsilon/2\gamma\rfloor} \geq 2 \cdot 2^{\log|C_r|} b_1 \geq |C_r|$, a contradiction.

$(S_{yes}, S_{unknown}, S_{no})$ is a partition because it is initialized as a partition and we only moves elements between them. Note that whenever we move something to S_{yes}, we move a γ-thick shell around it to $S_{unknown}$. By triangle inequality of empirical distances between concepts, $\Delta_{emp}(S_{yes}, S_{no}) \geq \gamma = \epsilon/4\log|C_r|$ at the end of every step.

If no extreme case is found, at each iteration of the loop at line 12, $(\sum_{i\in[i^*-1]} b_i)$ concepts are moved to S_{yes} from S_{no}, and b_{i^*} concepts are moved to $S_{unknown}$ from S_{no}. Before the last iteration of the loop $|S_{yes}| + |S_{unknown}| \leq \frac{1}{3}|C_r|$, and the number of concepts moved to S_{yes} and $S_{unknown}$ in the last iteration is $\sum_{i\in[i^*-1]} b_i + b_i^* \leq \frac{1}{3}|C_r|$, so $|S_{no}| \geq (1 - \frac{1}{3} - \frac{1}{3})|C_r| > \frac{1}{9}|C_r|$. Because of the requirement $b_{i^*} < 2\sum_{i\in[i^*-1]} b_i$ in line 5, $\sum_{i\in[i^*-1]} b_i > \frac{1}{3}(\sum_{i\in[i^*-1]} b_i + b_{i^*})$ and thus $|S_{yes}| > \frac{1}{3}(|S_{yes}| + |S_{unknown}|)$ at the end of every loop. Combined with the loop-termination condition $|S_{yes}| + |S_{unknown}| > \frac{1}{3}|C_r|$, we have $|S_{yes}| > \frac{1}{9}|C_r|$.

If an extreme case is found at line 7, because we moved everything back to S_{no}, all concepts in S_{yes} or $S_{unknown}$ are added in that one call of line 12, and thus they are all $(i^*\gamma)$-close to c_c . Recall that $i^*\gamma \leq \epsilon/2$, so everything in S_{yes} or $S_{unknown}$ is $\epsilon/2$-close to c_c. The analysis on $|S_{yes}|$ is a bit subtle. Similar to the previous paragraph, we have $\sum_{i\in[i^*-1]} b_i > \frac{1}{3}(\sum_{i\in[i^*-1]} b_i + b_{i^*})$. Combined with $\sum_{i\in[i^*-1]} b_i + b_{i^*} > \frac{1}{3}|C_r|$ to trigger line 7, we have $\sum_{i\in[i^*-1]} b_i > \frac{1}{9}|C_r|$. Since the wiping of S_{yes} and $S_{unknown}$ at the beginning of line 7 only opens more possible concepts to be added to S_{yes}, we have $|S_{yes}| \geq \sum_{i\in[i^*-1]} b_i > \frac{1}{9}|C_r|$. ◀

With the partition sub-algorithm described, we detail the main algorithm for mixed state case in Algorithm 2.

Algorithm 2 algorithm for mixed state case.

```
   Data: Concept class $C$, Sampling Oracle $\mathcal{O}_{c^*,D}$
   Result: hypothesis $h$
 1 $C_r \leftarrow C$ ;
 2 $T \leftarrow \Theta\left(\frac{\log^2|C|(\log|C|+\log(1/\delta))}{\epsilon^2}\right)$ ;
 3 while do
 4     Call $\mathcal{O}_{c^*,D}$ $T$ times, getting $T$ samples
         $\{(x_1, c^*(x_1)), (x_2, c^*(x_2)), \ldots (x_T, c^*(x_T))\}$;
 5     $(S_{yes}, S_{unknown}, S_{no}, flag\_extreme, c_c) \leftarrow$ (Algorithm 1)$(C_r, \epsilon)$;
 6     Construct the measurement $M$ in Theorem 7 between $S_{yes}$ and $S_{no}$ with the
         state $\sigma_i$ corresponding to concept $c_i$ being $\sigma_i = \bigotimes_{j\in[T]} c_i(x_j)$;
 7     $Measure\_result \leftarrow M(\bigotimes_{j\in[T]} c^*(x_j))$.;
 8     if $Measure\_result = no$ then
 9         remove $S_{yes}$ from $C_r$;
10     end
11     if $Measure\_result = yes$ and $flag\_extreme = false$ then
12         remove $S_{no}$ from $C_r$.;
13     end
14     if $Measure\_result = yes$ and $flag\_extreme = true$ then
15         $h \leftarrow c_c$;
16         Terminate;
17     end
18 end
```

Now we state and prove our result for mixed state case:

▶ **Theorem 9.** *Algorithm 2 is a proper (ϵ, δ)-PAC learner for any quantum circuit concept class C, using*

$$O\left(\frac{\log^3|C|(\log|C| + \log(1/\delta))}{\epsilon^2}\right)$$

samples.

Proof. By Lemma 8, Algorithm 2 removes at least $\frac{1}{9}|C_r|$ concepts from C_r in each loop unless it terminates, so it terminates in $O(\log|C|)$ loops at line 16. Combined with the fact that Algorithm 2 takes $O\left(\frac{\log^2|C|(\log|C|+\log(1/\delta))}{\epsilon^2}\right)$ samples each loop, its sample complexity is

$$O\left(\frac{\log^3|C|(\log|C| + \log(1/\delta))}{\epsilon^2}\right)$$

.

As for the correctness of the algorithm, first note that by Lemma 8 the empirical distance between any pair of concepts in S_{yes} and S_{no} is at least $\gamma_0 = \epsilon/(4\log|C|)$.

Consider any pair of concepts $c_i \in S_{yes}$ and $C_j \in S_{no}$, with the corresponding states σ_i and σ_j. By definition of empirical distance,

$$\sum_{k=1}^{T} \Delta_{tr}\left(c_i(x_k), c_j(x_k)\right) \geq T\gamma_0 \tag{12}$$

Then by Cauchy-Schwarz Inequality,

$$\sum_{k=1}^{T} \Delta_{tr}\left(c_i(x_k), c_j(x_k)\right)^2 \geq T\gamma_0^2. \tag{13}$$

Then the fidelity between σ_i and σ_j is bounded by

$$\begin{aligned} F(\sigma_i, \sigma_j) &= F\left(\bigotimes_k c_i(x_k), \bigotimes_k c_j(x_k)\right) \\ &= \Pi_k F\left(c_i(x_k), c_j(x_k)\right) \\ &\leq \Pi_k \sqrt{1 - \left(\Delta_{tr}\left(c_i(x_k), c_j(x_k)\right)\right)^2} \\ &\leq \exp\left[-\frac{1}{2}\sum_k \left(\Delta_{tr}\left(c_i(x_k), c_j(x_k)\right)\right)^2\right] \\ &= 2^{-\Omega\left(T\gamma_0^2\right)} \end{aligned} \tag{14}$$

where the last inequality is true because $1 - x \leq e^{-x}$.

There are only two possible ways for Algorithm 2 to make an error: first is to remove c^* from C_r in line 9 or line 12, and second is to output a far-away concept at line 15 because of the mismatch between empirical distance and true distance.

For the first error, note that c^* always has empirical distance zero to it self, no matter what $\{x_1, x_2, \ldots, x_T\}$ are sampled. By Theorem 7 and Equation 14 the error probability in each loop is bounded by

$$P_{error,1} \leq |C_r|^2 \cdot 2^{-\Omega(T\gamma_0^2)}. \tag{15}$$

Apply union bound over $O(\log |C|)$ loop we can bound the total error probability by

$$P_{total\,error,1} \leq \log |C||C|^2 \cdot 2^{-\Omega(T\gamma^2)} \leq O\left(\frac{\delta\,|C|^2 \log |C|}{\mathrm{poly}(|C|)}\right) \leq O(\delta) \tag{16}$$

For the second error, consider a pair of concepts that has distance bigger than ϵ. By Chernoff bound, the probability that their empirical distance is less than $\frac{1}{2}\epsilon$ is less than $2^{-\Omega(T\epsilon^2)}$. Union bound over all $O(|C|^2)$ pairs of concepts, we have

$$P_{total\,error,2} \leq |C|^2 \cdot 2^{-\Omega(T\epsilon^2)} \ll P_{total\,error,1}. \tag{17}$$

◀

References

1 Dorit Aharonov, Jordan Cotler, and Xiao-Liang Qi. Quantum algorithmic measurement. *arXiv preprint arXiv:2101.04634*, 2021.

2 Noga Alon, Shai Ben-David, Nicolo Cesa-Bianchi, and David Haussler. Scale-sensitive dimensions, uniform convergence, and learnability. *Journal of the ACM (JACM)*, 44(4):615–631, 1997.

3 Srinivasan Arunachalam and Ronald de Wolf. Guest column: A survey of quantum learning theory. *SIGACT News*, 48(2):41–67, 2017. doi:10.1145/3106700.3106710.

4 Srinivasan Arunachalam and Ronald de Wolf. Optimal quantum sample complexity of learning algorithms. In *Computational Complexity Conference*, volume 79 of *LIPIcs*, pages 25:1–25:31. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2017.

5 Srinivasan Arunachalam, Alex B Grilo, and Aarthi Sundaram. Quantum hardness of learning shallow classical circuits. *arXiv preprint arXiv:1903.02840*, 2019.

6 Koenraad MR Audenaert and Milán Mosonyi. Upper bounds on the error probabilities and asymptotic error exponents in quantum multiple state discrimination. *Journal of Mathematical Physics*, 55(10):102201, 2014.

7 Costin Bădescu and Ryan O'Donnell. Improved quantum data analysis. *arXiv preprint arXiv:2011.10908*, 2020.

8 Joonwoo Bae and Leong-Chuan Kwek. Quantum state discrimination and its applications. J. Phys. A: Math. Theor. 48 083001 (2015), 2017. doi:10.1088/1751-8113/48/8/083001.

9 Howard Barnum and Emanuel Knill. Reversing quantum dynamics with near-optimal quantum and classical fidelity. *Journal of Mathematical Physics*, 43(5):2097–2106, 2002.

10 Peter L Bartlett, Philip M Long, and Robert C Williamson. Fat-shattering and the learnability of real-valued functions. *journal of computer and system sciences*, 52(3):434–452, 1996.

11 Shai Bendavid, Nicolo Cesabianchi, David Haussler, and Philip M Long. Characterizations of learnability for classes of $\{0, ..., n\}$-valued functions. *Journal of Computer and System Sciences*, 50(1):74–86, 1995.

12 Anselm Blumer, Andrzej Ehrenfeucht, David Haussler, and Manfred K Warmuth. Occam's razor. *Information processing letters*, 24(6):377–380, 1987.

13 Anselm Blumer, Andrzej Ehrenfeucht, David Haussler, and Manfred K Warmuth. Learnability and the vapnik-chervonenkis dimension. *Journal of the ACM (JACM)*, 36(4):929–965, 1989.

14 Hao-Chung Cheng, Min-Hsiu Hsieh, and Ping-Cheng Yeh. The learnability of unknown quantum measurements. QIC, Vol. 16, No. 7-8, 0615-0656 (2016), 2015. arXiv:arXiv:1501.00559.

15 Jeongwan Haah, Aram W. Harrow, Zhengfeng Ji, Xiaodi Wu, and Nengkun Yu. Sample-optimal tomography of quantum states. In *Proceedings of the Forty-eighth Annual ACM Symposium on Theory of Computing*, STOC '16, pages 913–925, New York, NY, USA, 2016. ACM. doi:10.1145/2897518.2897585.

16 Steve Hanneke. The optimal sample complexity of pac learning. *The Journal of Machine Learning Research*, 17(1):1319–1333, 2016.

17 Aram W Harrow and Andreas Winter. How many copies are needed for state discrimination? *IEEE Transactions on Information Theory*, 58(1):1–2, 2012.

18 David Haussler. Decision theoretic generalizations of the PAC model for neural net and other learning applications. *Inf. Comput.*, 100(1):78–150, 1992. doi:10.1016/0890-5401(92)90010-D.

19 Hsin-Yuan Huang, Richard Kueng, and John Preskill. Information-theoretic bounds on quantum advantage in machine learning. *arXiv preprint arXiv:2101.02464*, 2021.

20 Michael J Kearns and Robert E Schapire. Efficient distribution-free learning of probabilistic concepts. *Journal of Computer and System Sciences*, 48(3):464–497, 1994.

21 Michael J. Kearns, Robert E. Schapire, and Linda Sellie. Toward efficient agnostic learning. *Machine Learning*, 17(2-3):115–141, 1994. doi:10.1007/BF00993468.

22 Michael J. Kearns and Umesh V. Vazirani. *An Introduction to Computational Learning Theory*. MIT Press, Cambridge, MA, USA, 1994.

23 Masoud Mohseni, AT Rezakhani, and DA Lidar. Quantum-process tomography: Resource analysis of different strategies. *Physical Review A*, 77(3):032322, 2008.

24 Ashley Montanaro. On the distinguishability of random quantum states. Comm. Math. Phys. 273(3), pp. 619-636, 2007, 2006. doi:10.1007/s00220-007-0221-7.

25 Ashley Montanaro. A lower bound on the probability of error in quantum state discrimination. In *Information Theory Workshop, 2008. ITW'08. IEEE*, pages 378–380. IEEE, 2008.

26 Balas K Natarajan. On learning sets and functions. *Machine Learning*, 4(1):67–97, 1989.

27 Ryan O'Donnell and John Wright. Efficient quantum tomography. In *Proceedings of the Forty-eighth Annual ACM Symposium on Theory of Computing*, STOC '16, pages 899–912, New York, NY, USA, 2016. ACM. doi:10.1145/2897518.2897544.

28 Pranab Sen. Random measurement bases, quantum state distinction and applications to the hidden subgroup problem. In *Computational Complexity, 2006. CCC 2006. Twenty-First Annual IEEE Conference on*, pages 14–pp. IEEE, 2005.

29 J. Prabhu Tej, Syed Raunaq Ahmed, A. R. Usha Devi, and A. K. Rajagopal. Quantum hypothesis testing and state discrimination. arXiv:1803.04944, 2018.

30 Leslie G. Valiant. A theory of the learnable. *Commun. ACM*, 27(11):1134–1142, 1984. doi:10.1145/1968.1972.

A PAC Learning Quantum Channels with Pure State Output

The algorithm follows ideas by Sen [28], who shows that random orthonormal measurement preserves trace distance between pure states. One can then apply random orthonormal measurements on each sampled output and take enough samples to amplify the distance between ϵ-far concepts to $1 - O(1/|C|)$ and show that the probability for the maximum likelihood estimate to select a ϵ-far concept over the target concept is less than $O(1/|C|)$. Take a union bound and we have a bounded error probability.

▶ **Theorem 10.** *Algorithm 3 is a proper (ϵ, δ)-PAC learner for any concept class C of quantum channels with pure state outputs, using*

$$O\left(\frac{(\log |C|) + \log(1/\delta)}{\epsilon^2}\right)$$

samples.

Algorithm 3 algorithm for pure state output.

```
1 Take $T = \Theta((\log |C| + \log(1/\delta))/\epsilon^2)$ samples $(x_1, \sigma_1), (x_2, \sigma_2), \ldots, (x_T, \sigma_T)$ ;
2 Do a random orthonormal measurement[a] $M_i$ on each output state $\sigma_i$. Let the
    measured outputs be $\{z_i\}$ ;
3 Output the concept $h \in C$ that is most likely to give the measured result of line 2:

      $h = \arg\max_{c \in C} \Pi_{i\in[T]} \Pr[M_i(c(x_i)) = z_i]$
```

[a] The measurement has d_2 outcomes, where d_2 is the dimension of output quantum state.

We need the following theorem to prove the correctness of Algorithm 3. First we state the result 1 of [28] (lemma 4 of arxiv version):

▶ **Theorem 11** (random orthonormal measurement [28])**.** *Let σ_1, σ_2 be two density matrices in $\mathbb{C}^d$. Define $r := \mathrm{rank}(\sigma_1 - \sigma_2)$. There exists a universal constant $k > 0$ such that if $r < k\sqrt{d}$ then with probability at least $1 - \exp(-kd/r)$ over the choice of a random orthonormal measurement basis M in $\mathbb{C}^d$, $\|M(\sigma_1) - M(\sigma_2)\|_1 > k \|\sigma_1 - \sigma_2\|_F$.* [17]

Note that if σ_1, σ_2 are pure states, $r < 2 < k\sqrt{n}$ for large enough n and $\|\sigma_1 - \sigma_2\|_1 \leq \sqrt{2} \|\sigma_1 - \sigma_2\|_F$ so that $\Delta_{tr}(M(\sigma_1), M(\sigma_2)) > k/\sqrt{2}\Delta_{tr}(\sigma_1, \sigma_2)$.

The following lemma shows how trace distance of the measured result grows when we take multiple samples.

[17] Recall that $M(\sigma)$ is the output distribution of the measurement M on state σ.

▶ **Lemma 12** (trace distance amplification). *Let $X_1, X_2, \dots, X_T$ be T independent distributions and so are $Y_1, Y_2, \dots, Y_T$. Denote the joint distribution $(X_1, X_2, \dots, X_T)$ as X and $(Y_1, Y_2, \dots, Y_T)$ as Y. Suppose that*

$$\sum_i \Delta_{tr}(X_i, Y_i) = T\epsilon, \tag{18}$$

then

$$\Delta_{tr}(X, Y) \geq 1 - 2^{-\Omega(T\epsilon^2)} \tag{19}$$

Proof. By Cauchy-Schwarz inequality,

$$\sum_i (\Delta_{tr}(X_i, Y_i))^2 \geq T\epsilon^2, \tag{20}$$

Then the joint fidelity is bounded by

$$\begin{aligned} F(X, Y) &= \Pi_i F(X_i, Y_i) \\ &\leq \Pi_i \sqrt{1 - (\Delta_{tr}(X_i, Y_i))^2} \\ &\leq \exp\left[-\frac{1}{2}\sum_i (\Delta_{tr}(X_i, Y_i))^2\right] = 2^{-\Omega(T\epsilon^2)}, \end{aligned} \tag{21}$$

where the last inequality is true because $1 - x \leq e^{-x}$. And the joint trace distance is

$$\Delta_{tr}(X, Y) \geq 1 - F(X, Y) = 1 - 2^{-\Omega(T\epsilon^2)}. \tag{22}$$

◀

The following lemma analyzes the effectiveness of maximum likelihood estimate.

▶ **Lemma 13.** *For any two distributions D, D^* have total variation distance α, $\Pr_{i\sim D^*}(D(i) \leq D^*(i)) \geq \alpha$*

Proof.

$$\begin{aligned} 0 &\leq \sum_{i:D(i)\leq D^*(i)} D(i) \\ &= \sum_{i:D(i)\leq D^*(i)} D(i) - D^*(i) + \sum_{i:D(i)\leq D^*(i)} D^*(i) \\ &= \frac{1}{2}\left[\sum_{i:D(i)\leq D^*(i)} (D(i) - D^*(i)) + \sum_{i:D^*(i)\leq D(i)} (D^*(i) - D(i))\right] + \sum_{i:D(i)\leq D^*(i)} D^*(i) \\ &= -\alpha + \Pr_{i\sim D^*}(D(i) \leq D^*(i)) \\ \Rightarrow &\Pr_{i\sim D^*}(D(i) \leq D^*(i)) \geq \alpha \end{aligned} \tag{23}$$

The third line is true because $\sum_{i:D(i)\leq D^*(i)}(D(i) - D^*(i)) = \sum_{i:D^*(i)\leq D(i)}(D^*(i) - D(i))$. ◀

We think D^* as the correct distribution and D is a distribution far away, with the total variation distance between them being $\alpha = 1 - \epsilon$. When we use maximum likelihood estimation to distinguish D^* from D, Lemma 13 says that the probability of error is less than ϵ. Now we are ready to prove theorem 10.

Proof. Let c^* be the target concept, and c a concept such that $\Delta(c^*, c) > \epsilon$. Recall that we took

$$T = \Theta\left(\frac{\log|C| + \log(1/\delta)}{\epsilon^2}\right)$$

samples. For all $i \in [T]$, apply Theorem 11 to the pair of states $(c^*(x_i), c(x_i))$, we get that with probability $1 - \exp(-kd_2/2)$ over random orthonormal measurements M_i,

$$\Delta_{tr}\left(M_i(c^*(x_i)), M_i(c(x_i))\right) > k/\sqrt{2}\Delta_{tr}(c^*(x_i), c(x_i)), \tag{24}$$

where k is a universal constant. Since you can pad some ancilla states to increase d_2 without changing trace distances if $\exp(-kd_2/2)$ is not small enough, we ignore this term. By Chernoff bound, with probability at least $1 - 2^{-\Omega(T\epsilon)}$ over $\{x_i\}$ sampled from D,

$$(1/T) \cdot \sum_i \Delta_{tr}\left(M_i(c^*(x_i)), M_i(c(x_i))\right) > (1/T) \cdot \sum_i k/\sqrt{2}\Delta_{tr}(c^*(x_i), c(x_i)) \geq \frac{k}{2\sqrt{2}}\epsilon. \tag{25}$$

So we can apply Lemma 12 to get that with probability at least $1 - 2^{-\Omega(T\epsilon)}$,

$$[\Delta_{tr}\left(\{M_i(c^*(x_i))\}, \{M_i(h(x_i))\}\right)] \geq 1 - 2^{-\Omega(T\epsilon^2)}. \tag{26}$$

Now, note that by Lemma 13, the probability that the maximal likelihood estimation (incorrectly) selects c is at most $(2^{-\Omega(T\epsilon^2)} + 2^{-\Omega(T\epsilon)})$. By taking a union bound over all such c, we get

$$\Pr[\Delta(c^*, h) > \epsilon] \leq (2^{-\Omega(T\epsilon^2)} + 2^{-\Omega(T\epsilon)}) \cdot |C| \leq \delta. \tag{27}$$

◀

B Lower Bounds

In this section we describe two simple lower bounds. One is an $\Omega((1-\delta)\ln|C|/\epsilon^2)/\ln(\ln|C|/\epsilon)$ lower bound on the sample complexity of approximate state discrimination for pure states, which in turn gives lower bounds on the sample complexity of PAC learning quantum channels. The other is an $\Omega(\sqrt{d})$ lower bound on the sample complexity of learning large dimensional *classical distribution* in the *agnostic* model, which in turn lower bounds approximate state discrimination and PAC learning quantum state in the agnostic model [18, 21].

B.1 Lower Bound for Pure State Case

▶ **Theorem 14.** *The sample complexity of (ϵ, δ)-approximate state discrimination on a set C of pure states is $\Omega((1-\delta)\ln|C|/\epsilon^2)/\ln(\ln|C|/\epsilon)$.*

Proof. This lower bound uses the ϵ-packing-net construction of [15]. In Lemma 5 of the arxiv version of [15], the authors showed the existence of a set C of d-dimensional pure states with the following three properties: the distance between each state is at least ϵ, the Holevo information χ_0 for states uniformly drawn from the set is $O(\epsilon^2\ln(d/\epsilon))$, and $\ln|C| = \Omega(d)$. With a simple reduction to communication protocol and Holevo theorem, [15] showed that to distinguish states in C with probability δ, $\frac{(1-\delta)\ln|C| - \ln 2}{\chi_0} = \Omega((1-\delta)\ln|C|/\epsilon^2)/\ln(\ln|C|/\epsilon)$ samples are required. Since every state in C is ϵ-far from each other, an (ϵ, δ)-approximate discriminator should be able to distinguish each state in C with probability δ, therefore the discriminator must take $\Omega((1-\delta)\ln|C|/\epsilon^2)/\ln(\ln|C|/\epsilon)$ samples. This matches the sample complexity of our pure state algorithm in terms of ϵ and $|C|$ with some logarithmic factors. ◀

▶ Remark 15. Unfortunately, running the same argument with the mixed state ϵ-packing nets of [15] does not give us tighter lower bound, so we don't have a matching lower bound for the mixed state case.

▶ **Corollary 16.** *The proper (ϵ,δ)-PAC sample complexity of a concept class C of pure states is $\Omega((1-\delta)\ln|C|/\epsilon^2)/\ln(\ln|C|/\epsilon)$.*

B.2 Agnostic Model

Agnostic model [18, 21] is a learning model related to the PAC model. In agnostic model, the target concept does not need to come from the concept class. We formally define the agnostic model for learning quantum channels as follows:

We consider a learner trying to learn a target concept c^* with respect to an unknown distribution D. The learner is also given a concept class C. Since the target concept might not be in the concept class C, the learner tries the output the concept c_{opt} that minimize the distance to the target concept c^*. Here, we consider the concept class C as a finite set of known d_1 to d_2 dimensional quantum channels, and D as a distribution over the Hilbert space of dimension d_1. Precisely, the learner is given access to a sample oracle $\mathcal{O}_{c^*,D}$ and outputs a hypothesis $h \in C$. The oracle $\mathcal{O}_{c^*,D}$ generates i.i.d. samples $(x_i, c^*(x_i))$, where each $x_i \leftarrow D$ is the classical description of a state drawn according to the distribution D, and $c^*(x_i)$ is the (potentially mixed) quantum state outputted by c^* on input x_i.

The distance between two concepts c and h under the distribution D is the expected trace distance to the target concept $\Delta(c,h) = \mathbb{E}_{x\in D}\left[\Delta_{tr}(c(x),h(x))\right]$. Let c_{opt} be the optimal output, $c_{opt} = \arg\min\left[\Delta(c,c^*)|c \in C\right]$. The goal of the learner is to find a hypothesis $h \in C$ with $\Delta(c^*,h) \le \Delta(c^*,c_{opt}) + \epsilon$.

A quantum learning algorithm A is a (ϵ,δ)-*agnostic learner* for C if the following holds: For every c^* and distribution D, given oracle access to $\mathcal{O}_{c^*,D}$, $A^{\mathcal{O}_{c^*,D}}$ outputs an $h \in C$ such that $\Delta(c^*,h) \le \Delta(c^*,c_{opt}) + \epsilon$ with probability at least $1-\delta$. The sample complexity of A is the maximum number of samples T that A needs to query $\mathcal{O}_{c^*,D}$ to output h. The (ϵ,δ)-*agnostic sample complexity* of a concept class C is the minimum sample complexity over all learners.

We show that there is no efficient quantum agnostic learner in the following theorem.

▶ **Theorem 17.** *For all $\epsilon < \frac{1}{10}$ and positive integer d, there exist a concept class C of dimension 0 to d whose (ϵ,δ)-agnostic sample complexity is $\Omega(\sqrt{d})$.*

Proof. We can get the $\Omega(\sqrt{d})$ lower bound with a simple concept class of that only has two concepts. Both of the concepts are constant channels that output classical distributions. Consider distributions on $d+1$ dimensions $e_0, e_1, \ldots, e_d$. The first concept C_1 has all weight on e_0. The second concept C_2 has weight uniformly distributed over $e_1, \ldots, e_d$. Now consider the following two set of distribution to be learned. $D_1 = \{D_{1,i}\}$ has weight $1/3$ on e_0 and weight $1/d$ on $2/3$ of dimensions $e_1 \ldots, e_d$. Anything in D_1 has distance $2/3$ to C_1 and distance $1/3$ to C_2, so it should be learned as C_2. $D_2 = \{D_{2,i}\}$ has weight $1/3$ on e_0 and weight $100/d$ on $2/300$ of dimensions $e_1 \ldots, e_d$. Anything in D_2 has distance $2/3$ to C_1 and distance $(1/3 + 99 * 2/300 + 1 * (1 - 2/300))/2 \sim 0.993$ to C_2, so it should be learned as C_1. However, D_1 and D_2 both looks pretty much like a uniform distribution on $e_1, \ldots, e_d$. To distinguish them we need to see a collision on $e_1, \ldots, e_d$. By a standard birthday bound, we need at least $\Omega(\sqrt{d/100})$ samples to see a collision. Therefore we need $\Omega(\sqrt{d})$ samples to learn classical distributions in agnostic model with constant error. In the regime of $|C| = \text{poly}(d)$, the lower bound means that it's impossible to find an efficient algorithm of sample complexity $O(\text{polylog}\,|C|)$. ◀

▶ Remark 18. Note that the construction of Theorem 17 is based on a classical distribution, so it means that agnostic learning of a classical distribution of many outputs efficiently is also impossible. To the knowledge of the authors, agnostic learning of classical distribution of many output has not been studied in the literature. Also note that classical distribution is not a subclass of pure quantum states, so Theorem 17 does not rule out that quantum channel with *pure* state outcomes can be efficiently agnostically learned.

▶ Remark 19. As mentioned in section 1.2.0.1, [7] studied the problem of quantum hypothesis selection, which can be viewed as a a relax version of agnostic learning, outputting a state h such that $\Delta(c^*, h) \leq 3.01\Delta(c^*, c_{opt}) + \epsilon$.

C PGM for "average BSD"

Please refer to the full version at `https://arxiv.org/abs/1810.10938`.

StoqMA Meets Distribution Testing

Yupan Liu
Shenzhen, China

Abstract

StoqMA captures the computational hardness of approximating the ground energy of local Hamiltonians that do not suffer the so-called sign problem. We provide a novel connection between StoqMA and distribution testing via reversible circuits. First, we prove that easy-witness StoqMA (viz. eStoqMA, a sub-class of StoqMA) is contained in MA. Easy witness is a generalization of a subset state such that the associated set's membership can be efficiently verifiable, and all non-zero coordinates are not necessarily uniform. This sub-class eStoqMA contains StoqMA with perfect completeness (StoqMA_1), which further signifies a simplified proof for $\mathsf{StoqMA}_1 \subseteq \mathsf{MA}$ [9, 12]. Second, by showing distinguishing reversible circuits with ancillary random bits is StoqMA-complete (as a comparison, distinguishing quantum circuits is QMA-complete [26]), we construct soundness error reduction of StoqMA. Additionally, we show that both variants of StoqMA that without any ancillary random bit and with perfect soundness are contained in NP. Our results make a step towards collapsing the hierarchy $\mathsf{MA} \subseteq \mathsf{StoqMA} \subseteq \mathsf{SBP}$ [9], in which all classes are contained in AM and collapse to NP under derandomization assumptions.

2012 ACM Subject Classification Theory of computation → Quantum complexity theory

Keywords and phrases StoqMA, distribution testing, error reduction, reversible circuits

Digital Object Identifier 10.4230/LIPIcs.TQC.2021.4

Related Version *Full Version*: https://arxiv.org/abs/2011.05733

Funding The author was supported by ISF Grant No. 1721/17 when he was affiliated with the Hebrew University of Jerusalem.

Acknowledgements The author thanks Alex B. Grilo for his contribution during the early stage of Section 4.1, and the proof of Proposition 19. The author also thanks anonymous reviewers for pointing out an error in the proof of Proposition 16 and valuable suggestions. Additionally, the author thanks Tomoyuki Morimae and Dorit Aharonov for helpful discussion. Circuit diagrams were drawn by the Quantikz package [28].

1 Introduction

This tale originates from Arthur-Merlin protocols, such as complexity classes MA and AM, introduced by Babai [5]. MA is a randomized generalization of the complexity class NP, namely the verifier could take advantage of the randomness. AM is additionally allowing two-message interaction. Surprisingly, two-message Arthur-Merlin protocols are as powerful as such protocols with a constant-message interaction, whereas it is a long-standing open problem whether $\mathsf{MA} = \mathsf{AM}$. It is evident that $\mathsf{NP} \subseteq \mathsf{MA} \subseteq \mathsf{AM}$. Moreover, under well-believed derandomization assumptions [31, 32], these classes collapse all the way to NP. Despite limited progresses on proving $\mathsf{MA} = \mathsf{AM}$, is there any intermediate class between MA and AM?

StoqMA is a natural class between MA and AM, initially introduced by Bravyi, Bessen, Terhal [9]. StoqMA captures the computational hardness of the stoquastic local Hamiltonian problems. The local Hamiltonian problem, defined by Kitaev [29], is substantially approximating the minimum eigenvalue (a.k.a. ground energy) of a sparse exponential-size matrix (a.k.a. local Hamiltonian) within inverse-polynomial accuracy. Stoquastic Hamiltonians [10] are a family of Hamiltonians that do not suffer the sign problem, namely all off-diagonal

16th Conference on the Theory of Quantum Computation, Communication and Cryptography (TQC 2021).
Editor: Min-Hsiu Hsieh; Article No. 4; pp. 4:1–4:22
Leibniz International Proceedings in Informatics
LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

entries in the Hamiltonian are non-positive. StoqMA also plays a crucial role in the Hamiltonian complexity – StoqMA-complete is a level in the complexity classification of 2-local Hamiltonian problems on qubits [17, 11], along with P, NP-complete, and QMA-complete.

Inspiring by the Monte-Carlo simulation in physics, Bravyi and Terhal [9, 12] propose a MA protocol for the stoquastic frustration-free local Hamiltonian problem, which further signifies StoqMA with perfect completeness (StoqMA_1) is contained in MA. A uniformly restricted variant[1] of this problem, which is also referred to as SetCSP [3][2], essentially captures the MA-hardness.

To characterize StoqMA through the distribution testing lens, we begin with an informal definition of StoqMA and leave the details in Section 2.2. For a language $\mathcal{L}$ in StoqMA, there exists a verifier V_x that takes $x \in \mathcal{L}$ as an input, where the verifier's computation is given by a classical reversible circuit, viewed as a quantum circuit. Besides a non-negative state[3] in the verifier's input as a witness, to utilize the randomness, ancillary qubits in the verifier's input consist of not only state $|0\rangle$ but also $|+\rangle := (|0\rangle + |1\rangle)/\sqrt{2}$. After applying the circuit, the designated output qubit is measured in the Hadamard basis[4]. A problem is in $\mathsf{StoqMA}(a, b)$ for some $a > b \geq 1/2$, if for *yes* instances, there is a witness making the verifier accept with probability at least a; whereas for *no* instances, all witness make the verifier accepts with probability at most b. The gap between a and b is at least an inverse polynomial since error reduction for StoqMA is unknown.

The optimality of *non-negative* witnesses suggests a novel connection to *distribution testing*. Let $|0\rangle|D_0\rangle + |1\rangle|D_1\rangle$ be the state before the final measurement, where $|D_k\rangle = \sum_{i\in\{0,1\}^{n-1}} \sqrt{D_k(i)}\,|i\rangle$ for $k = 0, 1$ and n is the number of qubits utilized by the verifier. A straightforward calculation indicates that the acceptance probability of a StoqMA verifier is linearly dependent on the *squared Hellinger distance* $d_H^2(D_0, D_1)$ between D_0 and D_1, which indeed connects to distribution testing! Consequently, to prove $\mathsf{StoqMA} \subseteq \mathsf{MA}$, it suffices to approximate $d_H^2(D_0, D_1)$ within an inverse-polynomial accuracy using merely polynomially many samples[5].

1.1 Main results

StoqMA with easy witness (eStoqMA). With this connection to distribution testing, it is essential to take advantage of the *efficient query access* of a non-negative witness where a witness satisfied with this condition is the so-called *easy witness*. For this sub-class of StoqMA (viz. eStoqMA) such that there exists an easy witness for any *yes* instances, we are then able to show an MA containment by utilizing both query and sample accesses to the witness. Informally, easy witness is a generalization of a subset state such that the associated state's membership is efficiently verifiable, and all non-zero coordinates are unnecessarily uniform. It is evident that a classical witness is also an easy witness, but the opposite is not necessarily true (See Remark 17). Now let us state our first main theorem:

[1] It is the projection uniform stoquastic local Hamiltonian problem, namely each local term in Hamiltonian is exactly a projection. See Definition 2.10 in [4].

[2] Namely, a modified constraint satisfaction problem such that both constraints and satisfying assignments are a subset.

[3] A witness here could be any quantum state, but the optimal witness is a non-negative state, see Remark 10.

[4] It is worthwhile to mention that we can define MA [10] (see Definition 7) in the same fashion, namely replacing the measurement on the output qubit by the computational basis.

[5] Each sample is actually the measurement outcome after running an independent copy of the verifier, see Remark 12.

▶ **Theorem 1** (Informal of Theorem 15). *$\mathsf{eStoqMA} = \mathsf{MA}$.*

It is worthwhile to mention that easy witness also relates to SBP (Small Bounded-error Probability) [7]. In particular, Goldwasser and Sipser [22] propose the celebrated Set Lower Bound protocol – it is an AM protocol for the problem of approximately counting the cardinality of such an efficient verifiable set. Recently, Watson [42] and Volkovich [40] separately point out that such a problem is essentially SBP-complete.

Although eStoqMA seems only a sub-class of StoqMA, we could provide an arguably simplified proof for $\mathsf{StoqMA}_1 \subseteq \mathsf{MA}$ [9]. Namely, employed the local verifiability of SetCSP [3], it is evident to show eStoqMA contains StoqMA with perfect completeness, which infers $\mathsf{StoqMA}_1 \subseteq \mathsf{MA}$. However, it remains open whether all StoqMA verifier has easy witness, whereas an analogous statement is false for classical witnesses (see Proposition 27).

Reversible Circuit Distinguishability is StoqMA-complete. It is well-known that distinguishing quantum circuits (a.k.a. the Non-Identity Check problem), namely given two efficient quantum circuits and decide whether there exists a pure state that distinguishes one from the other, is QMA-complete [26]. Moreover, if we restrict these circuits to be reversible (with the same number of ancillary bits), this variant is NP-complete [27]. What happens if we also allow *ancillary random bits*, viewed as quantum circuits with ancillary qubits which is initially state $|+\rangle$? It seems reasonable to believe this variant is MA-complete; however, it is actually StoqMA-complete, as stated in Theorem 2:

▶ **Theorem 2** (Informal of Theorem 22). *Distinguishing reversible circuits with ancillary random bits within an inverse-polynomial accuracy is* StoqMA*-complete.*

In fact, Theorem 2 is a consequence of the distribution testing explanation of a StoqMA verifier's maximum acceptance probability. We can view Theorem 2 as new strong evidence of $\mathsf{StoqMA} = \mathsf{MA}$. It further straightforwardly inspires a simplified proof of [27]:

▶ **Proposition 3** (Informal of Proposition 28). *Distinguishing reversible circuits without ancillary random bits is* NP*-complete.*

Apart from the role of randomness, Proposition 4 is analogous for StoqMA regarding the well-known derandomization property [21] of Arthur-Merlin systems with *perfect soundness*:

▶ **Proposition 4** (Informal of Proposition 23). StoqMA *with perfect soundness is in* NP*.*

Notably, the NP-containment in Proposition 4 holds even for $\mathsf{StoqMA}(a, b)$ verifiers with arbitrarily small gap $a - b$. It is arguably surprising since $\mathsf{StoqMA}(a, b)$ with an exponentially small gap (i.e., the precise variant) at least contains $\mathsf{NP}^{\mathsf{PP}}$ [33], but such a phenomenon does not appear in this scenario.

Soundness error reduction of StoqMA. Error reduction is a rudimentary property of many complexity classes, such as P, BPP, MA, QMA, etc. . It is peculiar that such property of StoqMA is open, even though this class has been proposed since 2006 [9]. An obstacle follows from the limitation of performing a single-qubit Hadamard basis final measurement, so we cannot directly take *the majority vote* of outcomes from the verifier's parallel repetition. Utilized the gadget in the proof of Theorem 2, we have derived soundness error reduction of StoqMA, which means we could take *the conjunction* of verifier's parallel repetition's outcomes:

▶ **Theorem 5** (Soundness error reduction of StoqMA). *For any polynomial $r = \mathrm{poly}(n)$,*

$$\mathsf{StoqMA}\left(\frac{1}{2}+\frac{a}{2}, \frac{1}{2}+\frac{b}{2}\right) \subseteq \mathsf{StoqMA}\left(\frac{1}{2}+\frac{a^r}{2}, \frac{1}{2}+\frac{b^r}{2}\right).$$

1.2 Discussion and open problems

Towards SBP = MA. As stated before, it is known $\mathsf{MA} \subseteq \mathsf{StoqMA} \subseteq \mathsf{SBP} \subseteq \mathsf{AM}$ [7, 9]. Note a subset state associated with an efficient membership-verifiable set is an easy witness. Could we utilize this connection and deduce proof of $\mathsf{SBP} \subseteq \mathsf{eStoqMA}$?

Owing to the wide uses of the Set Lower Bound protocol [22], such a solution would be a remarkable result with many complexity-theoretic applications. Unfortunately, even a QMA containment for this kind of approximate counting problem is unknown. Despite such smart usage of the Grover algorithm implies an $O(\sqrt{2^n/|S|})$-query algorithm [2, 8, 39], we are not aware of utilizing a quantum witness. Furthermore, an oracle separation between SBP and QMA [1] suggests that such a proof of $\mathsf{SBP} \subseteq \mathsf{QMA}$ is supposed to be in a non-black-box approach, which signifies a better understanding beyond a query oracle is required.

Besides SBP vs. MA, it remains open whether $\mathsf{StoqMA} = \mathsf{MA}$. It is natural to ask whether each StoqMA verifier has easy witness. However, we even do not know how to prove $\mathsf{StoqMA}(1-a, 1-1/\mathrm{poly}(n))$ has easy witness, where a is negligible (i.e., an inverse super-polynomial). In [4], they prove $\mathsf{StoqMA}(1-a, 1-1/\mathrm{poly}(n)) \subseteq \mathsf{MA}$ by applying the probabilistic method on a random walk, whereas the existence of easy witness seems to require a stronger structure[6].

Towards error reduction of StoqMA. Error reduction of StoqMA is an open problem since Bravyi, Bessen, and Terhal define this class in 2006 [9]. We first state this conjecture:

▶ **Conjecture 6** (Error reduction of StoqMA). *For any a, b such that $1/2 \leq b < a \leq 1$ and $a - b \geq 1/\mathrm{poly}(n)$, the following holds for any polynomial $l(n)$:* $\mathsf{StoqMA}(a, b) \subseteq \mathsf{StoqMA}\left(1-2^{-l(n)}, 1/2+2^{-l(n)}\right)$.

As [4] shows that StoqMA with a negligible completeness error is contained in MA, (completeness) error reduction of StoqMA plays a crucial role in proving $\mathsf{StoqMA} = \mathsf{MA}$. Instead of performing the majority vote among parallelly running verifiers, another commonplace approach is first reducing errors of completeness and soundness separately, then utilizing these two procedures alternatively with well-chosen parameters. For instance, the renowned polarization lemma of SZK [36, 6], and the space-efficient error reduction of QMA [19]. Since Theorem 5 already states soundness error reduction of StoqMA, is it possible to also construct a completeness error reduction? Namely, a mechanism that builds a new $\mathsf{StoqMA}(1/2 + a'/2, 1/2 + b'/2)$ verifier from the given $\mathsf{StoqMA}(1/2 + a/2, 1/2 + b/2)$ verifier such that a' is super-polynomially close to 1. It seems to require new ideas since a direct analog of the XOR lemma in the polarization lemma of SZK, such as Lemma 4.11 in [6], does not work here.

StoqMA with exponentially small gap. Fefferman and Lin prove [20] that PreciseQMA is as powerful as PSPACE, where PreciseQMA is a variant of $\mathsf{QMA}(a, b)$ with exponentially small gap $a - b$. Moreover, we know that both PreciseQCMA and PreciseMA are equal to $\mathsf{NP}^{\mathsf{PP}}$ [33], where PreciseQCMA is a precise variant of QMA with a classical witness of the verifier. It is evident that PreciseStoqMA is between $\mathsf{NP}^{\mathsf{PP}}$ and PSPACE, also the classical-witness variant of this class is precisely $\mathsf{NP}^{\mathsf{PP}}$ (see Section 3.3). Does PreciseStoqMA an intermediate class between $\mathsf{NP}^{\mathsf{PP}}$ and PSPACE, or even strong enough to capture the full PSPACE power?

[6] The candidate here is the set S of all good strings (see Appendix B) of the given SetCSP instance, which is unnecessary an optimal witness. It is thus unclear whether the frustration of S remains negligible.

Paper organization

Section 2 introduces useful terminologies and notations. Section 3 proves that easy-witness StoqMA is contained in MA, which indicates an arguably simplified proof of $\mathsf{StoqMA}_1 \subseteq \mathsf{MA}$, together with remarks on classical-witness StoqMA. Section 4 presents a new StoqMA-complete problem named reversible circuit distinguishability, and the complexity of this problem's exact variant, which infers StoqMA with perfect soundness is in NP. Section 5 provides error reduction of StoqMA regarding soundness error.

2 Preliminaries

2.1 Non-negative states

We assume familiarity with quantum computing on the levels of [34]. Beyond this, we then introduce some notations which are more particular for this paper: the *support* of $|\psi\rangle$, $supp(|\psi\rangle) := \{i \in \{0,1\}^n : \langle\psi|i\rangle \neq 0\}$, is the set strings with non-zero amplitude. A quantum state $|\psi\rangle$ is *non-negative* of $\langle i|\psi\rangle \geq 0$ for all $i \in \{0,1\}^n$. For any $S \subseteq \{0,1\}^n$, we refer to the state $|S\rangle := \frac{1}{\sqrt{|S|}} \sum_{i \in S} |i\rangle$ as the subset state corresponding to the set S [41].

2.2 Complexity class: MA and StoqMA

A (promise) problem $\mathcal{L} = (\mathcal{L}_{\text{yes}}, \mathcal{L}_{\text{no}})$ consists of two non-overlapping subsets $\mathcal{L}_{\text{yes}}, \mathcal{L}_{\text{no}} \subseteq \{0,1\}^*$. These classes MA and StoqMA considered in this paper using the language of reversible circuits, as Definition 7 and Definition 9.

▶ **Definition 7** (MA, adapted from [9]). *A promise problem $\mathcal{L} = (\mathcal{L}_{\text{yes}}, \mathcal{L}_{\text{no}}) \in \mathsf{MA}$ if there exists an* MA *verifier such that for any input $x \in \mathcal{L}$, an associated uniformly generated verification circuit V_x using only classical reversible gates (i.e. Toffoli, CNOT, X) on $n := n_w + n_0 + n_+$ qubits and a computational-basis measurement on the output qubit, where n_w is the number of qubits for a witness, and n_0 (or n_+) is the number of $|0\rangle$ (or $|+\rangle$) ancillary qubits, such that*

Completeness. *If $x \in \mathcal{L}_{\text{yes}}$, then there exists an n-qubit non-negative witness $|w\rangle$ such that* $\Pr[V_x \text{ accepts } |w\rangle] \geq 2/3$.

Soundness. *If $x \in \mathcal{L}_{\text{no}}$, we have $\Pr[V_x \text{ accepts } |w\rangle] \leq 1/3$ for any n-qubit witness $|w\rangle$.*

For simplicity, we denote $|\bar{0}\rangle := |0\rangle^{\otimes n_0}$ and $|\bar{+}\rangle := |+\rangle^{\otimes n_+}$ for the rest of this paper. We refer the equivalence between Definition 7 and the standard definition of MA to as Remark 8, which is first observed by [10].

▶ Remark 8 (Equivalent definitions of MA). The standard definition of MA only allows classical witnesses, viz. binary strings. To show it is equivalent to Definition 7, it suffices to prove the optimal witness for *yes* instances is classical. Notice that $\Pr[V_x \text{ accepts } |w\rangle] = \langle\psi_{\text{in}}| V_x^\dagger \Pi_{\text{out}} V_x |\psi_{\text{in}}\rangle$ where $|\psi_{\text{in}}\rangle := |w\rangle \otimes |\bar{0}\rangle \otimes |\bar{+}\rangle$ and $\Pi_{\text{out}} = |0\rangle\langle 0|_1 \otimes I_{\text{else}}$. Since $V_x^\dagger \Pi_{\text{out}} V_x$ is a diagonal matrix, the optimal witness of V_x is classical.

Analogously, we could define NP using classical reversible gates by setting $n_+ = 0$ in Definition 7. Now we proceed with the definition of StoqMA.

▶ **Definition 9** (StoqMA, adapted from [9]). *A promise problem $\mathcal{L} = (\mathcal{L}_{\text{yes}}, \mathcal{L}_{\text{no}}) \in \mathsf{StoqMA}$ if there is a* StoqMA *verifier such that for any input $x \in \mathcal{L}$, a uniformly generated verification circuit V_x using Toffoli, CNOT, X gates on $n := n_w + n_0 + n_+$ qubits and a Hadamard-basis measurement on the output qubit, where n_w is the number of qubits for a witness, and n_0*

(or n_+) is the number of $|0\rangle$ (or $|+\rangle$) ancillary qubits, such that for efficiently computable functions $a(n)$ and $b(n)$:

Completeness. *If $x \in \mathcal{L}_{\text{yes}}$, then there exists an n-qubit non-negative witness $|w\rangle$ such that* $\Pr[V_x \text{ accepts } |w\rangle] \geq a(n)$.

Soundness. *If $x \in \mathcal{L}_{\text{no}}$, we have* $\Pr[V_x \text{ accepts } |w\rangle] \leq b(n)$ *for any n-qubit witness $|w\rangle$.*

Moreover, $a(n)$ and $b(n)$ satisfy $1/2 \leq b(n) < a(n) \leq 1$ and $a(n) - b(n) \geq 1/\text{poly}(n)$.

Error reduction of StoqMA remains open since this class was defined in 2006 [9] because this class does not permit amplification of gap between thresholds a, b based on majority voting. Hence, this gap is at least an inverse polynomial. We leave the remarks regarding the non-negativity of witnesses and parameters to Remark 10.

▶ Remark 10 (Optimal witnesses of a StoqMA verifier is non-negative). Analogous to QMA, the maximum acceptance probability of a StoqMA verifier V_x is precisely the maximum eigenvalue of $M_x := \langle\bar{0}| \langle\bar{+}| V_x^\dagger |+\rangle \langle+|_1 V_x |\bar{0}\rangle |\bar{+}\rangle$ due to $\Pr[V_x \text{ accepts } |\psi\rangle] = \langle\psi| M_x |\psi\rangle$. Notice the matrix M_x is entry-wise non-negative. Owing to the Perron-Frobenius theorem (see Theorem 8.4.4 in [24]), a straightforward corollary is that the eigenvector ψ (i.e., the optimal witness) maximizing the acceptance probability has non-negative amplitudes in the computational basis, namely it suffices to consider only non-negative witness for *yes* instances. Additionally, it is clear-cut that the acceptance probability for any non-negative witness $|\psi\rangle$, regardless of the optimality, is at least 1/2 by a direct calculation.

2.3 Distribution testing

Distribution testing is generally about telling whether one probability distribution is close to the other. We further recommend a comprehensive survey [15] for a detailed introduction. We begin with the squared Hellinger distance $d_H^2(D_0, D_1)$ between two (sub-)distributions D_0, D_1, where $d_H^2(D_0, D_1) := \frac{1}{2}\| \, |D_0\rangle - |D_1\rangle \, \|_2^2$ and $|D_k\rangle = \sum_i \sqrt{D_k(i)} \, |i\rangle$ for any $k = 0, 1$. This distance is comparable with the total variation distance (see Proposition 1 in [18]). We then introduce a specific model used for this paper, namely the *dual access model*:

▶ **Definition 11** (Dual access model, adapted from [14]). *Let D be a fixed distribution over $[2^n]$. A dual oracle for D is a pair of oracles $(\mathsf{S}_D, \mathsf{Q}_D)$:*

- *Sample access: S_D returns an element $i \in \{0,1\}^n$ with probability $D(i)$. And it is independent of all previous calls to any oracle.*
- *Query access: Q_D takes an input a query element $j \in \{0,1\}^{n-1}$, and returns the quotient $D(0||j)/D(1||j)$ where $D(a||j)$ is the probability weight that D puts on $a||j$ for $a \in \{0,1\}$.*

We then explain how to implement these oracles here in Remark 12:

▶ Remark 12 (Implementation of dual access model). The sample access oracle in Definition 11 could be implemented by running an independent copy of the circuit that generates the state $|0\rangle |D_0\rangle + |1\rangle |D_1\rangle$, and measuring all qubits on the computational basis. Meanwhile, the query access oracle is substantially an efficient evaluation algorithm corresponding to the quotient $D_0(i)/D_1(i)$ for given index i.

In [14], Canonne and Rubinfeld show that approximating the total variation distance between two distributions within an additive error ϵ requires only $\Theta(1/\epsilon^2)$ oracle accesses (see Theorems 6 and 7 in [14]). However, suppose we allow to utilize only sample accesses. In that case, such a task requires $\Omega(N/\log N)$ samples even within constant accuracy (see Theorem 9 in [18]), where N is the dimension of distributions.

3 StoqMA with easy witnesses

This section will prove that StoqMA with easy witnesses, viz. eStoqMA, is contained in MA. *Easy witness* is named in the flavor of the seminal *easy witness lemma* [25], which means that an n-qubit non-negative state witness of a StoqMA verifier has a *succinct* representation. In particular, there exists an efficient algorithm to output the quotient $D_0(i)/D_1(i)$ for given index i. It is a straightforward generalization of subset states where the membership of the corresponding subset is efficiently verifiable. We here define eStoqMA formally:

▶ **Definition 13** (eStoqMA). *A promise problem $\mathcal{L} = (\mathcal{L}_{\text{yes}}, \mathcal{L}_{\text{no}}) \in$ eStoqMA if there is a StoqMA verifier such that for any input $x \in \mathcal{L}$, a uniformly generated verification circuit V_x using only Toffoli, CNOT, X gates on $n := n_w + n_0 + n_+$ qubits and a Hadamard-basis measurement on the output qubit, where n_w is the number of qubits for a witness, and n_0 (or n_+) is the number of $|0\rangle$ (or $|+\rangle$) ancillary qubits, such that for efficiently computable functions $a(n)$ and $b(n)$:*

Completeness. *There exists an n-qubit non-negative witness $|w\rangle := \sum_{i\in\{0,1\}^n} \sqrt{D_w(i)}\,|i\rangle$ such that $\Pr[V_x \text{ accepts } |w\rangle] \geq a(n)$, and there is an efficient algorithm Q_w that outputs $D_w(0||i)/D_w(1||i)$ (or $D_w(1||i)/D_w(0||i)$) of index $1||i$ (or $0||i$) sampled from the distribution D_w where $i \in \{0,1\}^{n-1}$.*

Soundness. *For any n-qubit witness $|w\rangle$, $\Pr[V_x \text{ accepts } |w\rangle] \leq b(n)$.*

Moreover, $a(n)$ and $b(n)$ satisfy $1/2 \leq b(n) < \alpha(n) \leq 1$ and $a(n) - b(n) \geq 1/\text{poly}(n)$.

▶ Remark 14 (Subset-state witnesses require only membership). To show a subset-state witness $|w\rangle$ is an easy witness, it suffices to decide the membership of $\text{supp}(|w\rangle)$ for the associated algorithm Q_w. Notice any coordinate $D_w(j)$ in D_w is $1/|\text{supp}(|w\rangle)|$ if $j \in \text{supp}(|w\rangle)$; otherwise $D_w(j) = 0$. Moreover, if $D_w(1||i) = 0$ for some i, the corresponding point will never be sampled. Hence, the quotient $D_w(0||i)/D_w(1||i)$ is 1 if both $0||i$ and $1||i$ belong to $\text{supp}(|w\rangle)$ (i.e., $D_w(0||i) = D_w(1||i) \neq 0$); otherwise the quotient is 0.

Distribution testing techniques inspire an MA containment of eStoqMA, as Theorem 15. Precisely, employed with the dual access model (see Definition 11) adapted from Canonne and Rubinfeld [14], we obtain an empirical estimation within inverse-polynomial accuracy of an eStoqMA verifier's acceptance probability, where both sample complexity and time complexity are efficient.

▶ **Theorem 15** (eStoqMA ⊆ MA). *For any $1/2 \leq b < a \leq 1$ and $a - b \geq 1/\text{poly}(n)$,*

$$\mathsf{eStoqMA}(a,b) \subseteq \mathsf{MA}\left(\tfrac{9}{16}, \tfrac{7}{16}\right).$$

In [9, 12], Bravyi, Bessen, and Terhal proved $\mathsf{StoqMA}_1 \subseteq \mathsf{MA}$, utilizing a relatively complicated random walk based argument. By taking advantage of eStoqMA, we here provide an arguably simplified proof by plugging Proposition 16 into Theorem 15:

▶ **Proposition 16.** $\mathsf{StoqMA}_1 \subseteq \mathsf{eStoqMA}$.

The proof of Proposition 16 straightforwardly follows from the definition of SetCSP (see Definition 30), namely any $\text{SetCSP}_{0,1/\text{poly}}$ instance certainly has easy witness, and it is indeed optimal. We further leave the technical details regarding SetCSP in Appendix B.

How strong is the eStoqMA? Remark 17 suggests eStoqMA seems more powerful than classical-witness StoqMA (i.e., cStoqMA):

▶ Remark 17 (eStoqMA is not trivially contained in cStoqMA). Classical witness is clearly also easy witness, but the opposite is unnecessarily true. Even though Merlin could send the algorithm Q_{D_w} as classical witness to Arthur, Arthur only can prepare $|w\rangle$ by a post-selection, which means cStoqMA does not trivially contain eStoqMA.

Furthermore, the proof of StoqMA(a,b) with classical witnesses is in MA [23] could preserve completeness and soundness parameters. By inspection, it is clear-cut that this proof even holds when the gap $a-b$ is *arbitrarily small*, whereas the proof of Theorem 15 works only for inverse-polynomial accuracy. Further remarks of classical witness' limitations can be found in Section 3.3.

3.1 eStoqMA $\subseteq$ MA: the power of distribution testing

To derive an MA containment of eStoqMA, it suffices to distinguish two non-negative states (viz., approximating the maximum acceptance probability) within an inverse-polynomial accuracy regarding the inner product (i.e., squared Hellinger distance). It seems plausible to prove StoqMA $\subseteq$ MA by taking *samples* and post-processing. However, the known sample complexity lower bound (See Section 2.3) indicates that (almost) exponentially many samples are unavoidable. Fortunately, we could circumvent this barrier for showing eStoqMA $\subseteq$ MA, since easy witness guarantees *efficient query access* to $D_0(i)/D_1(i)$ for given index i. In particular, employing both sample and query oracle accesses to D_0, D_1, such approximation within an additive error ϵ requires merely $\Theta(1/\epsilon^2)$ samples and queries! This advantage first noticed by Rubinfeld and Servedio [35], and then almost fully characterized by Canonne and Rubinfeld [14]. Recently, this technique also has algorithmic applications used in quantum-inspired classical algorithms for machine learning [16, 38].

▶ **Lemma 18** (Approximating a single-qubit Hadamard-basis measurement). *In the dual access model, there is a randomized algorithm $\mathcal{T}$ which takes an input x, $1/2 \leq b(|x|) < a(|x|) \leq 1$, as well as access to $(\mathsf{S}_D, \mathsf{Q}_D)$, where the non-negative state before the measurement is $|\psi\rangle = \sum_{i\in[2^n]} \sqrt{D(i)}\,|i\rangle$. After making $O\left(1/(a-b)^2\right)$ calls to the oracles, $\mathcal{T}$ outputs either* ACCEPT *or* REJECT *such that:*

- *If $\frac{1}{2}\|\,|D_0\rangle + |D_1\rangle\,\|_2^2 \geq a$, $\mathcal{T}$ outputs* ACCEPT *with probability at least* $9/16$*;*
- *If $\frac{1}{2}\|\,|D_0\rangle + |D_1\rangle\,\|_2^2 \leq b$, $\mathcal{T}$ outputs* ACCEPT *with probability at most* $7/16$*,*

where D_k $(k\in\{0,1\})$ is a sub-distribution such that $\forall i \in \{0,1\}^{n-1}, D_k(i) := D(k||i)$.

Proof Intuition. To construct this algorithm $\mathcal{T}$, the main idea is writing the acceptance probability p_{acc} of a StoqMA verifier's easy witness as an expectation over D_1 (or D_0) of some random variable regarding coordinates quotients $D_0(i)/D_1(i)$. Note that the quotient $\sqrt{D_0(i)}/\sqrt{D_1(i)}$ could be computed by running the evaluation algorithm Q_w (i.e., query oracle access). Hence, $\mathcal{T}$ only require to calculate an empirical estimation of $\mathbb{E}[X]$ (see the RHS of Equation (1)) within $1/\text{poly}(|x|)$ accuracy. Such an approximation could be achieved by averaging poly$(|x|)$ sample with a standard concentration bound, which is analogous to Theorem 6 in [14].

Now we proceed with the explicit construction (i.e., Algorithm 1) and analysis.

Proof of Lemma 18. We begin with estimating the quantity $\||D_0\rangle + |D_1\rangle\|_2^2 / 2\,\|D_1\|_1$ up to some additive error $\epsilon := (a-b)/8$. We first observe that

$$\frac{\||D_0\rangle+|D_1\rangle\|_2^2}{2\|D_1\|_1} = \frac{1}{2}\sum_{i\in\{0,1\}^{n-1}} \left(1+\frac{\sqrt{D_0(i)}}{\sqrt{D_1(i)}}\right)^2 \frac{D_1(i)}{\|D_1\|_1} = \mathop{\mathbb{E}}_{i\sim D_1/\|D_1\|_1}\left[\frac{1}{2}\left(1+\frac{\sqrt{D_0(i)}}{\sqrt{D_1(i)}}\right)^2\right]. \quad (1)$$

Since the inner product is symmetric, it also implies $\frac{\||D_0\rangle+|D_1\rangle\|_2^2}{2\|D_0\|_1} = \mathbb{E}_{i\sim\frac{D_0}{\|D_0\|_1}}\left[\frac{1}{2}\left(1+\frac{\sqrt{D_1(i)}}{\sqrt{D_0(i)}}\right)\right]$.

Algorithm 1 $O(1/(a-b)^2)$-additive approximation tester $\mathcal{T}$ of $\frac{1}{2}\left\||D_0\rangle+|D_1\rangle\right\|_2^2$.

```
Require: $\mathsf{S}_D$ and $\mathsf{Q}_D$ oracle accesses; parameters $\frac{1}{2} \le b < a \le 1$.
Set $m, m' := \Theta(1/\epsilon^2)$, where $\epsilon := (a-b)/8$;
Draw samples $o_1, \cdots, o_{m'}$ from
  $D_{\mathrm{out}}$ := marginal distribution of the designated output qubit;
Compute $\hat{Z} := \frac{1}{m'}\sum_{i=1}^{m'} Z_i$, where $Z_i := o_i$;
Draw samples $s_1, \cdots, s_m$ from $D$;
For $i = 1, \cdots, m$ Do
    If $\hat{Z} \ge \frac{1}{2}$ Then with $\mathsf{Q}_D$, get $X_i := \frac{1}{2}\left(1+\frac{\sqrt{D_0(s_i)}}{\sqrt{D_1(s_i)}}\right)^2$;
    Else with $\mathsf{Q}_D$, get $X_i := \frac{1}{2}\left(1+\frac{\sqrt{D_1(s_i)}}{\sqrt{D_0(s_i)}}\right)^2$;
End
Compute $\hat{X} := \frac{1}{m}\sum_{i=1}^{m} X_i$;
If $\hat{Z} \ge \frac{1}{2}$ and $\hat{X}\hat{Z} \ge \frac{1}{2}(a+b)$ Then output ACCEPT;
Else If $\hat{Z} < \frac{1}{2}$ and $\hat{X}(1-\hat{Z}) \ge \frac{1}{2}(a+b)$ Then output ACCEPT;
Else output REJECT;
```

Notice $\mathcal{T}$ only require to achieve an empirical estimate of this expected value, which suffices to utilize $m = O\left(1/(a-b)^2\right)$ samples s_i from D_1, querying $\frac{D_0(s_i)}{D_1(s_i)}$, and computing $X_i = \frac{1}{2}\left(1+\frac{\sqrt{D_0(s_i)}}{\sqrt{D_1(s_i)}}\right)^2 \|D_1\|_1$. We here provide the explicit construction of $\mathcal{T}$, as Algorithm 1.

Analysis. Define random variables Z_i as in Algorithm 1. We obviously have $\mathbb{E}[Z_i] = \|D_1\|_1 \in [0,1]$. Since all Z_is' are independent, a Chernoff bound ensures

$$\Pr\left[\left|\hat{Z} - \|D_1\|_1\right| \le \epsilon\right] \ge 1 - 2e^{-2m'/\epsilon^2}, \tag{2}$$

which is at least 3/4 by an appropriate choice of m'.

Note drawing samples from p_0 implicitly by post-selecting the output qubit to be 0. However, due to the inner product's symmetry and $\|D_0\|_1 + \|D_1\|_1 = 1$, there must exist $i \in \{0,1\}$ such that $\|D_i\|_1 \ge 1/2$. Hence, the required sample complexity will be enlarged merely by a factor of 2.

Let us also define random variables X_i as in Algorithm 1. W.L.O.G. assume that $\|D_1\|_1 \ge 1/2 \ge \|D_0\|_1$. By Equation (1), we obtain $\mathbb{E}_{i\sim D_1/\|D_1\|_1}[X_i] = \left\||D_0\rangle+|D_1\rangle\right\|_2^2/2\|D_1\|_1$. Because the X_i's are independent and takes value in $[1/2,1]$, by Chernoff bound,

$$\Pr\left[\left|\hat{X} - \frac{\left\||D_0\rangle+|D_1\rangle\right\|_2^2}{2\|D_1\|_1}\right| \le \epsilon\right] \ge 1 - 2e^{-2m/\epsilon^2}. \tag{3}$$

Therefore, by our choice of m, $\hat{X}$ is an ϵ-additive approximation of $\left\||D_0\rangle+|D_1\rangle\right\|_2^2/2\|D_1\|_1$ with probability at least 3/4. Note that X_i, Z_i are independent, we obtain $\mathbb{E}\left[\hat{X}\hat{Z}\right] = \frac{1}{2}\left\||D_0\rangle+|D_1\rangle\right\|_2^2$. Hence, notice $1/2 \le \|D_1\|_1 \le 1$ and $1/2 \le \frac{1}{2}\left\||D_0\rangle+|D_1\rangle\right\|_2^2 \le 1$, by combining Equations (2) and (3), we obtain with probability 9/16:

$$\hat{X}\hat{Z} \le \left(\frac{\left\||D_0\rangle+|D_1\rangle\right\|_2^2}{2\|D_1\|_1}+\epsilon\right)(\|D_1\|_1+\epsilon) \le \frac{1}{2}\left\||D_0\rangle+|D_1\rangle\right\|_2^2 + \epsilon^2 + \epsilon + 2\epsilon \le \frac{1}{2}\left\||D_0\rangle+|D_1\rangle\right\|_2^2 + 4\epsilon;$$

$$\hat{X}\hat{Z} \ge \left(\frac{\left\||D_0\rangle+|D_1\rangle\right\|_2^2}{2\|D_1\|_1}-\epsilon\right)(\|D_1\|_1-\epsilon) \ge \frac{1}{2}\left\||D_0\rangle+|D_1\rangle\right\|_2^2 + \epsilon^2 - \epsilon - 2\epsilon \ge \frac{1}{2}\left\||D_0\rangle+|D_1\rangle\right\|_2^2 - 4\epsilon.$$

It implies that $\Pr\left[\left|\hat{X}\hat{Z} - \frac{1}{2}\left\||D_0\rangle + |D_1\rangle\right\|_2^2\right| \leq 4\epsilon\right] \geq 9/16$. We thereby conclude that

- If $\frac{1}{2}\left\||D_0\rangle + |D_1\rangle\right\|_2^2 \geq a$, then $\hat{X}\hat{Z} \geq a - 4\epsilon$ and $\mathcal{T}$ outputs ACCEPT w.p. at least 9/16.
- If $\frac{1}{2}\left\||D_0\rangle + |D_1\rangle\right\|_2^2 \leq b$, then $\hat{X}\hat{Z} \leq b + 4\epsilon$ and $\mathcal{T}$ outputs ACCEPT w.p. at most 7/16.

Furthermore, the algorithm $\mathcal{T}$ makes $m' + 2m$ calls for S_D and m calls for Q_D . ◀

It is worthwhile to mention that this construction in the proof of Theorem 15 is optimal regarding the sample complexity, as Theorem 7 stated in [14].

Finally, we complete the proof of Theorem 15 by Lemma 18.

Proof of Theorem 15. Given an $\mathsf{eStoqMA}(a, b)$ verifier V_x, we here construct a MA verifier V_x' that follows from Algorithm 1 in the proof of Lemma 18:

(1) For each call to the sample oracle S_{D_w}, we run the $\mathsf{eStoqMA}$ verifier V_x (without measuring the output qubit) with the witness w, and draw samples by performing measurements:
   - For samples s_i $(1 \leq i \leq m)$ from distribution D, measure all qubits utilized by the verification circuit in the computational basis;
   - For samples o_j $(1 \leq j \leq m')$ from distribution D_{out}, measure the designated output qubit in the computational basis.

(2) For each call to the query oracle Q_{D_w} with index i, find the corresponding index i' at the beginning by performing the permutation associated with $V_x^\dagger$ on i, and then evaluate the value $D_w(i'')/D_w(i')$ by utilizing the given algorithm associated with this easy witness, where i'' is given by flipping the first bit of i'.

(3) Compute an empirical estimation of $\frac{1}{2}\left\||D_0\rangle + |D_1\rangle\right\|_2^2$ as Algorithm 1, and then decide whether V_x accepts w.

The circuit size of V_x' is a polynomial of $|x|$ since both sample and query complexity are efficient. We thus conclude that the new MA verifier V_x' is efficient, and only requires $O\left(1/(a-b)^2\right)$ copies of the witness w, which finishes the completeness case.

For the soundness case, the acceptance probability p_{acc} of the $\mathsf{eStoqMA}$ verifier V_x for all witnesses is obviously upper-bounded by b, regardless of whether such a witness is easy or not. Furthermore, entangled witnesses are useless since we draw samples by performing measurements separately. Hence, the maximum acceptance probability of the new MA verifier V_x' is also at most b. ◀

3.2 StoqMA with perfect completeness is in eStoqMA

We here complete proof of Proposition 16. By Theorem 15, it infers $\mathsf{StoqMA}_1 \subseteq \mathsf{MA}$.

Proof of Proposition 16. By Theorem 31, we know that $\text{SetCSP}_{0,1/\text{poly}}$ is StoqMA_1-complete, so it suffices to show that $\text{SetCSP}_{0,1/\text{poly}}$ is contained in $\mathsf{eStoqMA}_1$.

By Lemma 35, given a $\text{SetCSP}_{0,b}$ instance C, we can construct a $\mathsf{StoqMA}\left(1, 1 - b/2\right)$ verifier. The corresponding subset $S \subseteq \{0,1\}^n$, where S satisfies all set-constraints of C, is an optimal witness. It is left to show that this subset states is an easy witness.

We achieve the proof by inspection. Let S be the set of all good strings of C, then set-unsat$(C, S) = 0$. Note $x \in S$ is a good string of C iff x is a good string of all set-constraints $C_i (1 \leq i \leq m)$, the membership of S thus can be decided efficiently, which infers the subset state $|S\rangle$ is easy witness by Remark 14. ◀

3.3 Limitations of classical-witness StoqMA

As we have shown StoqMA with easy witness is contained in MA. What about classical witness, namely cStoqMA? In fact, we could show such a containment that preserves both completeness and soundness parameters.

▶ **Proposition 19** ([23]). *For any $1/2 \leq b < a \leq 1$ and $a - b \geq 1/\text{poly}(n)$, $\mathsf{cStoqMA}(a, b) \subseteq \mathsf{MA}(2a - 1, 2b - 1)$.*

Proof Sketch. We only illustrate the intuition: for any $s \in \{0,1\}^n$ and any reversible circuit U, we have $\langle s| U^\dagger |+\rangle \langle +|_1 U |s\rangle = \frac{1}{2} + \frac{1}{2} \langle s| U^\dagger X_1 U |s\rangle$ since $|+\rangle \langle +| = \frac{1}{2}(X + I)$. The detailed proof is left in Appendix A.1. ◀

The proof of Proposition 19 immediately infers the *precise variant* of StoqMA with classical witnesses, where the completeness-soundness gap is exponentially small, is equal to PreciseMA. However, the proof of Theorem 15 no longer works for precise scenarios, indicating that StoqMA with classical witness seems not interesting.

Furthermore, it is not hard to see that classical witness is optimal for StoqMA_1 verifier[7]. However, it does not mean that a classical witness is optimal for *any* StoqMA_1 verifier. In fact Appendix A.2 provides a simple counterexample by considering an identity as a verifier. However, this impossibility result is unknown for easy witness yet.

4 Complexity of reversible circuit distinguishability

This section will concentrate on the complexity classification of distinguishing reversible circuits, namely given two efficient reversible circuits, and decide whether there is a non-negative state that *cannot* tell one from the other. With ancillary random bits, this problem is StoqMA-complete, as Theorem 22. However, this problem's exact variant, namely assuming two reversible circuits are indistinguishable with respect to any non-negative witness for *no* instances (viz., StoqMA with perfect soundness), is NP-complete (see Proposition 23). Moreover, Theorem 22 also implies that distinguishing reversible circuits without any ancillary random bit is NP-complete, which signifies a simplified proof of [27].

4.1 Reversible circuit distinguishability is StoqMA-complete

We begin with the formal definition of the *Reversible Circuit Distinguishability* problem.

▶ **Definition 20** (Reversible Circuit Distinguishability). *Given a classical description of two reversible circuits C_0, C_1 (using Toffoli, CNOT, X gates) on $n := n_w + n_0 + n_+$ qubits, where n_w is the number of qubits of a non-negative state witness $|w\rangle$, n_0 is the number of $|0\rangle$ ancillary qubits, and n_+ is the number of $|+\rangle$ ancillary qubits. Let the resulting state before measuring the output qubit be $|R_i\rangle := C_i |w\rangle |\bar{0}\rangle |\bar{+}\rangle$, $i \in \{0,1\}$. Promise that C_0 and C_1 with respect to witness state(s) are either α-indistinguishable or β-distinguishable, decide whether*

- **Yes** *(α-indistinguishable): there exists a non-negative witness $|w\rangle$ such that $\langle R_0|R_1\rangle \geq \alpha$;*
- **No** *(β-distinguishable): for any non-negative witness $|w\rangle$, then $\langle R_0|R_1\rangle \leq \beta$;*

where $\alpha - \beta \geq 1/\text{poly}(n)$[8].

[7] By combining $\mathsf{StoqMA}_1 \subseteq \mathsf{MA}_1$ and the gadget in the proof of Proposition 36, we could construct a StoqMA_1 verifier such that a classical witness is optimal.

[8] Note $\langle R_0|R_0\rangle = \langle R_1|R_1\rangle = 1$ which differs from $\langle D_0|D_0\rangle + \langle D_1|D_1\rangle = 1$ previously used in Section 3, we obtain that the acceptance probability $p_{\text{acc}} = \frac{1}{2} + \frac{1}{2}\langle R_0|R_1\rangle = 1 - \frac{1}{2} \cdot \frac{1}{2}\| |R_0\rangle - |R_1\rangle \|_2^2$.

Since Definition 20 seems slightly inconsistent with known results regarding distinguishing circuits [26, 27, 37], it is worthwhile to mention a slightly different version (see Remark 21) of Definition 20, which is co-StoqMA-complete.

▶ Remark 21 (Equivalence Check of Reversible Circuits is co-StoqMA-complete). Consider the same scenario in Definition 20, and the task is checking whether C_0 and C_1 are approximately equivalent (with respect to witness states). More concretely, decide whether $\langle R_0|R_1\rangle \geq \alpha$ for any $|w\rangle$; or there exists $|w\rangle$ such that $\langle R_0|R_1\rangle \leq \beta$. The co-StoqMA-completeness straightforwardly follows from the constructions in the proof of Theorem 22.

Now we state the main theorem in Section 4.

▶ **Theorem 22** (Reversible Circuit Distinguishability is StoqMA-complete). *For any $\alpha - \beta \geq 1/\mathrm{poly}(n)$, (α, β)-Reversible Circuit Distinguishability is* StoqMA $(1/2 + \alpha/2, 1/2 + \beta/2)$-*complete.*

We will then proceed with an intuitive explanation regarding proof of Theorem 22.

Proof Intuition. The StoqMA-containment proof is inspired by the SWAP test for distinguishing two quantum states [13], since it could be thought of as a StoqMA verification circuit with the maximum acceptance probability 1. We below provide a procedure (see Figure 1) to distinguish two reversible circuits C_0, C_1 using a non-negative witness, and such a procedure is apparently a StoqMA verifier. The StoqMA-hardness proof is straightforward: replacing C_0 and C_1 by identity and $V_x^\dagger X_1 V_x$ (see Figure 2), respectively, where V_x is the given StoqMA verification circuit.

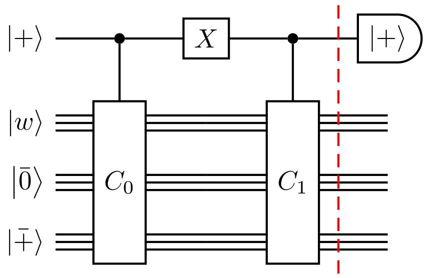


Figure 1 RCD is in StoqMA.

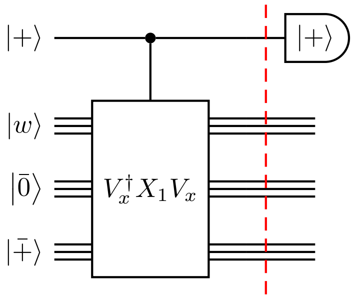


Figure 2 RCD is StoqMA-hard.

Now we proceed with the technical details.

Proof of Theorem 22. We first show (α, β)-RCD is StoqMA $(1/2 + \alpha/2, 1/2 + \beta/2)$-hard. Consider a StoqMA verifier V_x as Figure 2, let $C_0 := V_x^\dagger X_1 V_x$ where the X gate in the middle acts on the output qubit, and let C_1 be identity. Then for any witness $|w\rangle$, we obtain:

$$\begin{aligned} \Pr\left[V_x \text{ accepts } |w\rangle\right] &= \langle w| \langle \bar{0}| \langle \bar{+}| \left(V_x^\dagger |+\rangle\langle +|_1 V_x\right) |w\rangle |\bar{0}\rangle |\bar{+}\rangle ; \\ \langle R_0|R_1\rangle &= \langle w| \langle \bar{0}| \langle \bar{+}| \left(V_x^\dagger X_1 V_x\right) |w\rangle |\bar{0}\rangle |\bar{+}\rangle . \end{aligned} \tag{4}$$

Note that $|+\rangle\langle +| = (X + I)/2$, we thereby complete the StoqMA-hardness proof by Equation (4): $\Pr\left[V_x \text{ accepts } |w\rangle\right] = 1/2 + \langle R_0|R_1\rangle/2$.

Now it is left to show the StoqMA $(1/2 + \alpha/2, 1/2 + \beta/2)$ containment of (α, β)-RCD. Given reversible circuits C_0, C_1, we construct a StoqMA verifier as Figure 1. Hence, we obtain the state before measuring the output qubit (viz. the red dash line):

$$\text{Ctrl}-C_1 \cdot X_1 \cdot \text{Ctrl}-C_0 \left(\frac{|0\rangle + |1\rangle}{\sqrt{2}} \otimes |w\rangle \left|\bar{0}\right\rangle \left|\bar{+}\right\rangle \right) = \frac{1}{\sqrt{2}} |0\rangle |R_0\rangle + \frac{1}{\sqrt{2}} |1\rangle |R_1\rangle := |\text{RHS}\rangle .$$

We thus complete the StoqMA-containment proof:
$\Pr[V_x \text{ accepts } |w\rangle] = \| |+\rangle \langle +|_1 |\text{RHS}\rangle \|_2^2 = 1/2 + \langle R_0 | R_1 \rangle /2.$ ◀

4.2 Exact Reversible Circuit Distinguishability is NP-complete

We will prove that the exact variant of the Reversible Circuit Distinguishability is NP-complete. Moreover, it will signify that StoqMA with perfect soundness (even the gap between thresholds $\alpha, 1/2$ is arbitrarily small) is in NP.

▶ **Proposition 23** (Exact RCD is NP-complete). *Exact Reversible Circuit Distinguishability (RCD), namely* $(\alpha, 0)$*-Reversible Circuit Distinguishability for any* $0 \leq \alpha < 1$*, is* NP*-complete.*

Proof Sketch. It suffices to show an NP containment. By an analogous idea in [21], we could find two matched pairs (s, r) and (s', r') as classical witness, where s, s' are indices of non-zero coordinates in the given witness, and r, r' are random bit strings. Specifically, for *yes* instances, there exist two such pairs such that the resulting strings $C_0(s, r)$ [9] and $C_1(s', r')$ are identical; whereas it is evident that no matched pairs exist for *no* instances. The details are left in Appendix A.3. ◀

As a corollary, Proposition 23 will imply StoqMA with perfect soundness is in NP:

▶ **Corollary 24** (StoqMA with perfect soundness is in NP). $\bigcup_{a>1/2} \mathsf{StoqMA}\left(a, \frac{1}{2}\right) = \mathsf{NP}$.

StoqMA without any ancillary random bit is in NP. In fact, distinguishing reversible circuits without any ancillary random bit is NP-complete. By analogous reasoning, we also provide an alternating proof of *Strong Equivalence of Reversible Circuits* is co-NP-complete [27]. We leave the detailed proof in Appendix A.4.

5 Soundness error reduction of StoqMA

In this section, we will partially solve Conjecture 6 by providing a procedure that reduces the soundness error of any StoqMA verifier.

▶ **Theorem 25** (restated of Theorem 5). *For any* $r = \text{poly}(n)$*,*

$$\mathsf{StoqMA}\left(\frac{1}{2} + \frac{a}{2}, \frac{1}{2} + \frac{b}{2}\right) \subseteq \mathsf{StoqMA}\left(\frac{1}{2} + \frac{a^r}{2}, \frac{1}{2} + \frac{b^r}{2}\right).$$

Consequently, Theorem 25 infers a direct error reduction for StoqMA_1 by choosing appropriate parameters a, b, r.

▶ **Corollary 26** (Error reduction of StoqMA_1). *For any s such that* $1/2 \leq s \leq 1$ *and* $1 - s \geq 1/\text{poly}(n)$*,* $\mathsf{StoqMA}(1, s) \subseteq \mathsf{StoqMA}\left(1, 1/2 + 2^{-n}\right)$.

Proof. Choosing a, b such that $1 = 1/2 + a/2$ and $s = 1/2 + b/2$, we have $a = 1$ and $b = 2s - 1$. By Theorem 25, we obtain $\mathsf{StoqMA}\left(\frac{1}{2} + \frac{1}{2} \cdot 1, \frac{1}{2} + \frac{1}{2}(2s-1)\right) \subseteq \mathsf{StoqMA}\left(1, \frac{1}{2} + \frac{1}{2}(2s-1)^r\right)$. To finish the proof, it remains to choose a parameter r such that $r \geq (n+1)/\log_2(1/(2s-1))$, since $(2s-1)^r/2 \leq 2^{-n}$ implies that $2^{-r\log_2(1/(2s-1))-1} \leq 2^{-n}$. ◀

[9] A reversible circuit takes (s, r) as an input, and permutes it to the other binary string as the output.

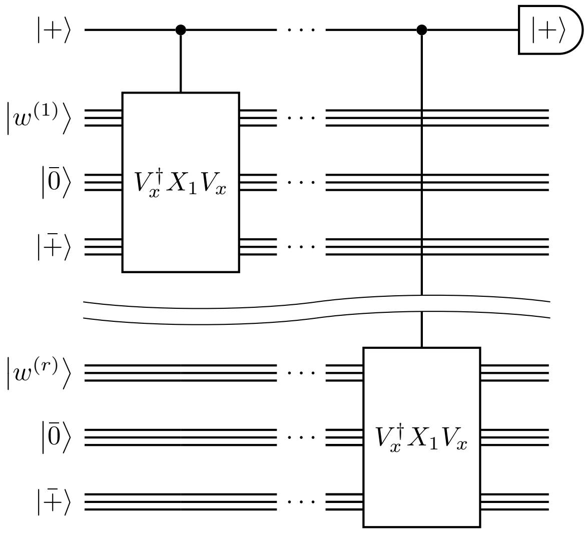


Figure 3 AND-type repetition procedure of a StoqMA verifier.

5.1 AND-type repetition procedure of a StoqMA verifier

Proof Intuition. The main idea is doing a parallel repetition of a StoqMA verifier V_x, and taking the conjunction (viz., AND) of the outcomes cleverly. More concretely, given a StoqMA verification circuit V_x where x is in $\mathcal{L} \in$ StoqMA, we result in a new StoqMA verifier by separately substituting an identity and $V_x^\dagger X_1 V_x$ for C_0, C_1 (as Figure 2). Notice the acceptance probability of a StoqMA verifier's non-negative witness $|w\rangle$, $\Pr[V_x \text{ accepts } |w\rangle] = \frac{1}{2} + \frac{1}{2}\langle D_0|D_1\rangle$, is linearly dependent to an inner product between states associated with two distributions D_0, D_1 where $|D_0\rangle := |w\rangle |\bar{0}\rangle |\bar{+}\rangle$ and $|D_1\rangle := V_x |w\rangle |\bar{0}\rangle |\bar{+}\rangle$. We could then take advantage of this new StoqMA verifier by running $r = \text{poly}(|x|)$ copies of these reversible circuits parallelly with the same target qubit, which is denoted as V_x' (see Figure 3).

For *yes* instances, it follows that an inner product of two tensor products of distributions is equal to the product of inner products of states associated with these distributions, namely, $\Pr[V_x' \text{ accepts } |w\rangle] = \frac{1}{2} + \frac{1}{2}\langle D_0|D_1\rangle^r$. However, it seems problematic for *no* instances, since a dishonest prover probably wants to cheat with an entangled witness instead of a tensor product among repetitive verifiers. We resolve this issue by an observation used in the QMA error reduction [30]: the maximum acceptance probability of a verifier V_x is the same as the maximum eigenvalue of a projection $\Pi_0 V_x^\dagger \Pi_1 V_x \Pi_0$ where Π_1 is the final measurement on the designated output qubit and $\Pi_0 := |\bar{0}\rangle\langle\bar{0}| \otimes |\bar{+}\rangle\langle\bar{+}|$. Eventually, an entangled witness will not help a dishonest prover. This is because the maximum eigenvalue of the tensor product of the projection $\Pi_0 V_x^\dagger \Pi_1 V_x \Pi_0$ is also the product of the maximum eigenvalue of this projection.

Finally, we proceed with the proof of Theorem 25.

Proof of Theorem 25. Given a promise problem $\mathcal{L} = (\mathcal{L}_{\text{yes}}, \mathcal{L}_{\text{no}}) \in \text{StoqMA}(1/2+a/2, 1/2+b/2)$. For any input $x \in \mathcal{L}$, we have a StoqMA verifier V_x which is equivalent to a new StoqMA verifier $\tilde{V}_x$ as Figure 2, by the StoqMA-hardness proof of reversible circuit distinguishability as Theorem 22. Namely, $\tilde{V}_x$ is starting on a $|+\rangle$ ancillary qubit, applying a controlled-unitary $V_x^\dagger X_1 V_x$ on $n_w + n_0 + n_+$ qubits, and measuring the designated output qubit.

Let $|R_w\rangle := |w\rangle \left|\bar{0}\right\rangle |\bar{+}\rangle$ where $|w\rangle$ is a witness, we obtain

$$\left\| |+\rangle \langle+|_1 \left(\frac{1}{\sqrt{2}} |0\rangle \otimes |R_w\rangle + \frac{1}{\sqrt{2}} |1\rangle \otimes \left(V_x^\dagger X_1 V_x\right) |R_w\rangle \right) \right\|_2^2 = \| \, |+\rangle \langle+|_1 V_x |R_w\rangle \, \|_2^2. \qquad (5)$$

By an observation used in the QMA error reduction, namely Lemma 14.1 in [30], we notice that the maximum acceptance probability of a StoqMA verifier V_x is proportion to the maximum eigenvalue of a matrix $M_x := \left\langle\bar{0}\right| \langle\bar{+}| V_x^\dagger X_1 V_x \left|\bar{0}\right\rangle |\bar{+}\rangle$ associated with V_x:

$$\Pr\left[V_x \text{ accepts } |w\rangle\right] = \frac{1}{2} + \frac{1}{2} \max_{|w\rangle} \operatorname{Tr}(M_x |w\rangle \langle w|) = \frac{1}{2} + \frac{1}{2}\lambda_{\max}(M_x). \qquad (6)$$

AND-type repetition procedure of a StoqMA verifier. We now construct a new StoqMA verifier V_x' using r copies of the witness $|w\rangle$ on $r(n_w + n_0 + n_+) + 1$ qubits. As Figure 3, V_x' is starting from a $|+\rangle$ ancillary qubit as a control qubit, then applying controlled-unitary $V_x^\dagger X_1 V_x$ on qubits associated with different copies of the witness $\left|w^{(i)}\right\rangle$ for any $1 \leq i \leq r$.

By an analogous calculation of Equation (5), we have derived the acceptance probability of a witness $w^{(1)} \otimes \cdots \otimes w^{(k)}$ of the new StoqMA verifier V_x':

$$\Pr\left[V_x' \text{ accepts } \left(w^{(1)} \otimes \cdots \otimes w^{(r)}\right)\right] = \frac{1}{2} + \frac{1}{2}\operatorname{Tr}\left(\left|w^{(i)}\right\rangle \left\langle w^{(i)}\right| M_x^{\otimes r}\right),$$

where M_x is defined in Equation (6). Hence, the maximum acceptance probability of V_x':

$$\max_{|w'\rangle} \Pr\left[V_x' \text{ accepts } |w'\rangle\right] = \frac{1}{2} + \frac{1}{2}\lambda_{\max}\left(M_x^{\otimes r}\right) = \frac{1}{2} + \frac{1}{2}\left(\lambda_{\max}(M_x)\right)^r, \qquad (7)$$

where the second equality thanks to the property of the tensor product of matrices. Equation (7) indicates that entangled-state witnesses are harmless since any entangled-state witness' acceptance probability is not larger than a tensor-product state witness'.

Finally, we complete the proof by analyzing the maximum acceptance probability of the new StoqMA verifier V_x' regarding the promises: For *yes* instances, we obtain $\lambda_{\max}(M_x) \geq a$ since there exists $|w\rangle$ such that $\Pr\left[V_x \text{ accepts } |w\rangle\right] \geq 1/2 + a/2$. By Equation (7), we have derived $\Pr\left[V_x' \text{ accepts } |w\rangle^{\otimes r}\right] = \frac{1}{2} + \frac{1}{2}\left(\lambda_{\max}(M_x)\right)^r \geq \frac{1}{2} + \frac{a^r}{2}$. For *no* instances, we have $\lambda_{\max}(M_x) \leq b$ since $\Pr\left[V_x \text{ accepts } |w\rangle\right] \leq 1/2 + b/2$ for all witness $|w\rangle$. By Equation (7), we further deduce $\forall w', \Pr\left[V_x' \text{ accepts } |w'\rangle\right] = \frac{1}{2} + \frac{1}{2}\left(\lambda_{\max}(M_x)\right)^r \leq \frac{1}{2} + \frac{b^r}{2}$. ◀

References

1 Scott Aaronson, Robin Kothari, William Kretschmer, and Justin Thaler. Quantum lower bounds for approximate counting via laurent polynomials. In *Proceedings of the 35th Computational Complexity Conference*, pages 1–47, 2020.

2 Scott Aaronson and Patrick Rall. Quantum approximate counting, simplified. In *Symposium on Simplicity in Algorithms*, pages 24–32. SIAM, 2020.

3 Dorit Aharonov and Alex B Grilo. Two combinatorial ma-complete problems. In *12th Innovations in Theoretical Computer Science Conference (ITCS 2021)*. Schloss Dagstuhl-Leibniz-Zentrum für Informatik, 2021.

4 Dorit Aharonov, Alex B Grilo, and Yupan Liu. StoqMA vs. MA: the power of error reduction. *arXiv preprint arXiv:2010.02835*, 2020.

5 László Babai. Trading group theory for randomness. In *Proceedings of the seventeenth annual ACM symposium on Theory of computing*, pages 421–429, 1985.

6 Itay Berman, Akshay Degwekar, Ron D Rothblum, and Prashant Nalini Vasudevan. Statistical difference beyond the polarizing regime. In *Theory of Cryptography Conference*, pages 311–332. Springer, 2019.

7 Elmar Böhler, Christian Glaßer, and Daniel Meister. Error-bounded probabilistic computations between MA and AM. *Journal of Computer and System Sciences*, 72(6):1043–1076, 2006.

8 Gilles Brassard, Peter Hoyer, Michele Mosca, and Alain Tapp. Quantum amplitude amplification and estimation. *Contemporary Mathematics*, 305:53–74, 2002.

9 Sergey Bravyi, Arvid J Bessen, and Barbara M Terhal. Merlin-arthur games and stoquastic complexity. *arXiv preprint quant-ph/0611021*, 2006.

10 Sergey Bravyi, David P Divincenzo, Roberto Oliveira, and Barbara M Terhal. The complexity of stoquastic local hamiltonian problems. *Quantum Information & Computation*, 8(5):361–385, 2008.

11 Sergey Bravyi and Matthew Hastings. On complexity of the quantum ising model. *Communications in Mathematical Physics*, 349(1):1–45, 2017.

12 Sergey Bravyi and Barbara Terhal. Complexity of stoquastic frustration-free hamiltonians. *SIAM Journal on Computing*, 39(4):1462–1485, 2010.

13 Harry Buhrman, Richard Cleve, John Watrous, and Ronald de Wolf. Quantum fingerprinting. *Physical Review Letters*, 87(16):167902, 2001.

14 Clément Canonne and Ronitt Rubinfeld. Testing probability distributions underlying aggregated data. In *International Colloquium on Automata, Languages, and Programming*, pages 283–295. Springer, 2014.

15 Clément L Canonne. A survey on distribution testing: Your data is big. but is it blue? *Theory of Computing*, pages 1–100, 2020.

16 Nai-Hui Chia, András Gilyén, Tongyang Li, Han-Hsuan Lin, Ewin Tang, and Chunhao Wang. Sampling-based sublinear low-rank matrix arithmetic framework for dequantizing quantum machine learning. In *Proceedings of the 52nd Annual ACM SIGACT Symposium on Theory of Computing*, pages 387–400, 2020.

17 Toby Cubitt and Ashley Montanaro. Complexity classification of local hamiltonian problems. *SIAM Journal on Computing*, 45(2):268–316, 2016.

18 Constantinos Daskalakis, Gautam Kamath, and John Wright. Which distribution distances are sublinearly testable? In *Proceedings of the Twenty-Ninth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 2747–2764. SIAM, 2018.

19 Bill Fefferman, Hirotada Kobayashi, Cedric Yen-Yu Lin, Tomoyuki Morimae, and Harumichi Nishimura. Space-efficient error reduction for unitary quantum computations. In *43rd International Colloquium on Automata, Languages, and Programming (ICALP 2016)*. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik, 2016.

20 Bill Fefferman and Cedric Yen-Yu Lin. A complete characterization of unitary quantum space. In *9th Innovations in Theoretical Computer Science Conference (ITCS 2018)*. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik, 2018.

21 Martin Furer, Oded Goldreich, Yishay Mansour, Michael Sipser, and Stathis Zachos. On completeness and soundness in interactive proof systems. *Advainces in Computing Research: A Research Annual,*, 5:429–442, 1989.

22 Shafi Goldwasser and Michael Sipser. Private coins versus public coins in interactive proof systems. In *Proceedings of the eighteenth annual ACM symposium on Theory of computing*, pages 59–68, 1986.

23 Alex B. Grilo. Private communication, 2020.

24 Roger A Horn and Charles R Johnson. *Matrix analysis*. Cambridge university press, 2012.

25 Russell Impagliazzo, Valentine Kabanets, and Avi Wigderson. In search of an easy witness: Exponential time vs. probabilistic polynomial time. *Journal of Computer and System Sciences*, 65(4):672–694, 2002.

26 Dominik Janzing, Pawel Wocjan, and Thomas Beth. "non-identity-check" is QMA-complete. *International Journal of Quantum Information*, 3(03):463–473, 2005.

27 Stephen P Jordan. Strong equivalence of reversible circuits is coNP-complete. *Quantum Information & Computation*, 14(15-16):1302–1307, 2014.

28 Alastair Kay. Tutorial on the quantikz package. *arXiv preprint arXiv:1809.03842*, 2018.

29 Alexei Kitaev. Quantum NP. *Talk at AQIP*, 99, 1999.

30 Alexei Y. Kitaev, Alexander Shen, and Mikhail N. Vyalyi. *Classical and quantum computation.* American Mathematical Soc., 2002.

31 Adam R Klivans and Dieter van Melkebeek. Graph nonisomorphism has subexponential size proofs unless the polynomial-time hierarchy collapses. *SIAM Journal on Computing*, 31(5):1501–1526, 2002.

32 Peter Bro Miltersen and N Variyam Vinodchandran. Derandomizing arthur–merlin games using hitting sets. *Computational Complexity*, 14(3):256–279, 2005.

33 Tomoyuki Morimae and Harumichi Nishimura. Merlinization of complexity classes above bqp. *Quantum Information & Computation*, 17(11-12):959–972, 2017.

34 Michael A Nielsen and Isaac Chuang. Quantum computation and quantum information, 2002.

35 Ronitt Rubinfeld and Rocco A Servedio. Testing monotone high-dimensional distributions. *Random Structures & Algorithms*, 34(1):24–44, 2009.

36 Amit Sahai and Salil Vadhan. A complete problem for statistical zero knowledge. *Journal of the ACM (JACM)*, 50(2):196–249, 2003.

37 Yu Tanaka. Exact non-identity check is NQP-complete. *International Journal of Quantum Information*, 8(05):807–819, 2010.

38 Ewin Tang. A quantum-inspired classical algorithm for recommendation systems. In *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing*, pages 217–228, 2019.

39 Ramgopal Venkateswaran and Ryan O'Donnell. Quantum approximate counting with nonadaptive grover iterations. In Markus Bläser and Benjamin Monmege, editors, *38th International Symposium on Theoretical Aspects of Computer Science, STACS 2021, March 16-19, 2021, Saarbrücken, Germany (Virtual Conference)*, volume 187 of *LIPIcs*, pages 59:1–59:12. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2021.

40 Ilya Volkovich. The untold story of SBP. In *International Computer Science Symposium in Russia*, pages 393–405. Springer, 2020.

41 John Watrous. Succinct quantum proofs for properties of finite groups. In *Proceedings 41st Annual Symposium on Foundations of Computer Science*, pages 537–546. IEEE, 2000.

42 Thomas Watson. The complexity of estimating min-entropy. *Computational Complexity*, 25(1):153–175, 2016.

A Missing proofs

A.1 Proof of Proposition 19: cStoqMA $\subseteq$ MA

Proof of Proposition 19. Given a cStoqMA verifier V_x on $n = n' + n_0 + n_w$ qubits where n' is the number of qubits of a witness, we construct a new MA verifier $\tilde{V}_x$ on $n = n' + n_0 + n_w$ qubits: first run the verification circuit V_x (without measuring the output qubit), then apply an X gate on the output qubit, after that run the verification circuit's inverse $V_x^\dagger$, finally measure the first $n' + n_0$ qubits in the computational basis; $\tilde{V}_x$ accepts iff the first n' bits of the measurement outcome is exactly $s_1 \cdots s_{n'}$ and the remained bits are all zero.

We then calculate the acceptance probability of a classical witness $|s\rangle$ of a cStoqMA verifier V_x, where $w = w_1 \cdots w_{n'} \in \{0,1\}^{n'}$. Notice $|+\rangle\langle+| = \frac{1}{2}(I + X)$, we obtain

$$\begin{aligned}\Pr\left[V_x \text{ accepts } s\right] &= \| \, |+\rangle\langle+|_1 \, V_x \, |s\rangle \left|\bar{0}\right\rangle |\bar{+}\rangle \, \|_2^2 \\ &= \tfrac{1}{2} + \tfrac{1}{2} \langle s| \left\langle\bar{0}\right| \langle\bar{+}| \, V_x^\dagger \left(X \otimes I_{n-1}\right) V_x \, |s\rangle \left|\bar{0}\right\rangle |\bar{+}\rangle \, .\end{aligned} \tag{8}$$

By a direct calculation, the acceptance probability of a classical witness $|s\rangle$ of $\tilde{V}_x$:

$$\Pr\left[\tilde{V}_x \text{ accepts } s\right] = \langle R|R\rangle \text{ where } |R\rangle := \left(\langle s| \left\langle\bar{0}\right| \otimes I_{n_+}\right) V_x^\dagger \left(X \otimes I_{n-1}\right) V_x \, |s\rangle \left|\bar{0}\right\rangle |\bar{+}\rangle \, . \tag{9}$$

It is evident that $|R\rangle$ is a subset state and $\mathrm{supp}(|R\rangle) \subseteq \{0,1\}^{n_+}$. Together with Equations (8) and (9), we have completed the proof by noticing $\Pr[V_x \text{ accepts } s] = \frac{1}{2} + \frac{1}{2}\langle\bar{+}|R\rangle = \frac{1}{2} + \frac{1}{2}\langle R|R\rangle = \frac{1}{2} + \frac{1}{2}\Pr\left[\tilde{V}_x \text{ accepts } s\right]$. ◀

Could we extend Proposition 19 from a classical witness to a probabilistic witness $\sum_{s_i} \sqrt{D(i)}\,|s_i\rangle$ with a polynomial-size support[10]? Notice that the crucial equality $\langle\bar{+}|\,|R\rangle = \langle R|R\rangle$ utilized in Proposition 19 does not hold anymore, we need *an efficient evaluation algorithm* calculating $D(i)$ given an index i. Moreover, we have to calculate each coordinate's contribution on the acceptance probability *separately*, so the accumulated additive error is still supposed to be inverse-polynomial, which indicates the support size of this probabilistic witness is *negligible* for some polynomial.

A.2 Classical witness is not optimal for any StoqMA_1 verifier

▶ **Proposition 27.** *Classical witness is not optimal for any* StoqMA_1 *verifier.*

Proof. Consider a StoqMA_1 verifier V_x that uses only identity gates, then

(1) For all classical witness $s_i \in \{0,1\}^{n_w}$, $\Pr[V_x \text{ accepts } s_i] = \frac{1}{2}$ since $\langle R_0|R_1\rangle = 0$ where the resulting state before the measurement is $|0\rangle \otimes |R_0\rangle + |1\rangle \otimes |R_1\rangle$.

(2) For any classical witness $s_i, s_j \in \{0,1\}^{n_w}$ such that s_i and s_j are identical except for the first bit, one can construct a witness $|s\rangle = \frac{1}{\sqrt{2}}|s_i\rangle + \frac{1}{\sqrt{2}}|s_j\rangle$, $\Pr[V_x \text{ accepts } s] = 1$ since $\langle R_0|R_1\rangle = 1$.

We thus conclude that classical witness is not optimal for this StoqMA_1 verifier. ◀

A.3 Proof of Proposition 23: Exact RCD is NP-complete

Proof of Proposition 23. Exact RCD is NP-hard, namely $\mathsf{NP} \subseteq \mathsf{StoqMA}(1, 1/2)$, straightforwardly follows from the proof of Proposition 36. It suffices to prove that the exact RCD is in NP. By Theorem 22, $(2\alpha-1, 0)$-RCD is $\mathsf{StoqMA}(\alpha, 1/2)$-complete. Let $|w\rangle$ be an n_w-qubit non-negative witness such that $|w\rangle := \sum_{s_i \in \mathrm{supp}(w)} \sqrt{D_w(s_i)}\,|s_i\rangle$, then $\Pr[V_x \text{ accepts } |w\rangle] = \frac{1}{2} + \frac{1}{2}\langle R_0|R_1\rangle = \frac{1}{2} + \frac{1}{2}\langle w|\langle\bar{0}|\langle\bar{+}|\, C_0^\dagger C_1 \,|w\rangle|\bar{0}\rangle|\bar{+}\rangle$.

For *yes* instances, note that $\langle R_0|R_1\rangle = 2\alpha - 1$ and $\alpha > 1/2$, we have derived

$$\langle R_0|R_1\rangle = \sum_{s_i, s_j \in \mathrm{supp}(w)} \sum_{r, r' \in \{0,1\}^{n_+}} \frac{\sqrt{D_w(s_i) D_w(s_j)}}{2^{n_+}} \langle s_i|\langle\bar{0}|\langle r|\, C_0^\dagger C_1 \,|s_j\rangle|\bar{0}\rangle|r'\rangle > 0. \tag{10}$$

Since $\forall s_i, s_j$, $D_w(s_i) D_w(s_j) \geq 0$, there exists $s_i, s_j \in \mathrm{supp}(w)$ and $r, r' \in \{0,1\}^{n_+}$ such that

$$\langle s_i|\langle\bar{0}|\langle r|\, C_0^\dagger C_1 \,|s_j\rangle|\bar{0}\rangle|r'\rangle = 1. \tag{11}$$

For *no* instances, combining $\langle R_0|R_1\rangle = 0$ and Equation (10), it infers

$$\forall s_i, s_j \in \mathrm{supp}(w), \forall r, r' \in \{0,1\}^{n_+}, \langle s_i|\langle\bar{0}|\langle r|\, C_0^\dagger C_1 \,|s_j\rangle|\bar{0}\rangle|r'\rangle = 0. \tag{12}$$

We eventually construct an NP verifier as follows. The input is the classical description of two reversible circuits C_0 and C_1, and the witness is two pairs of binary strings (s_0, r_0) and (s_1, r_1). The verifier accepts iff $C_0(s_0, 0^{n_0}, r_0)$ and $C_1(s_1, 0^{n_0}, r_1)$ are identical where $C_i (i = 0, 1)$ takes $(s_i, 0^{n_0}, r_i)$ as an input and permutes it as the output. Notice these strings s_0, r_0, s_1, r_1 exists for *yes* instances owing to Equation (11), whereas they do not exist for *no* instances due to Equation (12), which achieves the proof. ◀

[10] Such witnesses are clearly easy witnesses, but not all easy witnesses have polynomial-bounded size support. See the explicit construction in Section 3.2 as an example.

A.4 StoqMA without any ancillary random bit is in NP

▶ **Proposition 28.** StoqMA *without any ancillary random bit is* NP*-complete.*

Proof. It suffices to show that StoqMA without any ancillary random bit (viz. ancillary qubits which is initially $|+\rangle$) is in NP. As a straightforward corollary of Theorem 22, distinguishing reversible circuits without $|+\rangle$ ancillary qubit is complete for StoqMA without $|+\rangle$ ancillary qubit, which is essentially NP according to Section 2.2.

Consider reversible circuits C_0 and C_1 act on $n_w + n_0$ qubits where n_0 is the number of $|0\rangle$ ancillary qubits, we observe that if C_0 and C_1 are not distinguishable with respect to any classical witness, then $\exists s \in \{0,1\}^{n_w}, \langle s| \langle \bar{0}| C_0^\dagger C_1 |s\rangle |\bar{0}\rangle = 1$ since reversible circuits C_0 and C_1 are bijections. Otherwise, it is evident that $\forall w, \langle w| \langle \bar{0}| C_0^\dagger C_1 |w\rangle |\bar{0}\rangle = 0$ provided C_0 and C_1 are distinguishable with respect to any witness. It is thus sufficient to only consider classical witnesses for distinguishing C_0 and C_1, namely, classical witness is optimal.

Now we provide an NP verifier. The input is the classical description of two reversible circuits C_0 and C_1, and the witness is a n_w-bit string s. The verifier accepts iff $C_0(s, 0^{n_0})$ is identical to $C_1(s, 0^{n_0})$. Note by inspection, the analysis is completed by above showing classical witness is optimal, which finishes the proof. ◀

By analogous reasoning, we provide an alternating proof of [27] with respect to the variant of RCD defined in Remark 21.

▶ **Proposition 29.** *Equivalence check of reversible circuits without any ancillary random bit is* co-NP*-complete.*

Proof. Consider reversible circuits C_0, C_1 act on $n_w + n_0$ qubits, we observe that if C_0 and C_1 are not exactly equivalent, then $\exists s \in \{0,1\}^{n_w}, \langle s| \langle \bar{0}| C_0^\dagger C_1 |s\rangle |\bar{0}\rangle = 0$ since reversible circuits C_0 and C_1 are essentially bijections. Otherwise, it is evident that $\forall w, \langle w| \langle \bar{0}| C_0^\dagger C_1 |w\rangle |\bar{0}\rangle = 1$ provided C_0 and C_1 are exactly equivalent. Therefore, classical witness is optimal, and the remained proof follows from the proof of Proposition 28. ◀

B SetCSP$_{0,1/\text{poly}}$ is StoqMA$_1$-complete

We start from the definition of SetCSP with frustration:

▶ **Definition 30** (k-SetCSP$_{\epsilon_1,\epsilon_2}$, adapted from Section 4.1 in [3])**.** *Given a sequence of k-local set-constraints $C = (C_1, \cdots, C_m)$ on $\{0,1\}^n$, where k is a constant, n is the number of variables, and m is a polynomial of n. A set-constraint C_i acts on k distinct elements of $[n]$, and it consists of a collection $Y(C_i) = \{Y_1^{(i)}, \cdots, Y_{l_i}^{(i)}\}$of disjoint subsets $Y_j^{(i)} \subseteq \{0,1\}^k$. Promise that one of the following holds, decide whether*

- **Yes***: There exists a subset $S \subseteq \{0,1\}^n$ s.t.* $\text{set-unsat}(C, S) \leq \epsilon_1(n)$*;*
- **No***: For any subset $S \subseteq \{0,1\}^n$,* $\text{set-unsat}(C, S) \geq \epsilon_2(n)$*,*

where ϵ_1 and ϵ_2 are efficiently computable function and $\epsilon_2 - \epsilon_1 \geq 1/\text{poly}(n)$*.*

Now we briefly define a SetCSP instance C's frustration. We leave the formal definition in Proposition 34. The frustration of a set-constraint C regarding a subset S is $\text{set-unsat}(C, S) = \frac{1}{m}\sum_{i=1}^{m} \text{set-unsat}(C_i, S) = \frac{1}{m}\sum_{i=1}^{m}\left(\frac{|B_i(S)|}{|S|} + \frac{|L_i(S)|}{|S|}\right)$, where $B_i(S)$ is the set of bad strings of C_i, namely $\forall s \in B_i(S)$, $s|_{\text{supp}(C_i)} \notin \cup_{j=1}^{l_i} Y_j^{(i)}$; And $L_i(S)$ is the set of longing strings of the subset S regarding C_i.

We will prove Theorem 31 in the remainder of this section.

▶ **Theorem 31.** SetCSP$_{\text{negl},1/\text{poly}}$ *is* StoqMA$_{1-\text{negl}}$*-complete.*

B.1 $\mathrm{SetCSP}_{\mathbf{negl},\mathbf{1/poly}}$ is $\mathsf{StoqMA}(1-\mathbf{negl}, \mathbf{1/poly})$-hard

To prove Theorem 31, we will first show that $\mathrm{SetCSP}_{0,1/\mathrm{poly}}$ is StoqMA_1-hard.

▶ **Proposition 32** (SetCSP is hard for $\mathsf{StoqMA}(1-\mathrm{negl}, 1/\mathrm{poly})$). *For any super-polynomial $q(n)$ and polynomial $q_1(n)$, there exists a polynomial $q_2(n)$ such that $\mathrm{SetCSP}_{1/q(n),1/p_2(n)}$ is hard for $\mathsf{StoqMA}\left(1-1/q(n), 1/p_1(n)\right)$.*

Proof. The $\mathsf{StoqMA}\left(1-1/q(n), 1/p_1(n)\right)$-hardness proof is straightforwardly analogous to the circuit-to-Hamiltonian construction used in MA-hardness proof of SetCSP in [3]. The only difference is replacing $Y(C^{\mathrm{out}}) = \{\{00\},\{01\},\{11\}\}$ by $Y(C^{\mathrm{out}}) = \{\{00\},\{01\},\{10,11\}\}$ in Section 4.4.2, since the final measurement on the $(T+1)$-qubit is on the Hadamard basis instead of the computational basis. The rest of the proof follows from an inspection of Section 4.4 in [3]. ◀

Then Corollary 33 is an immediate corollary of Proposition 32 by substituting 0 for $1/q(n)$:

▶ **Corollary 33.** $\mathrm{SetCSP}_{0,1/\mathrm{poly}}$ *is* StoqMA_1*-hard.*

B.2 $\mathrm{SetCSP}_{a,b}$ is in $\mathsf{StoqMA}(1-a/2, 1-b/2)$

It now remains to show a StoqMA_1 containment of $\mathrm{SetCSP}_{0,1/\mathrm{poly}}$. We will complete the proof by mimicking the StoqMA containment of the stoquastic local Hamiltonian problem in Section 4 in [9]. The starting point is an alternating characterization of the frustration of a set-constraint C_i in a SetCSP instance C. The proof of Proposition 34 is deferred in the end of this section.

▶ **Proposition 34** (Local matrix associated with set-constraint). *For any k-local set-constraint $C_i (1 \leq i \leq m)$, given a subset $S \subseteq \{0,1\}^n$, the frustration*

$$\text{set-unsat}(C_i, S) = 1 - \sum_{j=1}^{|Y(C_i)|} \sum_{x,y \in Y_j^{(i)}} \tfrac{1}{|Y_j^{(i)}|} \langle S| \left(|x\rangle\langle y| \otimes I_{n-k}\right) |S\rangle.$$

Now we state the StoqMA containment of SetCSP, as Lemma 35.

▶ **Lemma 35.** *For any $0 \leq a < b \leq 1$, $\mathrm{SetCSP}_{a,b} \in \mathsf{StoqMA}\left(1-a/2, 1-b/2\right)$. Moreover, for a subset $S \subseteq \{0,1\}^n$ such that $S = \mathrm{argmin}_{S'}\, \text{set-unsat}(C, S')$, the subset state $|S\rangle$ is an optimal witness of the resulting* StoqMA *verifier.*

The proof of Lemma 35 tightly follows from Section 4 in [9]. We here provide a somewhat simplified proof using the SetCSP language by avoiding unnecessary normalization.

Proof of Lemma 35. Given a $\mathrm{SetCSP}_{a,b}$ instance $C = (C_1, \cdots, C_m)$. For each set-constraint $C_i (1 \leq i \leq m)$, we first construct a local Hermitian matrix M_i preserves the frustration, then construct a family of StoqMA verifiers for such a M_i. For any set-constraint C_i, we obtain a k-local matrix M_i by Proposition 34 such that for any subset $S \subseteq \{0,1\}^n$:

$$\text{set-unsat}(C_i, S) = 1 - \langle S| M_i \otimes I_{n-k} |S\rangle \text{ where } M_i = \sum_{j=1}^{|Y(C_i)|} \sum_{x,y \in Y_j^{(i)}} \frac{1}{|Y_j^{(i)}|} |x\rangle\langle y|. \quad (13)$$

Moreover, for a set $Y_j^{(i)}$ of strings associated with the set-constraint C_i, we further have

$$\begin{aligned}\sum_{x,y\in Y_j^{(i)}} |x\rangle\langle y| &= \sum_{x\in Y_j^{(i)}} |x\rangle\langle x| + \frac{1}{2}\sum_{x\neq y\in Y_j^{(i)}} \left(|x\rangle\langle y| + |y\rangle\langle x|\right)\\ &= \sum_{x\in Y_j^{(i)}} V_x |0\rangle\langle 0|^{\otimes k} V_x^\dagger + \frac{1}{2}\sum_{x\neq y\in Y_j^{(i)}} V_{x,y}\left(X\otimes |0\rangle\langle 0|^{\otimes k-1}\right)V_{x,y}^\dagger,\end{aligned} \tag{14}$$

where V_x is a depth-1 reversible circuit with X such that $\forall x, |x\rangle = U_x\left|0^k\right\rangle$, and $V_{x,y}$ is a $O(k)$-depth reversible circuit with CNOT and X such that $\forall x,y, U_{x,y}\left|0^k\right\rangle\left|10^{k-1}\right\rangle U_{x,y}^\dagger$.

Notice that the resulting local observables in Equation (14) are either $|0\rangle\langle 0|^{\otimes k}$ (i.e. a single-qubit computational-basis measurement) or $X\otimes|0\rangle\langle 0|^{\otimes k-1}$ (i.e. a single-qubit Hadamard-basis measurement). To construct a StoqMA verifier, we only allow local observables in form $X\otimes I^{\otimes O(k)}$. Namely, we are supposed to simulate a computational-basis measurement by ancillary qubits and a Hadamard-basis measurement, which is achieved by Proposition 36.

▶ **Proposition 36** (Adapted from Lemma 3 in [9]).

(1) *For any integer k, there exists an $O(k)$-depth reversible circuit W using k $|0\rangle$ ancillary qubits and a $|+\rangle$ ancillary qubits s.t.*

$$\forall|\psi\rangle, \langle\psi|\,|0\rangle\langle 0|^{\otimes k}\,|\psi\rangle = \langle\psi|\,\langle 0|^{\otimes k}\,\langle +|\,W^\dagger\left(X\otimes I^{\otimes 2k}\right)W\,|\psi\rangle\,|0\rangle^{\otimes k}\,|+\rangle\,.$$

(2) *For any integer k, there exists an $O(k)$-depth circuit V using $k-1$ $|0\rangle$ ancillary qubits s.t.*

$$\forall|\psi\rangle, \langle\psi|X\otimes|0\rangle\langle 0|^{\otimes k-1}\,|\psi\rangle = \langle\psi|\,\langle 0|^{\otimes k-1}\,W^\dagger\left(X\otimes I^{\otimes 2k-2}\right)W\,|\psi\rangle\,|0\rangle^{\otimes k-1}\,.$$

It is worthwhile to mention that the gadgets used in the proof (see Section A.4 in [9]) further provide proof of MA $\subseteq$ StoqMA that preserves both completeness and soundness parameters.

Let $\mathrm{Idx}\left(C_i\right)$ be the set of indices, and let $\alpha_{(j,x,y)}$ be the weight of an index (j,x,y),

$$\mathrm{Idx}\left(C_i\right) := \left\{(j,x,y) : 1\leq j\leq |Y_i(C)|, (x,y)\in\binom{Y_j^{(i)}}{2}\sqcup\left\{(x,x) : x\in Y_j^{(i)}\right\}\right\};$$

$$\alpha_{(j,x,y)} := \frac{1}{(1+\mathbb{I}(x\neq y))m|Y_j^{(i)}|}, \text{ where the indicator } \mathbb{I}(x\neq y)) = 1 \Leftrightarrow x\neq y.$$

Plugging Proposition 36 and Equation (14) into Equation (13), we have derived

$$1-\text{set-unsat}(C_i,S) = \sum_{l\in\mathrm{Idx}(C_i)} \alpha_l\langle S|\left(\langle 0|^{\otimes k}\,\langle +|\,U_k^\dagger\left(X\otimes I^{\otimes 2k}\right)U_k\,|0\rangle^{\otimes k}\,|+\rangle\right)\otimes I_{n-k}|S\rangle. \tag{15}$$

For a SetCSP instance $C=(C_1,\cdots,C_m)$, by Equation (15), by substituting $|+\rangle\langle +| = \frac{1}{2}(X+I)$ into Equation (15), we thus arrive at a conclusion that

$$\Pr\left[V_x \text{ accepts } |S\rangle\right] = \frac{1}{m}\sum_{i=1}^{m}\left(1-\frac{1}{2}\cdot\text{set-unsat}(C_i,S)\right) = 1-\frac{1}{2}\cdot\text{set-unsat}(C,S). \tag{16}$$

Note that the set of StoqMA verifiers V_x with the same number of input qubits and witness qubits is linear, namely a convex combination of l StoqMA verifiers $(V_1,p_1),\cdots,(V_l,p_l)$ can be implemented by additional $|+\rangle$ ancillary qubits and controlled $V_i (1\leq i\leq l)$. Therefore, by Equation (16), we conclude that $\forall a,b$, $\text{SetCSP}_{a,b}$ is in StoqMA $(1-a/2, 1-b/2)$. ◀

Finally, we achieve proof of Proposition 34:

Proof of Proposition 34. Given a k-local set-constraint C_i, the set of good strings $G_i = \sqcup_{1\leq j\leq |Y(C_i)|} Y_j^{(i)}$, and the set of bad strings $B_i = \{0,1\}^{|J(C_i)|} \setminus G_i$. Also, for any subset $S\{0,1\}^n$, the set of bad strings in S is $B_i(S)$. By direction calculation, notice that

$$\begin{aligned}\frac{|B_i(S)|}{|S|} &= \langle S| \left(\sum_{x\in B_i} |x\rangle\langle x| \otimes I_{n-k}\right)|S\rangle\\ \sum_{j=1}^{|Y(C_i)|}\frac{|L_j^{(i)}(S)|}{|S|} &= \langle S| \left(\sum_{x\in G_i} |x\rangle\langle x| \otimes I_{n-k}\right)|S\rangle - \sum_{j=1}^{|Y(C_i)|}\sum_{x,y\in Y_j^{(i)}} \frac{1}{|Y_j^{(i)}|}\langle S| \left(|x\rangle\langle y| \otimes I_{n-k}\right)|S\rangle.\end{aligned} \tag{17}$$

Plugging Equation (17) and $\{0,1\}^{|J(C_i)|} = B_i \sqcup G_i$ into set-unsat$(C_i,S) = \frac{|B_i(S)|}{|S|} + \sum_{j=1}^{|Y(C_i)|}\frac{|L_j^{(i)}(S)|}{|S|}$, we then finish the proof.

◀

Fault-Tolerant Syndrome Extraction and Cat State Preparation with Fewer Qubits

Prithviraj Prabhu
Department of Electrical and Computer Engineering, University of Southern California, Los Angeles, CA, USA

Ben W. Reichardt
Department of Electrical and Computer Engineering, University of Southern California, Los Angeles, CA, USA

Abstract

We reduce the extra qubits needed for two fault-tolerant quantum computing protocols: error correction, specifically syndrome bit measurement, and cat state preparation. For fault-tolerant syndrome extraction, we show an exponential reduction in qubit overhead over the previous best protocol. For a weight-w stabilizer, we demonstrate that stabilizer measurement tolerating one fault (distance-three) needs at most $\lceil \log_2 w \rceil + 1$ ancillas. If qubits reset quickly, four ancillas suffice. We also study the preparation of cat states, simple yet versatile entangled states. We prove that the overhead needed for distance-three fault tolerance is only logarithmic in the cat state size. These results could be useful both for near-term experiments with a few qubits, and for the general study of the asymptotic resource requirements of syndrome measurement and state preparation.

For a measured flag bits, there are 2^a possible flag patterns that can identify faults. Hence our results come from solving a combinatorial problem: the construction of maximal-length paths in the a-dimensional hypercube, corresponding to maximal-weight stabilizers or maximal-weight cat states.

2012 ACM Subject Classification Hardware → Quantum error correction and fault tolerance; Theory of computation → Quantum computation theory

Keywords and phrases Quantum error correction, fault tolerance, quantum state preparation, combinatorics

Digital Object Identifier 10.4230/LIPIcs.TQC.2021.5

Funding Research supported by Google and by MURI Grant FA9550-18-1-0161.

Acknowledgements The authors would like to thank Rui Chao, Sourav Kundu and Zhang Jiang for insightful conversations.

1 Introduction

A critical component of quantum error correction is syndrome measurement: a set of circuits used to pinpoint which qubits have errors. This process of error identification is itself susceptible to noise and may fail. To make this process robust, extra (ancilla) qubits can be used to identify damaging mid-circuit faults and mitigate the spread of errors. The objective of this paper is to reduce the overhead of ancilla qubits used in imparting this fault tolerance. In particular, we focus on optimizing the flag technique for distance-three fault tolerant stabilizer measurement. We also reduce qubit overhead in distance-three fault-tolerant cat state preparation. Cat states [11] have applications in many areas of quantum computing, including communication [9], information processing [12], and error correction [13, 14]. Besides practical applications, our results on cat state preparation are theoretically interesting since: i) we introduce the study of asymptotic estimates of qubit overhead for the fault-tolerant preparation of cat states of *arbitrary* size, and, ii) ideas developed for cat state preparation may provide clues for the fault-tolerant preparation of logical states of more complex codes.

16th Conference on the Theory of Quantum Computation, Communication and Cryptography (TQC 2021).
Editor: Min-Hsiu Hsieh; Article No. 5; pp. 5:1–5:13
Leibniz International Proceedings in Informatics
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

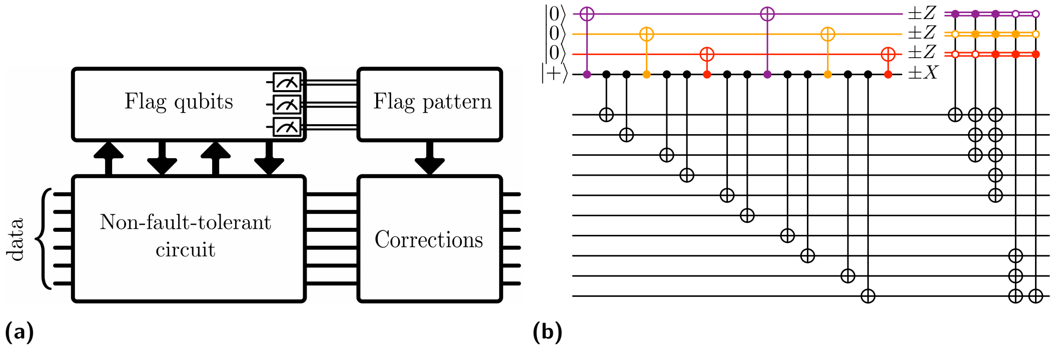


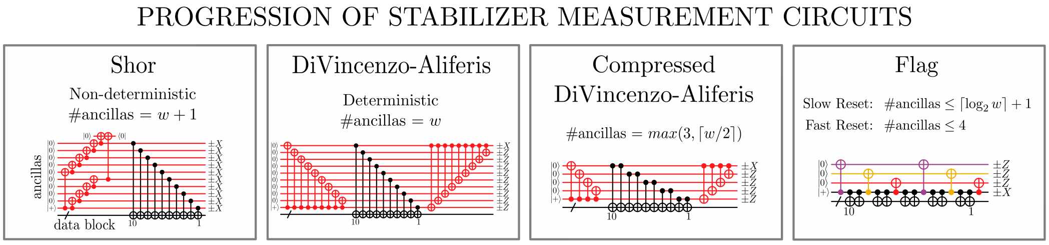
Figure 1 Functioning of a flag scheme. (a) Flag qubits interact with a non-fault-tolerant circuit to catch faults. Upon measurement, flag qubits yield a pattern of 1s and 0s. Based on the flag pattern, a correction is applied onto the data qubits. (b) Measurement of stabilizer $X^{\otimes 10}$ in the slow reset model, CSS fault-tolerant to distance three, using $a = 4$ ancilla qubits. Colored qubits and gates are used to impart distance-three fault tolerance.

We strive for low qubit overhead since quantum computers with limited qubits count resources preciously, and even minor improvements can free up extra qubits for other tasks. In topological codes where stabilizers are localized in space and low-weight, only a few flag qubits close to each stabilizer suffice to impart fault tolerance [16, 2, 3]. It has also been shown that with adaptive control and quickly resetting qubits, only four ancillas are needed for the universal fault-tolerant operation of some distance-three codes [4, 5]. In this paper, we present a general fault-tolerant protocol that works for a stabilizer of any size. If qubits are connected well enough, we show that only logarithmic overhead is required for fault-tolerant stabilizer measurement, an exponential space improvement over the previous linear overhead.

The general model of flag-based fault-tolerance is displayed in Figure 1a. Here, a set of flag ancilla qubits monitor operations in a non-fault-tolerant circuit and when measured at the end, produce flag patterns which uniquely identify mid-circuit faults. Based on the observed flag pattern, a correction is applied to the data to minimize the spread of errors. As an example, Figure 1b measures a stabilizer on 10 data qubits while tolerating one fault. The three colored qubits are the flag ancillas and the measured flag patterns each imply different corrections. Also note that the sequence of flag patterns $(100, 110, 111, 011, 001)$ is a path on the hypercube and also corresponds to the order of the flag CNOTs (edge between 100 and 110 implies a CNOT is applied onto flag qubit 2).

In this paper, we restrict discussion to the measurement of individual stabilizers of a quantum code, as in Shor-style fault-tolerant stabilizer measurement [13]. We do not focus on other methods which measure multiple stabilizers in parallel. Figure 2 displays improvements made over the years to Shor's method. Note that Shor's method can tolerate any number of faults by increasing the fault tolerance of the ancillary cat state preparation. The subsequent schemes forgo this property and are only fault-tolerant to distance three. DiVincenzo and Aliferis first make the circuit deterministic by removing the need for cat state verification [7]. This ensures that a circuit designer need not wait for a fault-tolerantly prepared cat state before measuring the stabilizer. Subsequent improvements were made in [15], [16] and [5] to reduce ancilla count by coupling each ancilla qubit to two data qubits instead of one.

With our flag method, the ancilla cat state is prepared and unprepared while collecting the stabilizer. As in Figure 1b, an X fault occurring anywhere on the $|+\rangle$ qubit may spread into the data, but will also leave its imprint on the flags. This is then measured out as a flag pattern. Due to the particular chosen arrangement of the flag CNOTs, any fault that can


PROGRESSION OF STABILIZER MEASUREMENT CIRCUITS
Shor
Non-deterministic
#ancillas = $w+1$
ancillas
data block
10
1
DiVincenzo-Aliferis
Deterministic
#ancillas = w
10
1
Compressed DiVincenzo-Aliferis
#ancillas = $max(3, \lceil w/2 \rceil)$
10
1
Flag
Slow Reset: #ancillas $\leq \lceil \log_2 w \rceil + 1$
Fast Reset: #ancillas ≤ 4
$|0\rangle$
$|0\rangle$
$|0\rangle$
$|+\rangle$
$\pm Z$
$\pm Z$
$\pm Z$
$\pm X$
10
1

(a) [13]. **(b)** [7]. **(c)** [15, 16, 5]. **(d)**

Figure 2 Historical progression of stabilizer measurement circuits. A weight-10 X stabilizer measurement circuit is provided as an example. CNOTs in black have targets on the 10 data qubits, collectively represented by the black wire. In (b)(c)(d), fault-tolerance is only guaranteed to distance-three and Pauli corrections (or Pauli frame updates) are applied to the data based on the Z basis measurements. (a) Shor's method uses $w+1$ ancillas and requires a fault-tolerantly prepared cat state. (b)(c) The following two methods use unverified cat states with subsequent error decoding. Non-deterministic cat state verification is replaced with a deterministic circuit, allowing for uninterrupted circuit operation. (d) The flag method prepares and unprepares an ancilla cat state while collecting the stabilizer. Exponentially more flag configurations can thus be accessed for fault diagnosis.

Table 1 Distance-3 cat state preparation: Weight-w cat states can be prepared fault-tolerantly to distance-3 with m measurements of ancilla qubits. Slow reset requires m ancilla qubits whereas with fast reset, only one ancilla qubit is required.

Type	Bounds
Deterministic error correction Theorem 5	$w \leq 3\,(2^m - 2m + 2)$
Adaptive error correction Theorem 6	$w \leq 3\,(2^m - 2m + 3)$

spread to a high-weight data error triggers one of the five shown flag patterns. Each flag pattern then applies a unique correction that ensures that there is at most one data qubit in error. This satisfies the condition for fault tolerance, which states that k faults in a circuit should cause no more than k qubits to have errors.

For the distance-three fault-tolerant measurement of a weight-w stabilizer, we propose two methods based on the speed of qubit reset. With fast qubit reset, Theorem 3, only three flag ancillas are required in total, but each flag needs to be measured once per four data qubits. If more flags are used in parallel, the number of accessible flag patterns grows exponentially and the number of measurements per ancilla converges to one. This is the regime of slow qubit reset, Theorem 4, which uses at most $\lceil \log_2 w \rceil$ flag ancillas measured only at the end.

Table 1 contains bounds on the ancilla overhead for preparing weight-w cat states fault-tolerantly to distance-three. If the flag qubits can reset quickly, Theorem 5 states that only one flag qubit is required and it needs to be reset and measured m times. Since the flag qubits operate independently, it is also possible to use m flag qubits, with each one being measured once. We further show how to use an adaptive circuit in Theorem 6 to marginally increase the number of flag patterns in use.

The rest of this paper is divided into three sections. Section 2 details the construction of the two paths on the hypercube that we use as flag sequences. Section 3 describes how to use these sequences for distance-three fault-tolerant syndrome measurement, and Section 4 deals with cat state preparation.

2 Flag sequences

A flag pattern or flag configuration is a string of 1s and 0s that arises from measuring out the flags. If a flag ancillas are used, then the a-bit flag configuration labels a vertex of the a-dimensional hypercube. We show how to construct two paths on the hypercube to produce maximal-length sequences of flag configurations. Since they are paths, only one bit is flipped between subsequent flag configurations. This bit flip corresponds to the application of a flag CNOT from the syndrome ancilla to the flag qubit indexed by the flipped bit, thus providing a blueprint to construct the fault-tolerant circuit.

The first type of sequence just requires a maximal-length traversal of the a-dimensional hypercube. A simple choice for this is the Gray code [10, 8].

▶ **Lemma 1.** *For $a \geq 1$, the Gray code creates a length-2^a Hamiltonian path in the a-dimensional hypercube.*

Proof. We provide a quick construction of the sequence. For $a = 1$, use the sequence $0, 1$. For $a > 1$, construct the sequence inductively. First, run the sequence for $a - 1$ with a 0 prepended, then run it backwards with a 1 prepended. ◀

For example, for $a = 2$, the sequence is $00, 01, 11, 10$. For $a = 3$, the sequence is $000, 001, 011, 010, 110, 111, 101, 100$.

In this paper we use a piece-wise definition of fault tolerance. Fault tolerance to distance-d implies that for all $k \leq t = \lfloor \frac{d-1}{2} \rfloor$, correlated errors of weight-k occur with k-th order probability. For distance-three CSS fault-tolerant syndrome bit measurement, any single fault should result in a data error with X and Z components having weight zero or one.

In order to ensure that the circuit is distance-three fault-tolerant, we need to ensure that a measurement fault does not trigger corrections of weight greater than one. Hence the second maximal-length sequence requires that there are no weight-one strings except at the start and end. As shown in Figure 1b, we may assign weight-one corrections to these two configurations, but for all others there exist multi-qubit corrections.

▶ **Lemma 2.** *For $a \geq 2$, in the a-dimensional hypercube $\{0, 1\}^a$ there exists a path $v_1 = 10^{a-1}, \ldots, v_n = 0^{a-1}1$ such that all intermediate vertices $v_2, \ldots, v_{n-1}$ have weight at least two, and each vertex appears at most once; with length $n = 2^a - 2a + 3$.*

Proof. Figure 3 illustrates the inductive construction of maximal-length flag sequences satisfying the above constraints. With $a = m - 1$ flag qubits, the sequence has length $2^a - 2a + 3$. The a-flag sequence is constructed by first running the previous flag sequence, on $a - 1$ flags, up to the second-to-last element (which for $a \geq 4$ is $\chi_{\{2,a-1\}}$[1]), and with 0 appended at the end. Then run the sequence backward, except with 1 appended at the end, and with the 2 and $a - 1$ coordinates swapped (the red and blue rows in the figure). Finally, finish the sequence from $\chi_{\{1,a\}}$ by walking through $\chi_{\{3,a\}}, \chi_{\{4,a\}}, \ldots, \chi_{\{a-2,a\}}, \chi_{\{2,a\}}$, with the appropriate weight-three sequences (shown in gray) interposed.

To ensure that no vertex is visited more than once, one need only check that the last $2a-5$ sequences are distinct from those that came before. For this, one can track by induction the $2a - 3$ hypercube vertices that are not visited by each walk: 0^a, the $a - 2$ weight-one strings $\chi_2, \ldots, \chi_{a-1}$, and the $a - 2$ weight-two strings $\chi_{\{1,3\}}, \chi_{\{3,4\}}, \chi_{\{4,5\}}, \ldots, \chi_{\{a-1,a\}}$. ◀

[1] $\chi_{\{x,y\}}$ implies bits at positions x and y are set to 1.

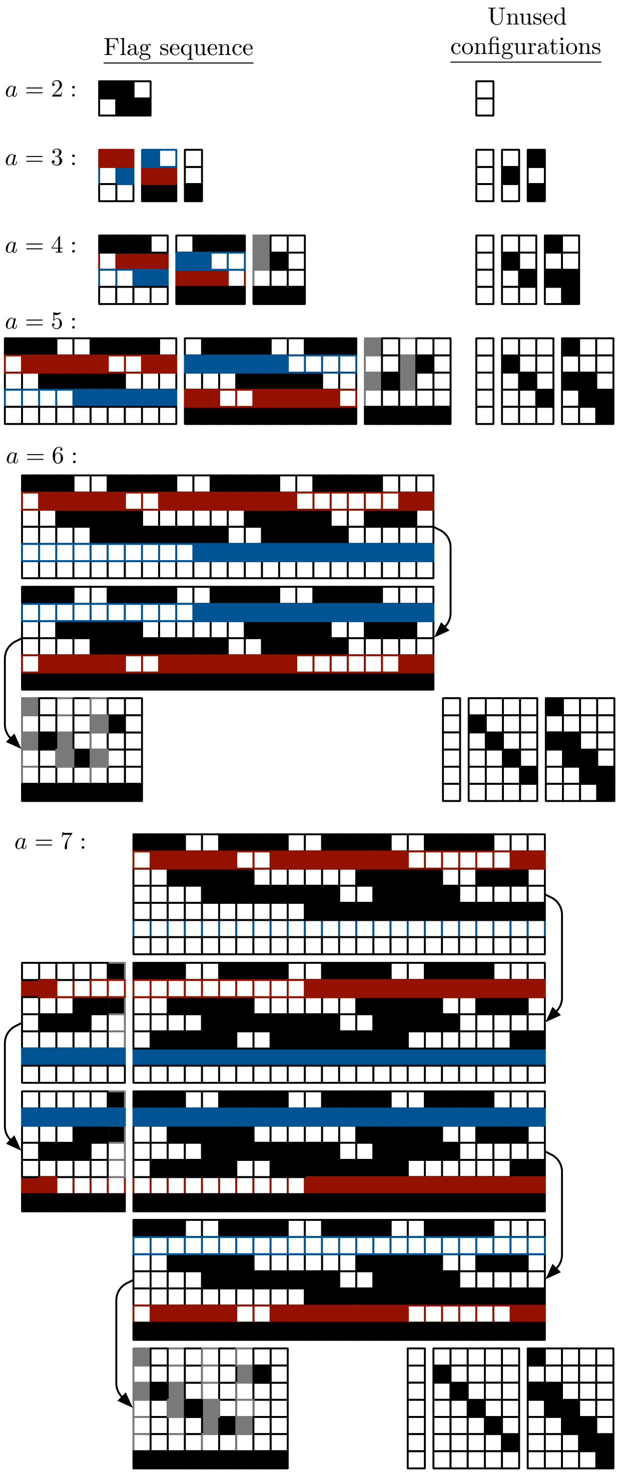


Figure 3 Flag sequences for distance-three fault-tolerant syndrome bit measurement, using a flag qubits, each measured once (the slow reset model). These sequences are walks through the a-dimensional hypercube, from 10^{a-1} to $0^{a-1}1$; passing through each vertex at most once and no other weight-one vertices. Flag configurations are stacked vertically and ordered initially left to right, with solid and empty squares representing 1 and 0, respectively, e.g., ▚ represents $10, 11, 01$.

3 Distance-three syndrome measurement

In this section, we outline two protocols for distance-three CSS fault-tolerant syndrome measurement. They differ based on the speed of qubit measurement and reset.

For $w \in \{4, 5, 6\}$, flag-fault-tolerant circuits are constructed the same way regardless of qubit reset speed. We show in Figure 4a that for $w = 6$, only two flag ancillas are required. Lower-weight stabilizers can be measured by removing data CNOTs and making appropriate changes to the Pauli corrections. For $7 \leq w \leq 10$, the different methods of construction yield the same circuits. It is only for $w > 10$ that the effects of qubit reset speed are pronounced.

3.1 Fast reset

▶ **Theorem 3.** *If qubits can be measured and reset quickly, then for any w, four ancilla qubits are sufficient to measure the syndrome of $X^{\otimes w}$, CSS fault-tolerantly to distance three. Moreover, the number of measurements needed is $\lceil \frac{w+2}{4} \rceil + 1$.*

Proof. For $w \in \{4, 5, 6\}$, the circuit using two flag ancillas is shown in Figure 4a. It runs through a sequence of three flag configurations and a multi-qubit correction is only applied for the flag configuration 11. For $w > 6$, the general construction is shown in Figure 5. Each repetition of the highlighted region adds the X parity of four more data qubits, while measuring and quickly reinitializing one flag qubit. In terms of the number of measurements m, the construction achieves up to $w = 4(m-1) - 2$. It is fault tolerant because X faults on the control wire cause flag configurations of alternating weights two or three, that localize the fault to three possible consecutive locations along the control wire: before, between or after two CNOT gates. The appropriate correction is for a fault between the CNOT gates. ◀

Theorem 3 may be optimal; it does not appear to be possible to use fewer than three flag qubits. With just one flag qubit, one can detect that an error has occurred, but not where. As illustrated in Figure 6, either the control wire is unprotected at some point or for $w \geq 4$ there is no consistent correction rule.

By a similar argument, two flag qubits are not enough. Any correction based on a single flag can have weight at most one, since the flag measurement itself could be faulty. However, if at some point in the middle the control wire is protected by just a single flag, a weight-one correction will not suffice. On the other hand, if both flags are used to protect the control wire across the entire sequence of CNOT gates, we are unable to locate faults well enough to correct them.

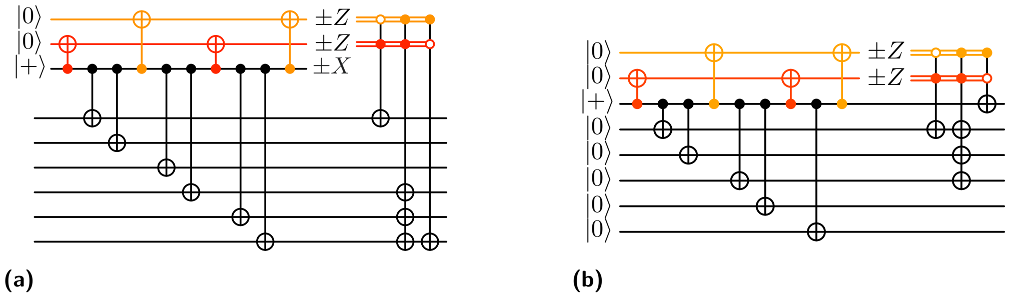


Figure 4 (a) Circuit to measure an $X^{\otimes 6}$ stabilizer, CSS fault-tolerant to distance three. (b) Circuit to prepare a six-qubit cat state, fault-tolerant to distance three.

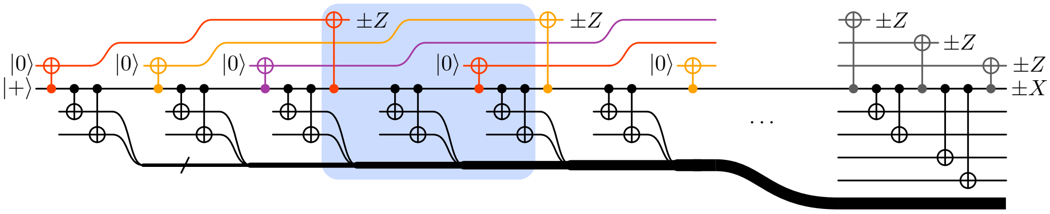


Figure 5 Distance-three fault-tolerant syndrome measurement only requires three flag qubits. The highlighted region can be repeated to fit the weight of the stabilizer being measured.

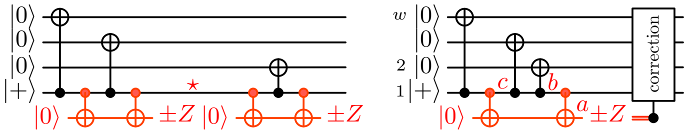


Figure 6 Distance-three error correction is not possible with only one flag qubit. Either (left) the control wire is unprotected at some point $\star$, from which an X fault can propagate to an error of weight at least two; or (right) faults at a, b, c, causing respective errors I, X_1, X_w have no consistent correction.

We would like to point out that this construction can also be used to prepare a w-qubit cat state fault tolerantly to distance three. The conversion to this circuit follows three steps: Remove one data qubit. Initialize the data qubits as $|0\rangle$. Remove the syndrome ancilla measurement, so as to retain the qubit in the support of the stabilizer. An example of this conversion is shown for $w = 6$ in Figure 4b. In Section 4, this method will be subsumed by a better protocol that uses just one ancilla qubit.

3.2 Slow reset

▶ **Theorem 4.** *The syndrome of $X^{\otimes w}$ can be measured CSS fault-tolerantly to distance three using $m \geq 3$ measurements, provided that*

$$w \leq 2\left(2^{m-1} - 2(m-1) + 3\right).$$

Proof. Two examples are shown in Figure 4a, for $w = 6$, and Figure 1b, for $w = 10$. As in these figures, in general we collect the syndrome two qubits at a time into a syndrome qubit that is initialized as $|+\rangle$. Between each of these pairs of CNOT gates, a CNOT is applied from the syndrome qubit into one of $m - 1$ flag qubits. This leads to a sequence of flag configurations, e.g., $100, 110, 111, 011, 001$ for the $w = 10$ example. Based on the observed flag configuration, a correction is applied as if an X fault had occurred between the corresponding pair of flag CNOT gates.

Observe that the flag sequence changes one bit at a time; it can be thought of as a path on the hypercube. It begins and ends with weight-one configurations, but otherwise the configurations all have weight at least two. This is important for distance-three fault tolerance because a fault could affect the flags, and only the first and last data corrections have weight one. Also, the flag configurations along the sequence are distinct, so each is associated with only one correction. The theorem then just follows from the flag sequence construction in Lemma 2. ◀

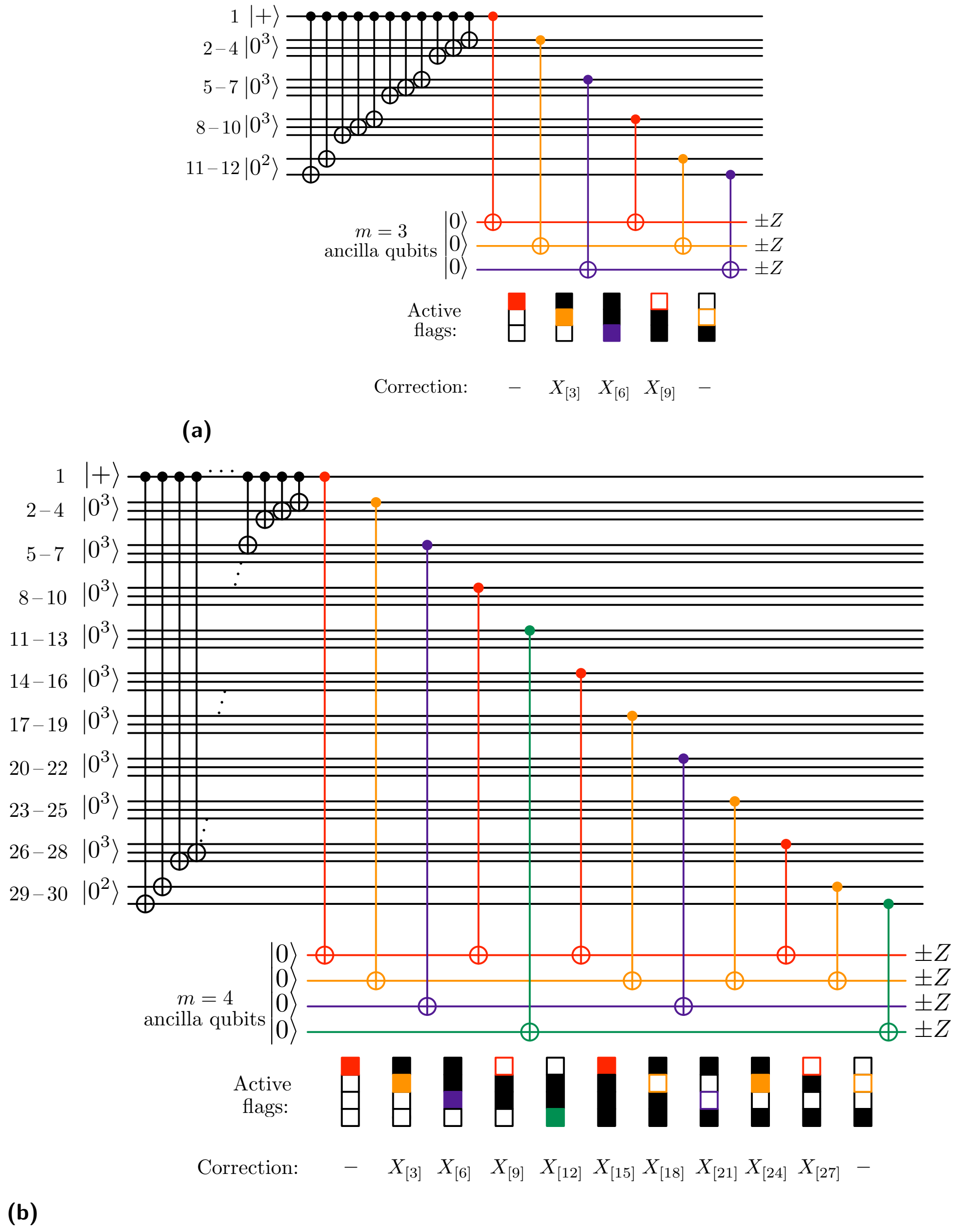


Figure 7 Distance-three fault-tolerant cat state preparation circuits. Note that, with fast reset, only one ancilla qubit is required.

The construction of Lemma 2 gives flag sequences of maximal length, $2^a - 2a + 3$. Indeed, this follows since the number of vertices with odd weight greater than one is $2^{a-1} - a$, and vertices must alternate between odd and even weight.

Note that the approach of Theorem 4, with slow reset, is different from the fast reset case of Theorem 3, in that a flag qubit is active and able to detect faults in more than one region of the circuit.

4 Distance-three cat state preparation

Next we turn to the question of distance-three fault-tolerant preparation of cat states. For preparing a two- or three-qubit cat state, any preparation circuit is automatically fault-tolerant, because every error has weight zero or one. For example, on three qubits $XXI \sim IIX$, since XXX is a stabilizer. Fault tolerance becomes interesting for preparing cat states on $w \geq 4$ qubits.

The ideas of Theorems 3 and 4 can also be applied to cat state preparation. For example, just as in Figure 4 a circuit for measuring $X^{\otimes 6}$ with three ancilla qubits corresponds to a circuit to prepare a six-qubit cat state with two ancillas, similarly adapting the construction of Theorem 4 allows preparing a $2(2^a - 2a + 3)$ qubit cat state with a ancilla qubits each measured once. However, we can do better.

▶ **Theorem 5.** *For $m \geq 2$, one ancilla qubit, measured m times, is sufficient to prepare a cat state on w qubits fault-tolerantly to distance three, for*

$$w \leq 3\left(2^m - 2m + 2\right).$$

Proof. Figure 7 illustrates our construction for the cases $m = 3$ and $m = 4$. In general, we prepare a w-qubit cat state using CNOT gates from the first qubit, so that the possible X errors from a single fault are $\mathbf{1}, X_1, X_{[2]}, X_{[3]}, \ldots$. (Here we are using the notation $[m] = \{1, 2, \ldots, m\}$ and $X_S = \prod_{j \in S} X_j$.) We then compute parities of subsets of the qubits into the ancillas, following the flag sequence from Lemma 2 and Figure 3. Although for clarity Figure 7 shows the m parity checks being made in parallel, they can also be made sequentially with just one ancilla qubit.

With the given correction rules, errors due to single faults are corrected up to possibly a weight-one remainder. (For example, in Figure 7a, errors $X_{[5]}$, $X_{[6]}$ and $X_{[7]}$ all result in the parity checks 111, for which the correction $X_{[6]}$ is applied.) The circuit also tolerates faults within the parity-check sub-circuit, because a single fault here can flip at most one parity, and no correction is applied for the weight-one configurations. ◀

By this method, the cat state is prepared in depth $w - 1$. The depth of the parity check circuit, however, increases exponentially as 2^{m-2} for $m \geq 3$ if we consider slow reset ($a = m$). This is evident from the flag sequences in Figure 3 as the maximum number of times any flag bit is switched. The total depth of the circuit is then $(w - 1) + 2^{m-2}$.

Note that the construction from Theorem 5 does not help for syndrome measurement, because the parity checks would in general become entangled with the data.

We can do slightly better if we allow an *adaptive* circuit, in which the parity checks are chosen based on the outcome of a flag qubit measurement. For example, Figure 8 gives a circuit to prepare a 15-qubit cat state using $m = 3$ measurements. Here, the result of measuring the red ancilla determines how the other two ancillas are used.

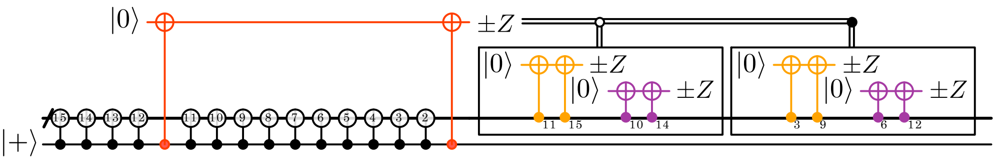


Figure 8 Circuit to prepare a 15-qubit cat state by adaptive error correction, fault-tolerant to distance three. Labels on the thick black wire indicate which data qubit in the block is being addressed as the control or target of the CNOT. If a fault occurs while preparing the cat state on the $|+\rangle$ qubit, it is partially localized by the red flag ancilla. The measurement result of this flag then determines a set of parity checks to completely localize a possible fault. After all the ancilla qubits have been measured, corrections are applied based on Table 2 and Table 3.

Table 2 Parity checks and correction rules when the red flag ancilla in Figure 8 is measured as 1.

$3 \oplus 9$	$6 \oplus 12$	Possible errors	Correction
0	0	$\mathbf{1}, X_1, X_{[2]}$	X_1
1	0	$X_{[3]}, X_{[4]}, X_{[5]}$	$X_{[4]}$
1	1	$X_{[6]}, X_{[7]}, X_{[8]}$	$X_{[7]}$
0	1	$X_{[9]}, X_{[10]}, X_{[11]}$	$X_{[10]}$

▶ **Theorem 6.** *Using an adaptive circuit, for $m \geq 2$, one ancilla qubit, measured m times, can be used to prepare a cat state on w qubits fault-tolerantly to distance three, for*

$$w \leq 3\left(2^m - 2m + 3\right).$$

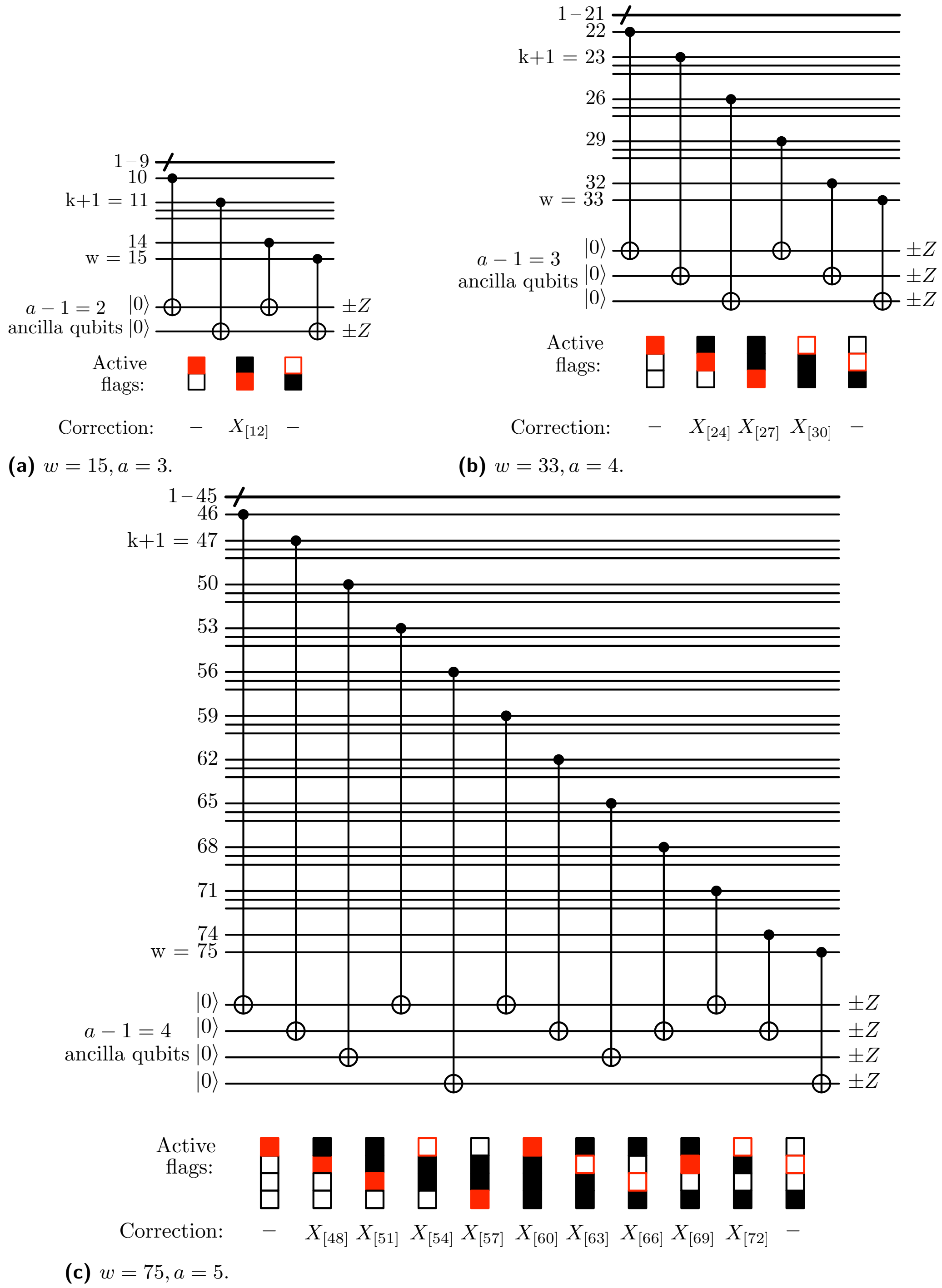
Proof. Our construction will follow the same basic structure as the circuit in Figure 8. Prepare the w data qubits as $|{+}0^{w-1}\rangle$, then apply $\mathrm{CNOT}_{1,w}, \mathrm{CNOT}_{1,w-1}, \ldots, \mathrm{CNOT}_{1,2}$ to get a cat state. Let $k = 3 \cdot 2^{m-1} - 2$. Just before $\mathrm{CNOT}_{1,k+1}$ and just after $\mathrm{CNOT}_{1,2}$, apply CNOTs into the first ancilla qubit, the red qubit in Figure 8, and measure it.

The remainder of the circuit depends on the measurement result. If it is 1, then a fault has been detected. The error on the cat state can be one of

$$\mathbf{1}, X_1, X_{[2]}, \quad X_{[3]}, X_{[4]}, X_{[5]}, \quad \ldots, \quad X_{[k-1]}, X_{[k]}, X_{[k+1]}.$$

The correction procedure needs to determine in which of the above $1 + \frac{k-1}{3}$ groups of three the error lies; then for any error in $\{X_{[3j]}, X_{[3j+1]}, X_{[3j+2]}\}$ the correction $X_{[3j+1]}$ works. Perhaps the easiest way to locate the error is by binary search using the Gray code in Lemma 1, e.g., by computing parities between qubits $3j$ for $j \in \{1, 2, \ldots, 1 + \frac{k-1}{3}\}$. Since the measurement of the red ancilla could have been incorrect, it is important that the all-0s outcome of the binary search correspond to the $\mathbf{1}, X_1, X_{[2]}$ error triple, as in Table 2. Using $m - 1$ measurements, we can search 2^{m-1} possibilities, which indeed is $1 + \frac{k-1}{3}$. (The search circuit can also be made nonadaptive, as in Figure 8.)

Next consider the case that the first measurement result is 0, so no fault has been detected. The error on the cat state can be one of $X_{[k+1]}, X_{[k+2]}, \ldots, X_{[w]} \sim \mathbf{1}$. We again use the remaining $m - 1$ ancilla qubits to measure parities of subsets of cat state qubits. Since there is no guarantee of a fault having occurred yet, we use flag sequences from Lemma 2, where the length of the weight-at-least-two flag sequence is $J = (2^{m-1} - 2(m-1) + 1)$. The parity checks are now done between qubits $\{k, k + 1 + 3j, k + 2 + 3J\}$ for $j \in \{0, 1, \ldots, J\}$, as shown in Figure 9 and Table 3. We do not allow weight-one flag configurations to be able to correct any errors since they can be triggered by a measurement fault on any one of the data qubits involved in the parity check.



(a) $w = 15, a = 3$.

(b) $w = 33, a = 4$.

(c) $w = 75, a = 5$.

Figure 9 If the red ancilla flag in Figure 8 is not triggered, these circuits are used to find and correct a possible error. The flag sequences (from Figure 3) and corresponding corrections are listed at the bottom. Note that these sequences are nonadaptive, and can be used either with a ancilla qubits, in a slow reset model, or with just one ancilla qubit in a fast reset model (because many of the CNOT gates commute).

Table 3 Parity checks and correction rules when the red flag ancilla is measured as 0.

$11 \oplus 15$	$10 \oplus 14$	Possible errors	Correction
0	1	Flag/data qubit error	None
1	1	$X_{[11]}, X_{[12]}, X_{[13]}$	$X_{[12]}$
1	0	$X_{[14]}$ or flag/data qubit error	None
0	0	$\mathbf{1}$	None

Consolidating, we are allowed up to $3J+1$ CNOTs before the red ancilla is initialized, and up to k CNOTs in the monitored region of the red ancilla. In total we can create a cat state on up to

$$w \leq 3J + k + 2 = 3\left(2^m - 2m + 3\right)$$

qubits, with m total measurements. ◀

We also tested protocols where multiple flags are used for the initial partial localization of a fault (in place of the red flag qubit). We found no improvement to our bounds on ancilla overhead. It appears that ancillas are better used in the parity checks than for partial fault localization.

5 Conclusion

In this paper, we optimize the overhead of distance-three fault tolerance for stabilizer measurement and cat state preparation. If the circuit on w qubits must tolerate one fault, we show that only $\sim \log w$ extra qubits are required. We detail the construction of a maximal-length path on the hypercube and show that it can be used to greatly increase the ability to catch and distinguish faults.

We describe two circuits for stabilizer measurement based on the speed of ancilla qubit reset. With slow reset, a weight-w stabilizer can be measured fault-tolerantly to distance-three using only $\lceil \log_2 w \rceil$ flag qubits for fault tolerance. With fast reset, only three flag qubits are required, but the number of times they are measured and reset grows as $\sim \frac{w}{4}$.

In our circuits for fault-tolerant cat state preparation we check for errors after the cat state is non-fault-tolerantly prepared. We show, using a deterministic and an adaptive circuit, that the overhead for fault tolerance can be as low as logarithmic in the size of the cat state. In fact, only one flag qubit suffices, as long as it can reset quickly.

There are numerous avenues for further improvements. The circuits detailed in this paper are only fault-tolerant to distance-three. Using more complex designs, flag-based fault tolerance can be used to effect fault tolerance to arbitrary distance [1, 6]. It may be interesting to try to develop higher-distance circuits for stabilizer measurement with logarithmic overhead.

From the perspective of stabilizer algebra, a cat state is a CSS ancilla state. A future avenue of research might look to extend these flag techniques to the fault-tolerant preparation of general CSS ancilla states.

In order to execute the circuits in this paper, one qubit needs to be connected to all the other qubits used. This does not bode well for architectures with limited connectivity. But by mixing flag and transversal gate concepts for fault tolerance, it is possible to construct stabilizer measurement circuits that can measure arbitrarily large stabilizers using only local interactions, fault-tolerantly. This can be especially useful in technologies such as superconducting qubits, where qubits only talk to neighbors on a 2-D lattice.

References

1 Christopher Chamberland and Michael E. Beverland. Flag fault-tolerant error correction with arbitrary distance codes. *Quantum*, 2:53, 2018. doi:10.22331/q-2018-02-08-53.

2 Christopher Chamberland, Aleksander Kubica, Theodore J. Yoder, and Guanyu Zhu. Triangular color codes on trivalent graphs with flag qubits. *New Journal of Physics*, 22(2):023019, 2020. doi:10.1088/1367-2630/ab68fd.

3 Christopher Chamberland, Guanyu Zhu, Theodore J. Yoder, Jared B. Hertzberg, and Andrew W. Cross. Topological and subsystem codes on low-degree graphs with flag qubits. *Phys. Rev. X*, 10:011022, 2020. doi:10.1103/PhysRevX.10.011022.

4 Rui Chao and Ben W. Reichardt. Error correction with only two extra qubits. *Phys. Rev. Lett.*, 121:050502, 2018. doi:10.1103/PhysRevLett.121.050502.

5 Rui Chao and Ben W. Reichardt. Fault-tolerant quantum computation with few qubits. *npj Quantum Information*, 4(1):42, 2018. doi:10.1038/s41534-018-0085-z.

6 Rui Chao and Ben W. Reichardt. Flag fault-tolerant error correction for any stabilizer code. *PRX Quantum*, 1:010302, 2020. doi:10.1103/PRXQuantum.1.010302.

7 David P. DiVincenzo and Panos Aliferis. Effective fault-tolerant quantum computation with slow measurements. *Phys. Rev. Lett.*, 98:220501, 2007. doi:10.1103/PhysRevLett.98.020501.

8 M. Gardner. The Binary Gray Code. In *Knotted Doughuts and other Mathematical Entertainments*, pages 22–39. W. H. Freeman and Company, New York, 1986.

9 Nicolas Gisin and Rob Thew. Quantum communication. *Nature Photonics*, 1(3):165–171, 2007. doi:10.1038/nphoton.2007.22.

10 F. Gray. Pulse code communication, 1953. US Patent 2,632,058. URL: http://www.google.com/patents/US2632058.

11 Daniel M. Greenberger, Michael A. Horne, and Anton Zeilinger. *Going Beyond Bell's Theorem*, pages 69–72. Springer Netherlands, Dordrecht, 1989. doi:10.1007/978-94-017-0849-4_10.

12 Jian-Wei Pan, Zeng-Bing Chen, Chao-Yang Lu, Harald Weinfurter, Anton Zeilinger, and Marek Żukowski. Multiphoton entanglement and interferometry. *Rev. Mod. Phys.*, 84:777–838, 2012. doi:10.1103/RevModPhys.84.777.

13 Peter W. Shor. Fault-tolerant quantum computation. In *Proc. 37th Symp. on Foundations of Computer Science (FOCS)*, page 96, 1996. doi:10.1109/SFCS.1996.548464.

14 Andrew M. Steane. Active stabilization, quantum computation, and quantum state synthesis. *Phys. Rev. Lett.*, 78(11):2252–2255, 1997. doi:10.1103/PhysRevLett.78.2252.

15 Ashley M. Stephens. Efficient fault-tolerant decoding of topological color codes, 2014. arXiv:1402.3037.

16 Theodore J. Yoder and Isaac H. Kim. The surface code with a twist. *Quantum*, 1:2, 2017. doi:10.22331/q-2017-04-25-2.

A Note About Claw Function with a Small Range

Andris Ambainis
Center for Quantum Computer Science, Faculty of Computing, University of Latvia, Riga, Latvia

Kaspars Balodis
Center for Quantum Computer Science, Faculty of Computing, University of Latvia, Riga, Latvia

Jānis Iraids
Center for Quantum Computer Science, Faculty of Computing, University of Latvia, Riga, Latvia

Abstract

In the claw detection problem we are given two functions $f : D \to R$ and $g : D \to R$ ($|D| = n$, $|R| = k$), and we have to determine if there is exist $x, y \in D$ such that $f(x) = g(y)$. We show that the quantum query complexity of this problem is between $\Omega\left(n^{1/2}k^{1/6}\right)$ and $O\left(n^{1/2+\varepsilon}k^{1/4}\right)$ when $2 \le k < n$.

2012 ACM Subject Classification Theory of computation → Quantum query complexity

Keywords and phrases collision, claw, quantum query complexity

Digital Object Identifier 10.4230/LIPIcs.TQC.2021.6

Funding Supported by the project "Quantum algorithms: from complexity theory to experiment" funded under ERDF programme 1.1.1.5.

1 Introduction

In this note we study the Claw problem in which given two discrete functions $f : D \to R$ and $g : D \to R$ ($|D| = n$, $|R| = k$) we have to determine if there is a collision, i.e., inputs $x, y \in D$ such that $f(x) = g(y)$. In contrast to the Element-Distinctness problem, where the input is a single function $f : D \to R$ and we have to determine if f is injective, Claw is non-trivial even when $k < n$. This is the setting we focus on.

Both Claw and Element-Distinctness have wide applications as useful subroutines in more complex algorithms [5, 12] and as a means of lower bounding complexity [10, 1].

Claw and Element-Distinctness were first tackled by Buhrman et al. in 2000 [8] where they gave an $O\left(n^{3/4}\right)$ algorithm and $\Omega\left(n^{1/2}\right)$ lower bound. In 2003 Ambainis, introducing a novel technique of quantum walks, improved the upper bound to $O\left(n^{2/3}\right)$ in the query model [4]. It was soon realized that a similar approach works for Claw [9, 13, 15]. Meanwhile Aaronson and Shi showed a lower bound $\Omega\left(n^{2/3}\right)$ that holds if the range $k = \Omega\left(n^2\right)$ [2]. Eventually Ambainis showed that the $\Omega\left(n^{2/3}\right)$ bound holds even if $k = n$ [3]. The same lower bound has since been reproved using the adversary method [14]. Until now, only the $\Omega\left(n^{1/2}\right)$ bound based on reduction of searching was known for Claw with $k = o(n)$ [8].

We consider quantum query complexity of Claw where the input functions are given as a list of their values in black box. Let $Q(f)$ denote the bounded error quantum query complexity of f. For a short overview of black box model refer to Buhrman and de Wolf's survey [7]. Let $[n]$ denote $\{1, 2, \ldots, n\}$. Let $\textsc{Claw}_{n\to k} : [k]^{2n} \to \{0, 1\}$ be defined as

$$\textsc{Claw}_{n\to k}(x_1, \ldots, x_n, y_1, \ldots, y_n) = \begin{cases} 1, & \text{if } \exists i, j\; x_i = y_j \\ 0, & \text{otherwise} \end{cases}.$$

Our contribution is a quantum algorithm for $\textsc{Claw}_{n\to k}$ with quantum query complexity $Q(\textsc{Claw}_{n\to k}) = O\left(n^{1/2+\varepsilon}k^{1/4}\right)$ and a lower bound $Q(\textsc{Claw}_{n\to k}) = \Omega\left(n^{1/2}k^{1/6}\right)$. In section 2 we describe the algorithm, and in section 3 we give the lower bound.

16th Conference on the Theory of Quantum Computation, Communication and Cryptography (TQC 2021).
Editor: Min-Hsiu Hsieh; Article No. 6; pp. 6:1–6:5
Leibniz International Proceedings in Informatics
LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

2 Results

▶ **Theorem 1.** *For all $\varepsilon > 0$, we have $Q(\textsc{Claw}_{n \to k}) = O\big(n^{1/2+\varepsilon} k^{1/4}\big)$.*

Proof. Let $X = (x_1, \dots, x_n)$, $Y = (y_1, \dots, y_n)$ be the inputs of the function. We denote $k = n^{\varkappa}$.

Consider the following algorithm parametrized by $\alpha \in [0, 1]$.

1. **a.** Select a random sample $A = \{a_1, \dots, a_\ell\} \subseteq [n]$ of size $\ell = 4 \cdot n^{\alpha} \cdot \ln n$ and query the variables $x_{a_1}, \dots, x_{a_\ell}$.
   Denote by $X_A = \{x_a \mid a \in A\}$ the set containing their values. Do a Grover search for an element $y \in Y$ such that $y \in X_A$. If found, output 1.

   b. Select a random sample $A' = \{a'_1, \dots, a'_\ell\} \subseteq Y$ of size ℓ and query the variables $y_{a'_1}, \dots, y_{a'_\ell}$.
   Denote by $Y_{A'} = \{y_{a'} \mid a' \in A'\}$ the set containing their values. Do a Grover search for an element $x \in X$ such that $x \in Y_{A'}$. If found, output 1.
2. Run $\textsc{Claw}_{4b \ln n \to k}$ algorithm (with the value of b specified below) with the following oracle:

   a. To get x_i: do a pseudorandom permutation on $x_1, \dots, x_n$ using seed i and using Grover's minimum search return the first value x_j such that $x_j \notin X_A$.

   b. To get y_i: do a pseudorandom permutation on $y_1, \dots, y_n$ using seed i and using Grover's minimum search return the first value y_j such that $y_j \notin X_{A'}$.

Let $B = \{i \in [n] \mid x_i \notin X_A\}$, $B' = \{i \in [n] \mid y_i \notin Y_{A'}\}$ be the sets containing the indices of the variables which have values not seen in the steps 1a and 1b. We denote $|B| = b = n^{\beta}$.

Let us calculate the probability that after step 1a there exists an unseen value v which is represented in at least $n^{1-\alpha}$ variables, i.e., $v \notin X_A \wedge |\{i \in [n] \mid x_i = v\}| \geq n^{1-\alpha}$. Consider an arbitrary value $v^* \in [k]$ such that $|\{i \mid x_i = v^*\}| \geq n^{1-\alpha}$. For $i \in [\ell]$, let Z_i be the event that $x_{a_i} = v^*$. $\forall i \in [\ell]$ $\Pr[Z_i] \geq \frac{n^{1-\alpha}}{n}$. Let $Z = \sum_{i \in [\ell]} Z_i$. Then $\mathbb{E}[Z] = \ell \cdot \mathbb{E}[Z_1] \geq 4 \cdot n^{\alpha} \cdot \ln n \cdot \frac{n^{1-\alpha}}{n} = 4 \ln n$. Using Chernoff inequality (see e.g. [11]),

$$\Pr[Z = 0] \leq \exp\left(-\frac{1}{2}\,\mathbb{E}[Z]\right) \leq \exp(-2 \ln n) = \frac{1}{n^2}.$$

The probability that there exists such $v^* \in [k]$ is at most $\frac{n^{\varkappa}}{n^2} = o(1)$. Therefore, with probability $1 - o(1)$ after step 1a, every value $v \in X_B$ is represented in the input less than $n^{1-\alpha}$ times. The same reasoning can be applied to step 1b and the set B'. Therefore, with probability $1 - o(1)$ both b and b' are at most $k \cdot n^{1-\alpha} = n^{\varkappa+1-\alpha}$.

Similarly, we show that with probability $1 - o(1)$ each $x \in B$ appears as the first element from B in at least one of the permutations of the oracle in step 2. Let W_i^x be the event that $x \in B$ appears in the i-th permutation as the first element from B. $\mathbb{E}[W_i^x] = \frac{1}{b}$. Let $W^x = \sum_{i \in [4b \ln n]} W_i^x$. $\mathbb{E}[W^x] = 4b \ln n \cdot \frac{1}{b} = 4 \ln n$. $\Pr[W^x = 0] \leq \exp(-2 \ln n) = \frac{1}{n^2}$. $\Pr[\exists x \in B : W^x = 0] \leq \frac{n}{n^2} = \frac{1}{n} = o(1)$. The same argument works for B'. Therefore, if there is a collision, it will be found by the algorithm with probability $1 - o(1)$.

We also show that with probability $1 - o(1)$, in all permutations the first element from B appears no further than in position $4\frac{n}{b} \ln n$ (and similarly for B'). We denote by $P_{i,j}$ the event that in the i-th permutation in the j-th position is an element from B. $\mathbb{E}[P_{i,j}] = \frac{b}{n}$. We denote $P_i = \sum_{j \in \left[4 \cdot \frac{n}{b} \cdot \ln n\right]} P_{i,j}$. $\mathbb{E}[P_i] = 4 \cdot \ln n$. $\Pr[P_i = 0] \leq \exp(-2 \ln n) = \frac{1}{n^2}$. $\Pr[\exists i \in [4b \ln n] : P_i = 0] \leq \frac{4b \ln n}{n^2} \leq \frac{4n \ln n}{n^2} = o(1)$. Therefore, the Grover's minimum search will use at most $\tilde{O}\big(\sqrt{\frac{n}{n^{\beta}}}\big)$ queries.

The steps 1a and 1b use $\tilde{O}(n^\alpha)$ queries to obtain the random sample, and $O(\sqrt{n})$ queries to check if there is a colliding element on the other side of the input. The oracle in step 2 uses $\tilde{O}(\sqrt{\frac{n}{n^\beta}})$ queries to obtain one value of x_i or y_i.

Therefore the total complexity of the algorithm is

$$\tilde{O}\Big(n^\alpha + n^{\frac{1}{2}} + Q(\textsc{Claw}_{4b\ln n\to k})\cdot n^{\frac{1}{2}-\frac{1}{2}\beta}\Big).$$

By using the $O\big(n^{2/3}\big)$ algorithm in step 2,

$$\begin{aligned} Q(\textsc{Claw}_{4b\ln n\to k})\cdot n^{\frac{1}{2}-\frac{1}{2}\beta} &= n^{\frac{2}{3}\beta+\frac{1}{2}-\frac{1}{2}\beta} \\ &= n^{\frac{1}{2}+\frac{1}{6}\beta} \\ &\le n^{\frac{1}{2}+\frac{1}{6}(\varkappa+1-\alpha)} \\ &= n^{\frac{4+\varkappa-\alpha}{6}}, \end{aligned}$$

and the total complexity is minimized by setting $\alpha = \frac{4+\varkappa}{7}$. However, we can do better than that. Notice that the $O\big(n^{2/3}\big)$ algorithm might not be the best choice for solving $\textsc{Claw}_{4b\ln n\to k}$ in step 2.

Let $\mathcal{A}_0$ denote the regular $O\big(n^{2/3}\big)$ $\textsc{Claw}_{n\to k}$ algorithm. For $i>0$, let $\mathcal{A}_i$ denote a version of algorithm from Theorem 1 that in step 2 calls $\mathcal{A}_{i-1}$. Then we show that for all n and all $0\le\varkappa\le\frac{2}{3}$,

$$Q(\mathcal{A}_i) = \tilde{O}\Big(n^{T_i(\varkappa)}\Big),$$

where $T_i(\varkappa) = \frac{(2^i-1)\varkappa+2^{i+1}}{2^{i+2}-1}$.

The proof is by induction on i. For $i=0$, we trivially have that $Q(\mathcal{A}_0)=\tilde{O}\big(n^{2/3}\big)$. For the inductive step, consider the analysis of our algorithm. Let us set $\alpha = T_i(\varkappa)$. First, notice that $T_i(\varkappa)$ is non-decreasing in $\varkappa$ and $T_i\big(\frac{2}{3}\big)=\frac{2}{3}$ for all i. Thus for all $\varkappa\le\frac{2}{3}$, we have $T_i(\varkappa)\le\frac{2}{3}$, hence $\alpha\le\frac{2}{3}$ and $\frac{\varkappa}{1-\alpha+\varkappa}\le\frac{2}{3}$. Second, since the coefficient of $\varkappa$ is $\frac{2^i-1}{2^{i+2}-1}\le 1$ the function $T_i(\varkappa)$ is above $\varkappa$ for $\varkappa\le\frac{2}{3}$, establishing $\alpha-\varkappa\ge 0$. This confirms that $\alpha=T_i(\varkappa)$ is a valid choice of α.

It remains to show that the complexity of step 2 does not exceed $\tilde{O}\big(n^{T_i(\varkappa)}\big)$. By the inductive assumption and analysis of the algorithm, the complexity (up to logarithmic factors) of the second step is n to the power of $(1-\alpha+\varkappa)\cdot T_{i-1}\Big(\frac{\varkappa}{1-\alpha+\varkappa}\Big)+\frac{\alpha-\varkappa}{2}$. Finally, we have to show that

$$(1-T_i(\varkappa)+\varkappa)\cdot T_{i-1}\left(\frac{\varkappa}{1-T_i(\varkappa)+\varkappa}\right)+\frac{T_i(\varkappa)-\varkappa}{2}\le T_i(\varkappa).$$

By expanding $T_{i-1}(\varkappa)$ and with a slight rearrangement, we obtain

$$\frac{(2^{i-1}-1)\varkappa+2^i(1-T_i(\varkappa)+\varkappa)}{2^{i+1}-1}\le\frac{T_i(\varkappa)+\varkappa}{2}.$$

We can further rearrange the required inequality by bringing $T_i(\varkappa)$ to right hand side and everything else to the other. Then we get

$$\frac{(2^{i-1}-1+2^i-\frac{2^{i+1}-1}{2})\varkappa+2^i}{2^{i+1}-1}\le T_i(\varkappa)\left(\frac{1}{2}+\frac{2^i}{2^{i+1}-1}\right).$$

After simplification we obtain $\frac{(2^i-1)\varkappa+2^{i+1}}{2^{i+2}-1}\le T_i(\varkappa)$, which is true.

Since $\lim_{i\to\infty}\frac{2^i-1}{2^{i+2}-1}=\frac{1}{4}$ and $\lim_{i\to\infty}\frac{2^{i+1}}{2^{i+2}-1}=\frac{1}{2}$, the result follows. ◀

3 Lower Bound

We show a $\Omega\big(n^{1/2}k^{1/6}\big)$ quantum query complexity lower bound for $\textsc{Claw}_{n\to k}$.

▶ **Theorem 2.** *For all $k \geq 2$, we have $Q(\textsc{Claw}_{n\to k}) = \Omega\big(n^{1/2}k^{1/6}\big)$.*

Proof. Let $\text{p}\textsc{Search}_m : (* \cup [k])^m \to [k]$ be the partial function defined as

$$\text{p}\textsc{Search}_m(x_1, x_2, \ldots, x_m) = \begin{cases} x_i, & \text{if } x_i \neq *, \forall j \neq i : x_j = * \\ \text{undefined}, & \text{otherwise} \end{cases}.$$

Consider the function $f_{n,k} = \textsc{Claw}_{k\to k} \circ \text{p}\textsc{Search}_{\lfloor n/k \rfloor}$. One can straightforwardly reduce $f_{n,k}(x, y)$ to $\textsc{Claw}_{n\to k+2}(x', y')$ by setting

$$x'_i = \begin{cases} x_i, & \text{if } x_i \neq * \\ k+1, & \text{if } x_i = * \end{cases}$$

and

$$y'_i = \begin{cases} y_i, & \text{if } y_i \neq * \\ k+2, & \text{if } y_i = * \end{cases}.$$

Now we show that $Q(f_{n,k}) = \Omega\Big(k^{2/3}\sqrt{n/k}\Big) = \Omega\big(n^{1/2}k^{1/6}\big)$. The fact that $Q(\textsc{Claw}_{k\to k}) = \Omega\big(k^{2/3}\big)$ has been established by Zhang [16]. Furthermore, thanks to the work done by Brassard et al. in [6, Theorem 13] we know that for $\text{p}\textsc{Search}_m$ a composition theorem holds: $Q(h \circ \text{p}\textsc{Search}_m) = \Omega(Q(h) \cdot Q(\text{p}\textsc{Search}_m)) = \Omega(Q(h) \cdot \sqrt{m})$. Therefore,

$$Q(\textsc{Claw}_{n\to k}) \geq Q\Big(\textsc{Claw}_{k-2\to k-2} \circ \text{p}\textsc{Search}_{\lfloor \frac{n}{k-2} \rfloor}\Big) = \Omega\left(k^{2/3}\sqrt{\frac{n}{k}}\right) = \Omega\Big(n^{1/2}k^{1/6}\Big).$$

◀

4 Open Problems

Can we show that $Q\big(\textsc{Claw}_{n\to n^{2/3}}\big) = \Omega\big(n^{2/3}\big)$? In particular, our algorithm struggles with instances where there are $\frac{n^{2/3}}{2}$ singletons only two (or none) of which are matching and the remaining variables are evenly distributed with $\Theta\big(n^{1/3}\big)$ copies each, such that none are matching. Thus our algorithm then either has to waste time sampling all the high-frequency decoy values or have most variables not sampled by step 2. If this lower bound held, it would imply a better lower bound for evaluating constant depth formulas and Boolean matrix product verification [10, Theorem 5].

References

1 Scott Aaronson, Nai-Hui Chia, Han-Hsuan Lin, Chunhao Wang, and Ruizhe Zhang. On the Quantum Complexity of Closest Pair and Related Problems. In Shubhangi Saraf, editor, *35th Computational Complexity Conference (CCC 2020)*, volume 169 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 16:1–16:43, Dagstuhl, Germany, 2020. Schloss Dagstuhl–Leibniz-Zentrum für Informatik. doi:10.4230/LIPIcs.CCC.2020.16.

2 Scott Aaronson and Yaoyun Shi. Quantum lower bounds for the collision and the element distinctness problems. *Journal of the ACM (JACM)*, 51(4):595–605, 2004.

3 Andris Ambainis. Polynomial degree and lower bounds in quantum complexity: Collision and element distinctness with small range. *Theory of Computing*, 1(1):37–46, 2005.

4 Andris Ambainis. Quantum walk algorithm for element distinctness. *SIAM Journal on Computing*, 37(1):210–239, 2007.

5 Daniel J. Bernstein, Stacey Jeffery, Tanja Lange, and Alexander Meurer. Quantum algorithms for the subset-sum problem. In Philippe Gaborit, editor, *Post-Quantum Cryptography*, pages 16–33, Berlin, Heidelberg, 2013. Springer Berlin Heidelberg.

6 Gilles Brassard, Peter Høyer, Kassem Kalach, Marc Kaplan, Sophie Laplante, and Louis Salvail. Key establishment à la merkle in a quantum world. *Journal of Cryptology*, 32(3):601–634, 2019. doi:10.1007/s00145-019-09317-z.

7 Harry Buhrman and Ronald de Wolf. Complexity measures and decision tree complexity: a survey. *Theoretical Computer Science*, 288(1):21–43, 2002. Complexity and Logic. doi:10.1016/S0304-3975(01)00144-X.

8 Harry Buhrman, Christoph Dürr, Mark Heiligman, Peter Høyer, Frédéric Magniez, Miklos Santha, and Ronald de Wolf. Quantum algorithms for element distinctness. *SIAM Journal on Computing*, 34(6):1324–1330, 2005. doi:10.1137/S0097539702402780.

9 Andrew M. Childs and Jason M. Eisenberg. Quantum algorithms for subset finding. *Quantum Info. Comput.*, 5(7):593–604, 2005.

10 Andrew M. Childs, Shelby Kimmel, and Robin Kothari. The quantum query complexity of read-many formulas. In *Proceedings of the 20th Annual European Conference on Algorithms*, ESA'12, pages 337–348, Berlin, Heidelberg, 2012. Springer-Verlag. doi:10.1007/978-3-642-33090-2_30.

11 Fan Chung and Linyuan Lu. Concentration inequalities and martingale inequalities: a survey. *Internet Mathematics*, 3(1):79–127, 2006.

12 François Le Gall and Saeed Seddighin. Quantum meets fine-grained complexity: Sublinear time quantum algorithms for string problems, 2020. arXiv:2010.12122.

13 Frédéric Magniez, Miklos Santha, and Mario Szegedy. Quantum algorithms for the triangle problem. *SIAM Journal on Computing*, 37(2):413–424, 2007. doi:10.1137/050643684.

14 Ansis Rosmanis. Adversary lower bound for element distinctness with small range, 2014. arXiv:1401.3826.

15 Seiichiro Tani. Claw finding algorithms using quantum walk. *Theoretical Computer Science*, 410(50):5285–5297, 2009. Mathematical Foundations of Computer Science (MFCS 2007). doi:10.1016/j.tcs.2009.08.030.

16 Shengyu Zhang. Promised and distributed quantum search. In Lusheng Wang, editor, *Computing and Combinatorics*, pages 430–439, Berlin, Heidelberg, 2005. Springer Berlin Heidelberg.

Fast and Robust Quantum State Tomography from Few Basis Measurements

Daniel Stilck França ✉ 🏠

QMATH, Department of Mathematical Sciences, University of Copenhagen, Denmark

Fernando G.S L. Brandão ✉ 🏠

AWS Center for Quantum Computing, Pasadena, CA, USA
Institute for Quantum Information and Matter,
California Institute of Technology, Pasadena, CA, USA

Richard Kueng ✉ 🏠

Institute for Integrated Circuits, Johannes Kepler University Linz, Austria

Abstract

Quantum state tomography is a powerful but resource-intensive, general solution for numerous quantum information processing tasks. This motivates the design of robust tomography procedures that use relevant resources as sparingly as possible. Important cost factors include the number of state copies and measurement settings, as well as classical postprocessing time and memory. In this work, we present and analyze an online tomography algorithm designed to optimize all the aforementioned resources at the cost of a worse dependence on accuracy. The protocol is the first to give provably optimal performance in terms of rank and dimension for state copies, measurement settings and memory. Classical runtime is also reduced substantially and numerical experiments demonstrate a favorable comparison with other state-of-the-art techniques. Further improvements are possible by executing the algorithm on a quantum computer, giving a quantum speedup for quantum state tomography.

2012 ACM Subject Classification Theory of computation → Quantum computation theory; Hardware → Quantum technologies; Theory of computation → Quantum information theory; Mathematics of computing → Probabilistic inference problems

Keywords and phrases quantum tomography, low-rank tomography, Gibbs states, random measurements

Digital Object Identifier 10.4230/LIPIcs.TQC.2021.7

Related Version *Full Version*: https://arxiv.org/abs/2009.08216

Funding *Daniel Stilck França*: D.S.F. acknowledges financial support from VILLUM FONDEN via the QMATH Centre of Excellence (Grant no. 10059).
Fernando G.S L. Brandão: F.B. acknowledges funding from the US National Science Foundation (PHY1733907). The Institute for Quantum Information and Matter is an NSF Physics Frontiers Center.
Richard Kueng: R.K. acknowledges funding from the US National Science Foundation (PHY1733907). The Institute for Quantum Information and Matter is an NSF Physics Frontiers Center.

Acknowledgements We thank Chris Ferrie, David Gross, Thomas Grurl, Cécilia Lancien, Robert König, Oliver H. Schwarze and Joel Tropp for valuable input and helpful discussions.

1 Motivation

Quantum state tomography is the task of reconstructing a classical description of a quantum state from experimental data. This problem has a long and rich history [5] and remains a useful subroutine for building, calibrating and controlling quantum information processing devices. Over the last decade, unprecedented advances in the experimental control of

16th Conference on the Theory of Quantum Computation, Communication and Cryptography (TQC 2021).
Editor: Min-Hsiu Hsieh; Article No. 7; pp. 7:1–7:13
Leibniz International Proceedings in Informatics
LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

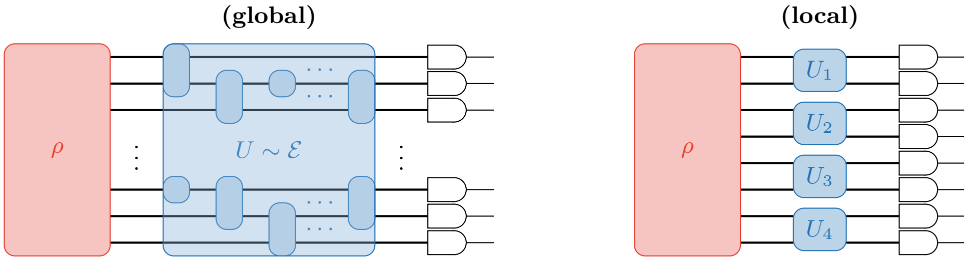


Figure 1 *Basis measurement primitive.* Global measurements (left) require implementing a global unitary that affects all n qudits prior to measuring in the computational basis. A k-local measurement primitive only allows for unitaries that affect groups of k (geometrically) local qudits; see the left-hand side for a visualization with $n = 8$ and $k = 2$.

quantum architectures have pushed traditional estimation techniques to the limit of their capabilities. This is mainly due to a fundamental curse of dimension: the dimension of state space grows exponentially in the number of qudits, i.e. a quantum system comprised of n d-dimensional qudits is characterized by a density matrix ρ of size $D = d^n$. The impact of this scaling behavior is further amplified by the probabilistic nature of quantum mechanics ("wave-function collapse"). Information about the state is only accessible via measuring the system. An informative quantum measurement is destructive and only yields probabilistic outcomes. Hence, many identically prepared samples of the quantum state are required to estimate even a single parameter of the underlying state. Characterizing the full state of a quantum system necessitates accurate estimation of many such parameters. Storing and processing the measurement data also requires substantial amounts of classical memory and computing power – another important practical bottleneck. To summarize: the curse of dimension and wave-function collapse have severe implications that necessitate the design of extremely resource-efficient protocols.

In this work, we focus on reconstructing the complete density matrix ρ from single-copy measurements. This is an actual restriction, as it excludes some of the most powerful tomography techniques known to this date [34, 19]. While very efficient in terms of state copies, these procedures are extremely demanding in terms of quantum hardware – an actual implementation would require exponentially long quantum circuits that act collectively on all the copies of the unknown state stored in a quantum memory.

We also adopt a measurement primitive that mimics the layout of modern quantum information processing devices. Apply a unitary U to the unknown state $\rho \mapsto U\rho U^\dagger$ and perform measurements in the computational basis $\{|i\rangle : i = 1, \ldots, D\}$. Fixing U and repeating this procedure many times allows for estimating the associated outcome distribution:

$$[p_U(\rho)]_i = \langle i|U\rho U^\dagger|i\rangle \quad \text{for } i = 1, \ldots, D. \tag{1}$$

This outcome distribution characterizes the diagonal elements of $U\rho U^\dagger$. In general, access to a single diagonal is insufficient to determine ρ unambiguously. Instead, multiple repetitions of this basic measurement primitive are necessary. We refer to Fig. 1 for an illustration. Different ensembles $\mathcal{E}$ of accessible unitary transformations give rise to different basis measurement primitives. When employed to perform state tomography – i.e. reconstruct an unknown state ρ up to accuracy ϵ in trace distance – the following fundamental scaling laws apply to *any* (single-copy) basis measurement primitive and *any* tomographic procedure:

Table 1 *Resource scaling for state tomography protocols based on global measurements (single copy):* Here, D denotes the Hilbert space dimension, r is the rank of the target state and ϵ is the desired precision (in trace distance). We have suppressed constants, as well as logarithmic dependencies in D and r. The first row summarizes known fundamental lower bounds, while the label "unknown" indicates a lack of rigorous theory support.

	meas. primitive	**basis settings**	**state copies**	**runtime**	**memory**
lower bounds	arbitrary	$\geq r$	$\geq Dr^2\epsilon^{-2}$	$\geq Dr^2\epsilon^{-2}$	$\geq Dr$
CS [40]	Haar	r	unknown	D^4	D^3
CS [27]	Clifford	$D^{2/3}r$	unknown	D^4	D^3
PLS [18]	2-design	D	$Dr^2\epsilon^{-2}$	D^3	D^2
this work	4-design	$r\epsilon^{-2}$	$Dr^2\epsilon^{-4}$	$D^2r^{5/2}\epsilon^{-5}$	$Dr\epsilon^{-2}$
this work	Clifford	$r^3\epsilon^{-2}$	$Dr^4\epsilon^{-4}$	$D^2r^6\epsilon^{-5}$	$Dr^2\epsilon^{-2}$

(i) The *number of basis measurement settings* M must scale at least linearly with the (effective) target rank $r = \mathrm{rank}(\rho)$: $M = \Omega(r)$. This corresponds to estimating a total of $DM = \Omega(rD)$ parameters [21, 25].

(ii) The *sampling rate* N, i.e. the number of independent state copies required to obtain sufficient data, must depend on rank, dimension and desired accuracy: $N = \Omega\left(Dr^2/\epsilon^2\right)$ [19].

(iii) The *classical storage* S is bounded by dimension times target rank: $S = \Omega(rD)$.

Constraint iii. follows from a simple parameter counting argument – specifying a general $D \times D$-matrix with rank r requires (order) rD parameters – while i. and ii. reflect fundamental limitations that have only been identified comparatively recently. These bounds cover three of the four most relevant cost parameters. For the last one we are not aware of a nontrivial rigorous lower bound:

(iv) The *classical runtime* associated with processing the measurement data to produce an estimated state $\sigma_\star$ should be as fast as possible.

The last decade has seen the development of several procedures that provably optimize some of these four cost factors up to logarithmic factors in the ambient dimension. We refer to Table 1 for a detailed tabulation of resource requirements. For now, we content ourselves with emphasizing that existing procedures have been designed to either minimize the number of measurement settings (compressed sensing approaches [17, 32, 28]) or the required number of samples per measurement (least-squares approaches [37, 18]). Neither of these approaches seems to be well-suited for optimizing classical postprocessing memory and time. Finally, we point out that currently available quantum technologies are not perfect [35]. Practical tomography procedures should be *robust* with respect to imperfections, most notably state preparation and measurement errors.

2 Overview of results

In this work, we develop a robust algorithm for almost resource-optimal quantum state tomography from (single-copy) basis measurements that comes with rigorous convergence guarantees. The theoretical results are closely related to quantum state distinguishability [23, 22, 3, 33] and strongest for global measurement primitives (Fig. 1, left) that are sufficiently generic. In the regime of low target rank r, the proposed method improves upon state-of-the art techniques at the cost of a worse dependence on target accuracy ϵ. The actual numbers are summarized in Table 1. The required number of basis measurement setting matches results from compressed sensing [17, 32, 28] – a technique that has been specifically designed to optimize this cost function – while the required number of state copies is comparable to projected least squares [37, 18] – which is known to be (almost) optimal in this regard. Classical runtime and memory cost are also reduced substantially. We also obtain rigorous results for k-local measurement primitives (Fig. 1, right), but the obtained theoretical numbers only become competitive if the locality parameter k is sufficiently large. We believe that this shortcoming is an artifact of poor constants and refer to App. B.4 of the extended version [9] for details.

2.1 Algorithm and theoretical runtime guarantee

The tomography algorithm – which we call *Hamiltonian updates* – is based on a variant of the versatile mirror-descent meta-algorithm [38, 10], see also [7]. Mirror descent and its cousin, matrix multiplicative weights, have led to considerable progress in algorithm design across several disciplines. Prominent examples include fast semidefinite programming solvers [20, 4, 31, 39, 8, 6, 7], quantum prediction techniques like shadow tomography [1], the online learning methods of [2] and the tomography protocol of [41]. The algorithm design is summarized in Algorithm 1. The key idea is to maintain and iteratively update a guess for the unknown state. The sequence of guess states is parametrized by Hamiltonians

$$\sigma_t = \frac{\exp(-H_t)}{\mathrm{tr}(\exp(-H_t))} \quad \text{for} \quad t = 0, 1, 2, \ldots \qquad \text{(Gibbs / thermal state)}$$

and initialized to an infinite temperature state $\sigma_0 = \mathbb{I}/D$ (maximum entropy principle). At each subsequent iteration, we choose a unitary rotation $U \sim \mathcal{E}$ *at random* from a fixed ensemble, estimate the outcome distribution (1) of the rotated target state $U\rho U^\dagger$ and compare it to the predicted outcome distribution $U\sigma_t U^\dagger$ of the current guess state. If the two outcome distributions differ by more than mere statistical fluctuations, σ_t is an inadequate guess for ρ.

We then update the guess state $\sigma_t \mapsto \sigma_{t+1}$ by including a small energy penalty in the associated Hamiltonian that penalizes the observed mismatch and repeat. Heuristically, it is reasonable to expect that this update rule makes progress as long as each newly selected basis provides actionable advice, i.e. discrepancies in the outcome distributions. As we prove in App. A of the extended version [9] that we indeed make progress in relative entropy. Things get more interesting when this is not the case. Predicted and estimated outcome distribution can be very close for two reasons (i): the current iterate σ_t is close to the unknown target ρ (*convergence*); (ii) the current basis measurement cannot properly distinguish between σ_t and ρ, even though they are still far apart (*false positive*). It is imperative to protect against wrongfully terminating the procedure due to the occurrence of a false positive. Hamiltonian Updates (Algorithm 1) suppresses the likelihood of wrongfully terminating by checking closeness in (up to) L additional random bases. The required size of such a control loop depends on the measurement primitive. Broadly speaking, generic measurement ensembles – like Haar-random unitary transformations – are very unlikely to produce false positives; while highly structured ensembles – like mutually unbiased bases – can be much more susceptible. The following relation introduces two ensemble-dependent summary parameters that capture this effect:

$$\mathrm{Pr}_{U\sim\mathcal{E}}\left[\|p_U(\rho) - p_U(\sigma_t)\|_{\ell_1} \geq \theta_{\mathcal{E}}(\rho, \sigma_t)\|\rho - \sigma_t\|_2\right] \geq \tau_{\mathcal{E}}(\rho, \sigma_t). \tag{2}$$

The parameter $\theta_{\mathcal{E}}(\rho, \sigma_t)$ relates an observed discrepancy in outcome distributions (measured in ℓ_1 distance) to the Frobenius distance in state space. As detailed below, it captures the minimal progress we can expect from a successful update $\sigma_t \mapsto \sigma_{t+1}$. The second parameter

Algorithm 1 *Hamiltonian Updates for quantum state tomography.*

```
Input: error tolerance $\epsilon$, number of loops $L$.
Initialize: $t = 0$, $H_t = 0$, CONVERGENCE=FALSE
while CONVERGENCE=FALSE do
    compute $\sigma_t = \exp(-H_t)/\mathrm{tr}(\exp(-H_t))$        ▷ current guess for the state $\rho$
    select random basis measurement $\{U|i\rangle\langle i|U^\dagger\}$
    compute outcome statistics $[p_i]$ of $\sigma_t$        ▷ classical computation
    estimate outcome statistics $[q_i]$ of $\rho$        ▷ quantum measurement
    check if $[p_i]$ and $[q_i]$ are $\epsilon$-close in $\ell_1$ distance
    if NO then set $P = \sum_{p_i > q_i} |i\rangle\langle i|$        ▷ collect outcomes for which $p_i > q_i$
        Set $\eta = \frac{1}{8}\|p - q\|_{\ell_1}$
        $H_{t+1} \leftarrow H_t + \eta U^\dagger P U$        ▷ energy penalty for mismatch (in this basis)
        update $\sigma_{t+1} = \exp(-H_{t+1})/\mathrm{tr}(\exp(-H_{t+1}))$
        $t \leftarrow t + 1$        ▷ update counter of number of iterations
    else if YES then        ▷ current guess may be close to $\rho$
        check $L$ additional random bases        ▷ suppress likelihood of false positives
        if $\ell_1$ distance is always $< \epsilon$ then        ▷ current guess is likely to be close
            set CONVERGENCE=TRUE
        end if
    end if
end while
Output: $H_t$
```

$\tau_{\mathcal{E}}(\rho, \sigma_t)$ lower bounds the probability of observing an outcome discrepancy that appropriately reflects the current stage of convergence. This parameter controls the size of the control loop. It is desirable to choose both parameters as large as possible, but there is a trade-off (making $\theta_{\mathcal{E}}(\rho, \sigma_t)$ larger necessarily diminishes $\tau_{\mathcal{E}}(\rho, \sigma_t)$) and both depend heavily on the measurement ensemble. One of our main theoretical contributions is a rigorous convergence guarantee for Hamiltonian updates (Algorithm 1) that only depends on the ambient dimension D, the target rank $r = \mathrm{rank}(\rho)$, as well as the worst-case ensemble parameters

$$\theta_{\mathcal{E}}(\rho) = \max_{\sigma \text{ state}} \theta_{\mathcal{E}}(\rho, \sigma) \quad \text{and} \quad \tau_{\mathcal{E}}(\rho) = \max_{\sigma \text{ state}} \tau_{\mathcal{E}}(\rho, \sigma). \tag{3}$$

▶ **Theorem 1** (informal statement). *Fix a measurement primitive $\mathcal{E}$, a desired accuracy ϵ and let ρ be a rank-r target state. With high probability, Algorithm 1 requires at most $T = \mathcal{O}\left(r \log(D)/(\theta_{\mathcal{E}}(\rho)\epsilon)^2\right)$ steps – each with a control loop of size $L = \mathcal{O}(\log(T)/\tau_{\mathcal{E}}(\rho))$ – to produce an output $\sigma_\star$ that obeys $\|\rho - \sigma_\star\|_1 \leq \epsilon$.*

This convergence guarantee is also stable with respect to imperfect implementations. In particular, we only need to estimate measurement outcome statistics to a certain degree of accuracy: $\mathcal{O}\left(Dr/(\theta_{\mathcal{E}}(\rho)\epsilon)^2\right)$ measurement repetitions suffice for each basis. This implies that the total number of measurement settings and state copies are bounded by

$$M = TL \simeq \mathcal{O}\left(r \log(D)/(\tau_{\mathcal{E}}(\rho)\theta_{\mathcal{E}}(\rho)^2\epsilon^2)\right) \quad \text{(measurement settings)}, \tag{4}$$

$$N \simeq \mathcal{O}\left(Dr^2 \log(D)/(\tau_{\mathcal{E}}(\rho)\theta_{\mathcal{E}}(\rho)^4\epsilon^4)\right) \quad \text{(sample complexity)}. \tag{5}$$

To increase readability, we have suppressed the logarithmic contribution in T.

2.2 Connections to quantum state distinguishability

The bounds for M in Eq. (4) and N in Eq. (5) are characterized by worst-case ensemble parameters (2). These are intimately related to quantum state distinguishability: how good is a fixed measurement primitive $\mathcal{E}$ at distinguishing state ρ from state σ in the single-shot limit? Ambainis and Emerson [3] showed that the optimal probability of successful discrimination is given by $p_{\mathrm{succ}} = \frac{1}{2} + \frac{1}{4}\mathbb{E}_{U\sim\mathcal{E}}\|p_U(\rho) - p_U(\sigma)\|_{\ell_1}$ and achieved by the maximum likelihood rule, see also [33]. It is possible to relate this bias to the Frobenius distance in state space:

$$\mathbb{E}_{U\sim\mathcal{E}}\|p_U(\rho) - p_U(\sigma)\|_{\ell_1} \geq \lambda_{\mathcal{E}}(\rho,\sigma)\|\rho - \sigma\|_2.$$

The proportionality constant $\lambda_{\mathcal{E}}(\rho,\sigma)$ measures how well the measurement primitive is equipped to distinguish ρ from σ. It is closely related to the ensemble parameters defined in Eq. (2) and has been the subject of considerable attention in the community. Tight bounds have been derived for a variety of measurement primitives, such as Haar random unitaries and approximate 4-designs [3, 33], random Clifford unitaries [29] and k-local (approximate) 4-designs [30]. Simple probabilistic arguments allow for converting these assertion into lower bounds on both $\theta_{\mathcal{E}}(\rho)$ and $\tau_{\mathcal{E}}(\rho)$. Inserting these bounds into Eq. (4) and Eq. (5) then implies the measurement and sample complexity assertions advertised in Table 1. We refer to Appendix B of the extended version [9] for a detailed case-by-case analysis and content ourselves here with an overview. We start with the strongest measurement primitive: Haar random unitaries and approximate 4-designs achieve $\theta_{\mathcal{E}}(\rho), \tau_{\mathcal{E}}(\rho) = \mathrm{const}$ for any target state. Hence, $M = \mathcal{O}(r\log(D))/\epsilon^2)$ basis settings and $N = \mathcal{O}(Dr^2\log(D)/\epsilon^4)$ state copies suffice. Clifford random measurements achieve $\theta_{\mathcal{E}}(\rho) \sim r^{-\frac{1}{2}}, \tau_{\mathcal{E}}(\rho) \sim r^{-2}$. That is, they only have a worse dependency on the rank, but perform as well as Haar measurements in terms of the ambient dimension. On the other hand, more local measurement settings defined by unitaries acting on at most k qubits have $\theta_{\mathcal{E}}(\rho) \sim \exp(-\mathcal{O}(n/k)), \tau_{\mathcal{E}}(\rho) \sim \exp(-\mathcal{O}(n/k))$, showing an (exponentially) worse dependency on the number of qudits when compared to Haar measurements. Empirical studies below do, however, suggest a much more favorable performance in practice.

This scaling highlights both a core strength and a core weakness of Hamiltonian updates. In terms of dimension D and rank r, these numbers saturate fundamental lower bounds on any tomographic procedure up to a logarithmic factor. However, the number of measurement settings also depends inverse quadratically on the accuracy. Furthermore, the accuracy enters as ϵ^{-4}, not ϵ^{-2} in the sample complexity. Thus, high accuracy solutions do not only require many samples, but also many basis measurement settings. This drawback is a consequence of a “curse of mirror descent (or multiplicative weights)”. These meta-algorithms are very efficient in terms of problem dimension, but scale comparatively poorly in accuracy [4]. However, inverse polynomial scaling in accuracy ϵ is an unavoidable feature of quantum state tomography. Hence, tomography is a reasonable setting to apply algorithms that trade dimensional dependency for accuracy. Moreover, for most applications, it suffices to recover the state up to precision $\epsilon = \mathcal{O}(\mathrm{polylog}(D)^{-1})$.

3 Summary and comparison to relevant existing work

We propose a variant of mirror descent [38, 10] to obtain resource-efficient algorithms for quantum state tomography. In recent years, mirror descent and its cousins have been extensively used to obtain fast SDP solvers [20, 4, 31, 39, 8, 6, 7], to develop prediction algorithms like shadow tomography [1], the online learning methods of [2] and the tomography protocol of [41]. Key advantages are resource efficiency, as well as intrinsic resilience towards noise. Empirical studies summarized in Fig. 2 confirm these theoretical assertions. A downside is, however, that the number of iterations may depend on the desired target accuracy ϵ. We focus on obtaining a ϵ-approximation in trace distance of a D-dimensional state ρ from (random) basis measurements on i.i.d. copies (*global classical description*). Our goal is to optimize the different resources required for that task. These include the number of state copies (sample complexity), the cost for processing measurement data (classical postprocessing), as well as the associated memory cost. The multipronged resource efficiency of our results becomes particularly pronounced if the underlying target state has (approximately) low-rank $r \ll D$. This is a natural assumption in most applications, but can also be relaxed to states with low Rényi entropy, see App. G of the extended version [9].

Thus, our results are similar in spirit to the tomography algorithms based on compressed sensing (CS) [17, 32, 14, 36, 28], or projected least squares (PLS) [37, 18]. These also focus on rigorous and (nearly) optimal sample complexity in the low-rank regime combined with efficient postprocessing. Table 1 summarizes the resources required for these protocols, as well as our new results. These compare favorably with existing methods. We note that for approximate 4-design measurements, both sample complexity and memory – as functions of D and r – are essentially optimal [34, 19]. Compared to existing approaches, we obtain significant savings in both runtime and memory. Moreover, as pointed out in [41], there are also qualitative advantages.

Current schemes that minimize the number of basis settings [40, 27] are only known to do so with perfect knowledge of the underlying measurement outcomes. This will never be the case in practice, due to statistical fluctuations. Thus, to the best of our knowledge, our work is the first to rigorously obtain recovery guarantees with imperfect knowledge of outcomes and basis settings that only scale logarithmically with the ambient dimension and linearly with rank (albeit with the extra ϵ dependency).

The focus of this work differs from other recent applications of mirror descent to quantum learning [2, 1, 6]. Broadly speaking, these works focus on obtaining a classical description of the state – a shadow – that approximately reproduces a fixed set of target observables. This is a different and weaker form of recovery. Moreover, these works prioritize sample complexity, not necessarily classical postprocessing resources. Minimizing these classical resources is a core focus of this work.

Having said this, the idea of using (variants of) mirror descent for quantum state (and process) tomography is not completely new. Similar ideas were proposed in Refs. [13, 16] and have been experimentally tested [11, 24]. More recently, Youssry, Tomamichel and Ferrie proposed and analyzed state tomography based on matrix exponentiated gradient descent [41]. They focused on the practically relevant case of local (single-qubit) Pauli measurements and established convergence to the target state as the number of samples goes to infinity. They also pointed out conceptual advantages, such as online implementation and noise-robustness. The results presented here add to this promising picture. We equip (a variant of) mirror descent with rigorous performance guarantees in the non-asymptotic setting, optimize actual implementations and establish robustness in a more general setting. Moreover, our results apply to any measurement procedure that is capable of distinguishing arbitrary pairs of quantum states.

We also want to point out that the method presented here could also be implemented on a quantum computer. This would result in substantial runtime savings – a quantum speedup for quantum state tomography. Suppressing polylogarithmic terms, a runtime of order $\tilde{\mathcal{O}}(D^{\frac{3}{2}}r^3\epsilon^{-9})$ suffices to obtain a *classical* description of the target state. We refer to App. E of the extended version [9] for details and proofs. To the best of our knowledge, this

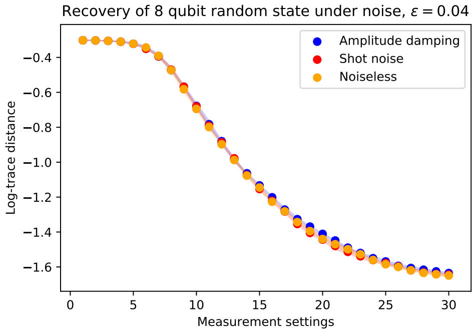


Figure 2 *Convergence of Algorithm 1 for different noise models.* We consider Haar-random global measurements of a 8-qubit pure target state with target accuracy $\epsilon = 0.04$. Different colors track convergence for different noise models: (blue) amplitude damping noise with parameter $\epsilon/4$; (red) white noise with standard deviation $\epsilon/4$ that mimics one-shot noise; (orange) zero noise. All logarithms are base 10 and the shaded areas indicate 25% and 75% quartiles, estimated from 20 samples.

is the first quantum speedup for low-rank tomography beyond the results of Kerenidis and Prakash [26] which cover pure, real target states ($r = 1$) exclusively and work under the stronger assumption of access to a controlled unitary that prepares the state.

Finally, we want to emphasize that the proposed reconstruction procedure can be empowered by advantageous measurement structure. Storage-efficiency stems from the fact that we can keep track of the Hamiltonian – not the associated Gibbs state – which inherits structure from the underlying measurement procedure. Runtime savings are achieved by only exponentiating the Hamiltonian approximately and exploiting fast matrix-vector multiplication. We refer to App. D of the extended version [9] for details and content ourselves here with a vague, but instructive, analogy: View Algorithm 1 as an adaptive cool-down procedure. We start with a Gibbs state at infinite temperature and, at each step, we cool down the system in a controlled fashion that guides the thermal state towards the unknown target. Importantly, each update is small and the number of total cooling steps is also benign. Hence, we never truly leave the moderate temperature regime and avoid computational bottlenecks that typically only arise at low temperatures. In turn, the output of our algorithm is in the form of a Hamiltonian whose Gibbs state is close to the target state. A list of Gibbs state eigenvalues and corresponding eigenvectors can be obtained by block Krylov iterations, see App. F of the extended version [9]. Runtime and memory cost of this conversion procedure can never exceed those of Algorithm 1.

4 Numerical experiments

We complement our theoretical assertions with empirical test evaluations for systems comprised of up to 10 qubits. The results look promising and may establish Algorithm 1 as a practical tool for quantum state tomography. We remark that our numerical implementation has two additional details when compared with the one described in Algorithm 1. Although these modifications do not change the asymptotic runtime analysis of the algorithm, they can substantially reduce runtime and sample complexity in practice.

The first alteration we do is to recycle the last measurement data after a successful update. More precisely, after each update $\sigma_t \to \sigma_{t+1}$, we then check if the new iteration σ_{t+1} is still distinguishable from ρ under the previous measurement basis. Only if this is

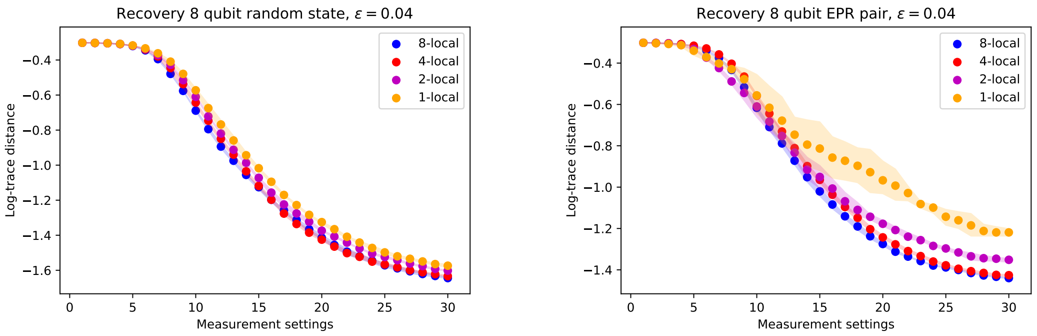


Figure 3 *Convergence of Algorithm 1 for different measurement localities.* Different colors track convergence (in logarithmic trace distance) for 8-qubit basis measurements with different localities and target accuracy $\epsilon = 0.04$. Individual basis measurements are subject to white noise with standard deviation $\epsilon/4$. (Left) Reconstruction of a generic pure target state. (Right) Reconstruction of a highly structured target state (EPR/Bell state). All logarithms are base 10 and the shaded area indicate 25% and 75% quartiles, estimated from 20 samples.

not the case, we move on to sample a new measurement setting. Otherwise, we re-use the already known measurement basis to drive another update in the same direction. We observe empirically that this minor modification has very desirable consequences. It leads to a much faster convergence throughout early stages of the algorithm and, by extension, reduces the number of required measurement settings significantly.

What is more, this recycling procedure cannot change the asymptotic scaling of the algorithm. To see this, note that the modification can only affect postprocessing complexity. Indeed, it clearly does not require us to sample more states or measurement settings. Finding another violation can only bring us closer to the state in relative entropy. And the postprocessing time can only double in the worst case. This worst case scenario happens when after updating every basis once, we have already converged in that basis and checking again does not lead to further convergence. We will refer to this variation as the *last step recycling strategy*. It is explained in detail in the appendix (Algorithm 2 of the extended version [9]).

Other variations of this basic principle come to mind. For instance, we need not stop at testing the current iteration against the previous measurement basis. We can also test it against all measurements that have already accumulated. This variation can further reduce the (total) number of basis settings required to converge. Fig. 4 confirms this intuition. However, this strategy comes at the expense of an increase in the computational complexity of the postprocessing. We refer to this strategy as the *complete recycling strategy*.

Apart from these practical improvements, we have also tested desirable fundamental properties of Algorithm 1. Chief among them is noise resilience. As advertised in Sec. 2 and proved in App. C of the extended version [9], the performance of the algorithm under arbitrary noise of bounded intensity is indistinguishable from the noiseless case. This feature is empirically confirmed by Fig. 2. For detecting a random pure state on 8 qubits, different noise sources – such as shot noise and amplitude damping – affect convergence in a very mild fashion only (*robustness*). It is also interesting to note that the convergence in trace norm appears to be polynomial for the first measurements and then switches to an exponential phase.

Another interesting figure of merit is measurement locality. The assertions that underpin Algorithm 1 do, in principle, extend to local measurement primitives. But, as detailed in App. B.4 of the extended version [9], the resulting numbers look rather pessimistic and scale unfavorably with measurement locality k. Empirical studies do paint a much more favorable picture, see Fig. 3. The two subplots address reconstruction of a typical 8-qubit target state (left), as well as a highly structured one (right). A direct comparison lends credence to a conjecture voiced in App. B.4 of the extended version [9] below: generic or typical states are easier to reconstruct with local measurements than highly structured ones.

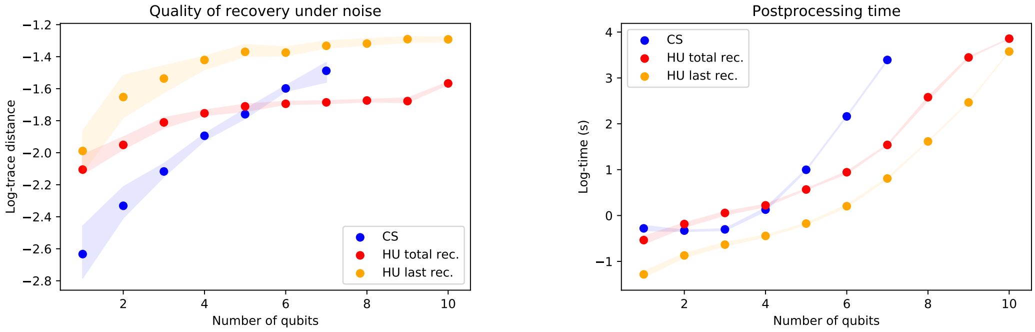


Figure 4 *Comparison between Algorithm 1 (HU) and compressed sensing (CS) tomography.* (Left) Reconstruction of a random n-qubit pure state from 15 globally random basis measurements corrupted by amplitude damping noise ($p = 0.005$). Different colors track the logarithmic trace distance error achieved by either CS (blue) or variants of HU (orange and red) for $\epsilon = 0.01$. Shaded regions indicate the $25 - 75$ percentiles over 20 independent runs. (Right) Empirical runtime for executing (naive implementations of) the three different reconstruction procedures on a conventional laptop. CVX [12] – a standard solver for semidefinite programs – could not go beyond 7 qubits.

Last but not least, we compare Algorithm 1 against the state of the art regarding tomography from very few basis measurements. Compressed sensing (CS) [17, 14, 27, 28] has been designed to fit a low rank solution to the observed measurement data by also minimizing the trace norm over the cone of positive semidefinite matrices ($X \succeq 0$):

$$\text{minimize}_{X\succeq 0} \quad \text{tr}(X) \quad \text{subject to} \qquad \sum\nolimits_{i=1}^{M} \|\hat{p}_{U_i}(\rho) - p_{U_i}(X)\|_{\ell_2}^2 \leq \epsilon. \tag{6}$$

Fig. 4 compares Algorithm 1 with compressed sensing (CS). CS is contingent on solving a semidefinite program. We used CVX [12], a standard SDP solver, in Python. Algorithm 1 has also been implemented in Python. Open source code is available at [15]. We see that Hamiltonian Updates is more noise-resilient than CS. The rightmost plot also underscores the importance of memory improvements. A high-end desktop computer already struggles to solve SDP (6) for 8 qubits (even though the extrapolated computation time Fig. 4 still seems reasonable), while 10 qubits (and more) have not been a problem for Algorithm 1. We believe that Fig. 4 conveys both quantitative and qualitative advantages of Hamiltonian Updates over CS methods. This seems particularly noteworthy, because we compared both procedures for pure target states (rank$(\rho) = 1$) – a use-case tailor-made for CS approaches. We also stress that the implementation of the algorithm used to generate this data was not optimized, there is room for further improvements.

Let us conclude with the most important take-away from Figs. 2, 3 and 4. The theoretical assertions from Sec. 2 carry over to practice. Moreover, recycling of data ensures that the number of measurement settings remains small even if we try to characterize the state up to high precision. Our theoretical results suggest that order 10^5 algorithm iterations, and thus also measurement settings, might be required to obtain a $\epsilon = 10^{-2}$-approximation of a pure state in dimension $D = 2^{10}$. But our numerics demonstrate that already order 10^1 suffice to achieve convergence. The main theoretical drawbacks of Algorithm 1 – most notably, the poor scaling in accuracy – may be a non-issue in practical use cases. These findings establish our algorithm as a rare instance of a method that is provably (essentially) optimal and has a competitive performance in practice.

References

1 Scott Aaronson. Shadow tomography of quantum states. In *STOC'18—Proceedings of the 50th Annual ACM SIGACT Symposium on Theory of Computing*, pages 325–338. ACM, New York, 2018. doi:10.1145/3188745.3188802.

2 Scott Aaronson, Xinyi Chen, Elad Hazan, Satyen Kale, and Ashwin Nayak. Online learning of quantum states. *Journal of Statistical Mechanics: Theory and Experiment*, 2019(12):124019, 2019. doi:10.1088/1742-5468/ab3988.

3 Andris Ambainis and Joseph Emerson. Quantum t-designs: t-wise independence in the quantum world. In *22nd Annual IEEE Conference on Computational Complexity (CCC 2007), 13-16 June 2007, San Diego, California, USA*, pages 129–140. IEEE Computer Society, 2007. doi:10.1109/CCC.2007.26.

4 Sanjeev Arora and Satyen Kale. A combinatorial, primal-dual approach to semidefinite programs. *J. ACM*, 63(2):12:1–12:35, 2016. doi:10.1145/2837020.

5 K Banaszek, M Cramer, and D Gross. Focus on quantum tomography. *New J. Phys*, 15(12):125020, 2013. doi:10.1088/1367-2630/15/12/125020.

6 Fernando G. S. L. Brandão, Amir Kalev, Tongyang Li, Cedric Yen-Yu Lin, Krysta M. Svore, and Xiaodi Wu. Quantum SDP solvers: large speed-ups, optimality, and applications to quantum learning. In *46th International Colloquium on Automata, Languages, and Programming*, volume 132 of *LIPIcs. Leibniz Int. Proc. Inform.*, pages Art. No. 27, 14. Schloss Dagstuhl. Leibniz-Zent. Inform., Wadern, 2019.

7 Fernando G. S. L. Brandão, Richard Kueng, and Daniel Stilck França. Faster quantum and classical SDP approximations for quadratic binary optimization. *preprint arXiv:1909.04613*, 2019.

8 Fernando G. S. L. Brandão and Krysta M. Svore. Quantum speed-ups for solving semidefinite programs. In Chris Umans, editor, *58th IEEE Annual Symposium on Foundations of Computer Science, FOCS 2017, Berkeley, CA, USA, October 15-17, 2017*, pages 415–426. IEEE Computer Society, 2017. doi:10.1109/FOCS.2017.45.

9 Fernando G.S.L. Brandão, Richard Kueng, and Daniel Stilck França. Fast and robust quantum state tomography from few basis measurements, 2020. arXiv:2009.08216v2. arXiv:2009.08216.

10 Sébastien Bubeck. Convex optimization: Algorithms and complexity. *Found. Trends Mach. Learn.*, 8(3-4):231–357, 2015. doi:10.1561/2200000050.

11 Robert J. Chapman, Christopher Ferrie, and Alberto Peruzzo. Experimental demonstration of self-guided quantum tomography. *Phys. Rev. Lett.*, 117:040402, July 2016. doi:10.1103/PhysRevLett.117.040402.

12 Inc. CVX Research. CVX: Matlab software for disciplined convex programming, version 2.0. http://cvxr.com/cvx, 2012.

13 Christopher Ferrie. Self-guided quantum tomography. *Phys. Rev. Lett.*, 113:190404, November 2014. doi:10.1103/PhysRevLett.113.190404.

14 Steven T Flammia, David Gross, Yi-Kai Liu, and Jens Eisert. Quantum tomography via compressed sensing: error bounds, sample complexity and efficient estimators. *New J. Phys.*, 14(9):095022, 2012. doi:10.1088/1367-2630/14/9/095022.

15 Daniel Stilck Franca. Hamiltonian updates tomography. https://github.com/dsfranca/hamiltonian_updates_tomography, 2020.

16 Christopher Granade, Christopher Ferrie, and Steven T Flammia. Practical adaptive quantum tomography. *New J. Phys*, 19(11):113017, November 2017. doi:10.1088/1367-2630/aa8fe6.

17 David Gross, Yi-Kai Liu, Steven T. Flammia, Stephen Becker, and Jens Eisert. Quantum state tomography via compressed sensing. *Phys. Rev. Lett.*, 105:150401, 2010. doi:10.1103/PhysRevLett.105.150401.

18 Madalin Guţă, Jonas Kahn, Richard Kueng, and Joel A Tropp. Fast state tomography with optimal error bounds. *J. Phys. A*, 53(20):204001, 2020. doi:10.1088/1751-8121/ab8111.

19 Jeongwan Haah, Aram Wettroth Harrow, Zhengfeng Ji, Xiaodi Wu, and Nengkun Yu. Sample-optimal tomography of quantum states. *IEEE Trans. Inf. Theory*, 63(9):5628–5641, 2017. doi:10.1109/TIT.2017.2719044.

20 Elad Hazan. *Efficient algorithms for online convex optimization and their applications*. PhD thesis, Princeton University, 2006.

21 Teiko Heinosaari, Luca Mazzarella, and Michael M. Wolf. Quantum tomography under prior information. *Commun. Math. Phys*, 318(2):355–374, 2013. doi:10.1007/s00220-013-1671-8.

22 Carl W. Helstrom. Quantum detection and estimation theory. *J. Statist. Phys.*, 1:231–252, 1969. doi:10.1007/BF01007479.

23 Alexander S. Holevo. Statistical decision theory for quantum systems. *J. Multivariate Anal.*, 3:337–394, 1973. doi:10.1016/0047-259X(73)90028-6.

24 Zhibo Hou, Jun-Feng Tang, Christopher Ferrie, Guo-Yong Xiang, Chuan-Feng Li, and Guang-Can Guo. Experimental realization of self-guided quantum process tomography. *Phys. Rev. A*, 101:022317, February 2020. doi:10.1103/PhysRevA.101.022317.

25 Michael Kech and Michael M. Wolf. Constrained quantum tomography of semi-algebraic sets with applications to low-rank matrix recovery. *Inf. Inference*, 6(2):171–195, 2017. doi:10.1093/imaiai/iaw019.

26 Iordanis Kerenidis and Anupam Prakash. A Quantum Interior Point Method for LPs and SDPs. *ACM Transactions on Quantum Computing*, 1(1):1–32, December 2020. doi:10.1145/3406306.

27 Richard Kueng. Low rank matrix recovery from few orthonormal basis measurements. In *2015 International Conference on Sampling Theory and Applications (SampTA)*, pages 402–406, 2015.

28 Richard Kueng, Holger Rauhut, and Ulrich Terstiege. Low rank matrix recovery from rank one measurements. *Appl. Comput. Harmon. Anal.*, 42(1):88–116, 2017. doi:10.1016/j.acha.2015.07.007.

29 Richard Kueng, Huangjun Zhu, and David Gross. Distinguishing quantum states using Clifford orbits. *preprint arXiv:1609.08595*, 2016.

30 Cécilia Lancien and Andreas Winter. Distinguishing multi-partite states by local measurements. *Comm. Math. Phys.*, 323(2):555–573, 2013. doi:10.1007/s00220-013-1779-x.

31 James R. Lee, Prasad Raghavendra, and David Steurer. Lower bounds on the size of semidefinite programming relaxations. In Rocco A. Servedio and Ronitt Rubinfeld, editors, *Proceedings of the Forty-Seventh Annual ACM on Symposium on Theory of Computing, STOC 2015, Portland, OR, USA, June 14-17, 2015*, pages 567–576. ACM, 2015. doi:10.1145/2746539.2746599.

32 Yi-Kai Liu. Universal low-rank matrix recovery from Pauli measurements. In J. Shawe-Taylor, R. S. Zemel, P. L. Bartlett, F. Pereira, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 24*, pages 1638–1646. Curran Associates, Inc., 2011. URL: http://papers.nips.cc/paper/4222-universal-low-rank-matrix-recovery-from-pauli-measurements.pdf.

33 William Matthews, Stephanie Wehner, and Andreas Winter. Distinguishability of quantum states under restricted families of measurements with an application to quantum data hiding. *Comm. Math. Phys.*, 291(3):813–843, 2009. doi:10.1007/s00220-009-0890-5.

34 Ryan O'Donnell and John Wright. Efficient quantum tomography. In Daniel Wichs and Yishay Mansour, editors, *Proceedings of the 48th Annual ACM SIGACT Symposium on Theory of Computing, STOC 2016, Cambridge, MA, USA, June 18-21, 2016*, pages 899–912. ACM, 2016. doi:10.1145/2897518.2897544.

35 John Preskill. Quantum Computing in the NISQ era and beyond. *Quantum*, 2:79, 2018. doi:10.22331/q-2018-08-06-79.

36 Carlos A. Riofrio, David Gross, Steven T. Flammia, Thomas Monz, Daniel Nigg, Rainer Blatt, and Jens Eisert. Experimental quantum compressed sensing for a seven-qubit system. *Nat. Commun.*, 8(1), 2017. doi:10.1038/ncomms15305.

37 Takanori Sugiyama, Peter S. Turner, and Mio Murao. Precision-guaranteed quantum tomography. *Phys. Rev. Lett.*, 111:160406, 2013. doi:10.1103/PhysRevLett.111.160406.

38 Koji Tsuda, Gunnar Rätsch, and Manfred K. Warmuth. Matrix exponentiated gradient updates for on-line learning and Bregman projection. *J. Mach. Learn. Res.*, 6:995–1018, 2005. URL: http://jmlr.org/papers/v6/tsuda05a.html.

39 Joran van Apeldoorn, András Gilyén, Sander Gribling, and Ronald de Wolf. Quantum SDP-solvers: Better upper and lower bounds. In Chris Umans, editor, *58th IEEE Annual Symposium on Foundations of Computer Science, FOCS 2017, Berkeley, CA, USA, October 15-17, 2017*, pages 403–414. IEEE Computer Society, 2017. doi:10.1109/FOCS.2017.44.

40 Vladislav Voroninski. Quantum tomography from few full-rank observables. *preprint arXiv:1309.7669*, 2013.

41 Akram Youssry, Christopher Ferrie, and Marco Tomamichel. Efficient online quantum state estimation using a matrix-exponentiated gradient method. *New J. Phys.*, 21(3):033006, 2019. doi:10.1088/1367-2630/ab0438.

Pauli Error Estimation via Population Recovery

Steven T. Flammia ✉
AWS Center for Quantum Computing, Pasadena, CA, USA
IQIM, California Institute of Technology, Pasadena, CA, USA

Ryan O'Donnell ✉
Computer Science Department, Carnegie Mellon University, Pittsburgh, PA, USA

Abstract

Motivated by estimation of quantum noise models, we study the problem of learning a Pauli channel, or more generally the Pauli error rates of an arbitrary channel. By employing a novel reduction to the "Population Recovery" problem, we give an extremely simple algorithm that learns the Pauli error rates of an n-qubit channel to precision ϵ in ℓ_∞ using just $O(1/\epsilon^2)\log(n/\epsilon)$ applications of the channel. This is optimal up to the logarithmic factors. Our algorithm uses only unentangled state preparation and measurements, and the post-measurement classical runtime is just an $O(1/\epsilon)$ factor larger than the measurement data size. It is also impervious to a limited model of measurement noise where heralded measurement failures occur independently with probability $\leq 1/4$.

We then consider the case where the noise channel is close to the identity, meaning that the no-error outcome occurs with probability $1-\eta$. In the regime of small η we extend our algorithm to achieve *multiplicative* precision $1 \pm \epsilon$ (i.e., additive precision $\epsilon\eta$) using just $O(\frac{1}{\epsilon^2\eta})\log(n/\epsilon)$ applications of the channel.

2012 ACM Subject Classification General and reference → Cross-computing tools and techniques; Hardware → Quantum error correction and fault tolerance

Keywords and phrases Pauli channels, population recovery, Goldreich–Levin, sparse recovery, quantum channel tomography

Digital Object Identifier 10.4230/LIPIcs.TQC.2021.8

Funding This work was supported by ARO grant W911NF2110001. R.O. is additionally supported by NSF grant FET-1909310. This material is based upon work supported by the National Science Foundation under grant numbers listed above. Any opinions, findings and conclusions or recommendations expressed in this material are those of the author and do not necessarily reflect the views of the National Science Foundation (NSF).

Acknowledgements We thank Robin Harper for discussions about Pauli channels.

1 Introduction

A major challenge in the analysis of engineered quantum systems is estimating and modeling noise. The most standard theoretical model for noise in the study of quantum error correction and fault tolerance [20] is the n-qubit *Pauli channel*:

$$\rho \mapsto \sum_{C \in \{0,1,2,3\}^n} p(C) \cdot \sigma_C \rho \sigma_C^\dagger. \tag{1}$$

Here $\sigma_C = \sigma_{C_1} \otimes \cdots \otimes \sigma_{C_n}$ is a tensor product of the Pauli operators $\sigma_0, \sigma_1, \sigma_2, \sigma_3$, and p is a probability distribution on $\{0,1,2,3\}^n$. The numbers $p(C)$ are referred to as the *Pauli error rates*. Additional motivation for the Pauli channel model comes from the practical technique of randomized compiling [13, 22], which converts a general noise channel Λ (with potentially coherent errors) to a Pauli channel Λ_P having the same process fidelity as the original channel. We refer to the $p(C)$ values for Λ_P as the "Pauli error rates" of the original general channel Λ.

16th Conference on the Theory of Quantum Computation, Communication and Cryptography (TQC 2021).
Editor: Min-Hsiu Hsieh; Article No. 8; pp. 8:1–8:16
Leibniz International Proceedings in Informatics
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

Given an experimental setup (possibly with randomized compiling), a natural challenge is to diagnose errors in the system via *Pauli error estimation*. Here the goal is to estimate the large Pauli error rates of an unknown channel by preparing states, passing them through the channel, and measuring them. The main desideratum is to minimize the number of measurements; additionally one would like to use simple state preparation and measurement processes and minimal computational overhead. We remark that full tomography for arbitrary n-qubit channels requires at least $4^n/\epsilon^2$ measurements, with more practical methods requiring at least $8^n/\epsilon^2$.

In this work, we give very simple and efficient algorithms for learning all of the large Pauli error rates of an n-qubit channel. Our first main result is the following:

▶ **Theorem 1.** *There is a learning algorithm that, given parameters $0 < \delta, \epsilon < 1$, as well as access to an n-qubit channel with Pauli error rates p, has the following properties:*

- *It prepares $m = O(1/\epsilon^2) \cdot \log(\frac{n}{\epsilon\delta})$ unentangled n-qubit pure states, where each of the mn 1-qubits states is chosen uniformly at random from $\{|0\rangle, |1\rangle, |+\rangle, |-\rangle, |i\rangle, |-i\rangle\}$;*
- *It passes these m states through the Pauli channel.*
- *It performs unentangled measurements on the resulting states, with each qubit being measured in either the $\{|0\rangle, |1\rangle\}$-basis, the $\{|+\rangle, |-\rangle\}$-basis, or $\{|i\rangle, |-i\rangle\}$-basis.*
- *It performs an $O(mn/\epsilon)$-time classical post-processing algorithm on the resulting mn measurement outcome bits.*
- *It outputs hypothesis Pauli error rates $\widehat{p}$ in the form of a list of at most $\frac{4}{\epsilon}$ pairs $(C, \widehat{p}(C))$, with all unlisted $\widehat{p}$ values treated as 0.*

The algorithm's hypothesis $\widehat{p}$ will satisfy $\|\widehat{p} - p\|_\infty \le \epsilon$ except with probability at most δ.

Note that our "sample complexity" of $\widetilde{O}(1/\epsilon^2)$ is optimal up to the logarithmic term: The task of estimating Pauli error rates strictly (and vastly) generalizes the problem of estimating the bias of an unknown coin to additive precision ϵ (and confidence $1-\delta$), and this is known to require $\Theta(1/\epsilon^2) \cdot \log(1/\delta)$ coin flips. For comparison of our bounds with previous work [8, 10, 11], see §1.2.

When the channel is modeling quantum noise, one hopes and expects that the nontrivial error rate, $\eta = 1 - p(0^n)$, is small. In this case, a natural and more ambitious goal is to first estimate η, and then to estimate all other Pauli error rates to *multiplicative* precision $1 \pm \epsilon$; i.e., additive precision $\pm\epsilon\eta$. (This ambition was also pursued in [8, 10].) Here the ideal sample complexity would be $O(\frac{1}{\epsilon^2\eta})$.[1] If one uses our Theorem 1 as a black box, it would use $\widetilde{O}(\frac{1}{\epsilon^2\eta^2})$ measurements. The extra factor of $1/\eta$ here is quite undesirable (as one might imagine a typical parameter setting to be something like $\eta = 10^{-2}$, $\epsilon = 10^{-1}$). We show that it can be eliminated:

▶ **Theorem 2.** *In the setting of Theorem 1, suppose the overall error rate is $\eta = 1 - p(0^n)$. One can augment the algorithm so that, given in addition a "noise floor" parameter $0 < \eta_0 < 1$, it has the following properties:*

- *It first makes at most $m_0 := O(1/\eta_0) \cdot \log(1/\delta)$ measurements (as in Theorem 1).*
- *It does $O(m_0 n)$-time classical processing, then either outputs "$\eta \le \eta_0$" and halts, or proceeds.*

[1] Again, one can compare the task to the vastly simpler one of estimating the face probabilities of a 6-sided die that comes up "1" with probability $1 - \eta$. When rolling many times, one obtains a non-1 outcome roughly every $1/\eta$ rolls. Thus the task becomes very similar to estimating the face probabilities of a 5-sided die to additive precision ϵ, but with a $1/\eta$ "slowdown".

- *It then operates as in Theorem 1, but makes* $m := O(\frac{1}{\epsilon^2 \eta}) \cdot \log(\frac{n}{\epsilon \delta})$ *measurements.*

Its outputs are correct, with a guarantee of $\|\widehat{p} - p\|_\infty \leq \epsilon\eta$*, except with probability at most* δ*.*

Finally, we show that our algorithm can be made impervious to a limited amount of measurement noise. Specifically, suppose that our measuring devices have the following property: When measuring a 1-qubit state from $\{|0\rangle, |1\rangle, |+\rangle, |-\rangle, |i\rangle, |-i\rangle\}$ in one of the bases $\{|0\rangle, |1\rangle\}$, $\{|+\rangle, |-\rangle\}$, or $\{|i\rangle, |-i\rangle\}$, the device fails (reading out "?") with probability ν, and otherwise behaves ideally. We assume that the failures are independent, and that the algorithm may know the parameter ν (thanks to prior estimation). In this case, we will see that it is almost automatic to obtain the following extension:

▶ **Theorem 3.** *Theorem 2 continues to hold for any any constant* $\nu \leq \frac{1}{4}$*.*

For the more challenging task of handling general SPAM (state preparation and measurement) error, see the discussion in §1.2.

1.1 Techniques

Our algorithm employs a novel reduction from Pauli error estimation to the task in classical unsupervised learning known as *Population Recovery.* Population Recovery was introduced by Dvir, Rao, Wigderson, and Yehudayoff in 2012 [7], and has been studied in numerous subsequent works [3, 16, 14, 23, 6, 15, 19, 2, 1, 5, 17]. A Population Recovery problem is specified by a *classical channel* $\mathcal{S}$ – i.e., a stochastic map $\mathcal{S} : \Sigma \to \Gamma$ for some finite alphabets Σ, Γ. The task is to learn an unknown probability distribution p on Σ^n to ℓ_∞-error ϵ, with the twist being that samples are mediated by the channel. That is, when the learner requests a sample, first $x \in \Sigma^n$ is drawn according to p, but then only $y = \Sigma(x_1)\Sigma(x_2)\cdots\Sigma(x_n)$ is revealed to the learner. The most well-studied cases are the binary symmetric channel and the binary erasure channel, the former being noticeably more challenging; lately, the deletion channel has also begun to be studied. (Each of these channels also requires specifying the crossover/erasure/deletion probability r.)

Our work shows how to efficiently convert the Pauli error estimation task to that of Population Recovery with respect to the so-called *binary Z-channel* with crossover probability $\frac{1}{3}$. This is the channel with $\Sigma = \Gamma = \{\mathtt{0}, \mathtt{1}\}$ in which 0's are "transmitted" correctly, but 1's are flipped to 0 with probability $\frac{1}{3}$. We observe that the known methods for Population Recovery with respect to the binary erasure channel with erasure probability r also apply equally well to the Z-channel with crossover probability r. We then use the fact that there is a known, highly efficient Population Recovery algorithm for erasures with probability at most $\frac{1}{2}$. [7, 16, 5, 19] (Indeed, the fact that even probability $\frac{1}{2}$ can be tolerated is the reason our Pauli error estimation algorithm can handle additional measurement noise as in Theorem 3.)

1.2 Previous work

The problem of Pauli error estimation was first studied in depth in work of the first author and Wallman [8]. It is not possible to directly compare those results with ours, for several reasons. The most immediate reason is that their complexity bounds typically include a factor of $\widetilde{O}(1/\Delta)$, where "Δ" is another parameter, the spectral gap of the Pauli channel being learned. We have $\Delta \leq 2\eta$, where $\eta = 1 - p(0^n)$ is the nontrivial error rate, and this is saturated in the most favorable case. However, in general Δ may be arbitrarily small, or even zero, for relatively simple channels. In practice, a user of the algorithm in [8] would set a spectral cutoff Δ_0 and allow estimation errors for channel eigenvalues in the interval

$(1-\Delta_0, 1]$, but no analysis is done in [8] of the extra error incurred by this cutoff. Thus, in the worst case, their results as formally stated do not give any guarantee.

On the other hand, the results of [8] are impervious to a much more challenging model of measurement error ("SPAM"). This model imposes that before the learner measures the channel's output, an *additional* unknown channel Ξ is applied to the state. (It is assumed that Ξ satisfies the extremely mild condition that its nontrivial error rate is bounded away from 1.) It might seem impossible to disentangle Ξ from the main channel Λ to be learned, but the authors of [8] use the fact that one is at liberty to pass a state ρ through Λ several times (say, k times) before it is subjected to Ξ; i.e., the learner may obtain $\Xi\Lambda^k\rho$ for ρ and $k \in \mathbb{N}$ of the learner's choosing. By carefully choosing k values up to $O(1/\Delta)$, the authors of [8] show that Ξ can essentially be expunged. (Note that, in practice, multiple uses of the channel are often far less costly than even a single measurement.)

Finally, the first algorithm in [8] judges its hypothesis with respect to the ℓ_2-norm, rather than the ℓ_∞ norm as in this paper. This distinction is relatively minor, however, as the norms are roughly equivalent for probability distributions: $\|\widehat{p}-p\|_\infty \le \|\widehat{p}-p\|_2 \le \|\widehat{p}-p\|_\infty^{1/2}$, and one may refine this further to take into account dependence on $\eta = 1 - p(0^n)$.

With these caveats, we state (simplifications of) the relevant main results in [8]:

▶ **Theorem 4** ([8]). *There exists a SPAM-tolerant algorithm that makes $\widetilde{O}(2^n \log(1/\Delta))/\epsilon^2$ measurements, with $O(1/\Delta)$ channel-uses per measurement, and with high probability outputs an estimate $\widehat{p}$ of the channel's Pauli error rates p satisfying $\|\widehat{p}-p\|_2 \le \epsilon\eta$.*

In the favorable case of $\Delta = \Theta(\eta)$, this is somewhat comparable to our Theorem 2; the above theorem has much better SPAM-tolerance, but a complexity that is greater by roughly 2^n.

The authors of [8] also present a heuristic for identifying a set S corresponding to large Pauli error rates with the following guarantee.

▶ **Theorem 5** ([8]). *For any set $S \subseteq \{0,1,2,3\}^n$, there exists a SPAM-tolerant algorithm that makes $\widetilde{O}(\log|S|)\log\log(1/\Delta)/\epsilon^4$ measurements, with $O(1/\Delta)$ channel-uses per measurement, and with high probability outputs estimates $\widehat{p}(C)$ for each $C \in S$ satisfying $|\widehat{p}(C) - p(C)| \le \epsilon\eta$.*

However, no guarantee is proven that the set S will contain the $|S|$ largest error rates.

The results in [11] are also somewhat incomparable to the present paper. The authors analyze Pauli channels with a recovery guarantee in the ∞-norm, but under the assumption that the Pauli channel has sparse and random support, and that the nonzero error rates are not too small (greater than some fixed ϵ_0). While the sparsity assumption is not critical in that analysis (the algorithm will approximate error rates smaller than ϵ_0 as zero with high probability), the random support assumption is used in an essential way. This is an undesirable assumption since it is very unlikely to hold in practice.[2] The sample complexity is also not stated directly in terms of quantum measurements, but rather in terms of queries to a "noisy eigenvalue oracle" with Gaussian noise. While this noisy oracle can be approximated by quantum measurements and finite sample complexity, quantum noise is not exactly Gaussian, so no direct comparison with the present work is possible without further analysis.

We remark that the techniques used in [8, 11] are Fourier-based, and the heuristic from [8] described above is similar to the Goldreich–Levin learning algorithm [9]. In §7, we give an alternate Fourier-based approach to Pauli error estimation, one that is equivalent to our

[2] Perhaps surprisingly, the algorithm performs well on real data despite grossly violating this assumption [11].

Population Recovery method "in disguise"; in fact, the Goldreich–Levin algorithm becomes equivalent to the Individual-to-Population Recovery reduction!

It is our belief that these Fourier techniques can actually be used to provide a common generalization of the results of this paper and of [8]; i.e., efficient SPAM-tolerant Pauli error estimation with no dependence on Δ. We leave this for future work.

2 Notation

▶ **Notation 6.** The 1-*qubit Pauli matrices* are the unitary, hermitian matrices

$$\sigma_0 = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, \quad \sigma_1 = \sigma_x = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \quad \sigma_2 = \sigma_y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}, \quad \sigma_3 = \sigma_z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}.$$

As operators on the Bloch sphere, $\sigma_1, \sigma_2, \sigma_3$ act as rotations by π about the 1st, 2nd, 3rd axis (aka x-, y-, z-axis), respectively. More generally, an *n-qubit Pauli matrix*, indexed by string $A \in \{0,1,2,3\}^n$, is $\sigma_A = \bigotimes_{j=1}^n \sigma_{A_j}$.

▶ **Notation 7.** For $a, b \in \{0,1,2,3\}$, there is some $c \in \{0,1,2,3\}$ such that $\sigma_a\sigma_b = \sigma_c$, up to a global phase. We introduce the notation $a \oplus b$ (equivalently, $b \oplus a$) for this c; so, e.g, $1 \oplus 3 = 2$, $0 \oplus b = b$, etc. We extend the notation coordinate-wise: if $A, B \in \{0,1,2,3\}^n$, then $A \oplus B = (A_1 \oplus B_1, \ldots, A_n \oplus B_n) \in \{0,1,2,3\}^n$ (and so $\sigma_A\sigma_B = \sigma_{A\oplus B}$, up to a global phase).

▶ **Notation 8.** We write the orthonormal eigenbasis for the Pauli operator σ_x as $|\chi^1_+\rangle, |\chi^1_-\rangle$. On the Bloch sphere these are the two unit vectors pointing in the positive (respectively, negative) direction along the 1st (x-)axis; they are often called $|+\rangle$, $|-\rangle$. We use similar notation $|\chi^2_+\rangle, |\chi^2_-\rangle$ (often called $|i\rangle, |-i\rangle$) and $|\chi^3_+\rangle, |\chi^3_-\rangle$ (often called $|0\rangle, |1\rangle$) for σ_2 and σ_3.

▶ **Notation 9.** For $a, b \in \{0,1,2,3\}$ we have that $\sigma_b|\chi^a_+\rangle$ is (up to a phase) $|\chi^a_\pm\rangle$, with the subscript being $+$ if σ_a and σ_b commute, and $-$ if σ_a and σ_b anticommute. To capture this, it will be convenient to introduce the following notation:

$$a \star b = b \star a = \begin{cases} \mathtt{0} & \text{if } |\{a, b, a \oplus b\}| < 3, \text{ i.e., } \sigma_a, \sigma_b \text{ commute;} \\ \mathtt{1} & \text{if } |\{a, b, a \oplus b\}| = 3, \text{ i.e., } \sigma_a, \sigma_b \text{ anticommute.} \end{cases}$$

Thus $\sigma_b|\chi^a_+\rangle = |\chi^a_{(-1)^{a\star b}}\rangle$ (up to a phase). We extend this notation coordinate-wise, writing $A \star B = (A_1 \star B_1, \ldots, A_n \star B_n) \in \{\mathtt{0}, \mathtt{1}\}^n$ for $A, B \in \{0,1,2,3\}^n$. For example, $(0,0,3,2,1) \star (3,1,1,2,2) = (\mathtt{0},\mathtt{0},\mathtt{1},\mathtt{0},\mathtt{1})$.

▶ **Fact 10.** If we identify $\{0,1,2,3\}$ with $\mathbb{F}_2^2$ by writing numbers in base 2, then $\oplus$ corresponds to the usual vector addition in $\mathbb{F}_2^2$, and $\star$ corresponds to the "symplectic" product: $a \star b = (a_1, a_2) \star (b_1, b_2) = a_1b_2 + a_2b_1$. This lets us see that $a \star (b \oplus c) = (a \star b) + (a \star c) \bmod 2$.

▶ **Notation 11.** For a quantity x, we denote an estimate of x by $\widehat{x}$. We use boldface font (e.g., $\boldsymbol{A}$) to denote a random variable. If $\boldsymbol{A}$ is drawn from the distribution p we denote this by $\boldsymbol{A} \sim p$, and let A denote a concrete assignment to the variable $\boldsymbol{A}$. Addition (of scalars or vectors) modulo 2 is denoted $+_2$. The Fourier transform of f is denoted $\tilde{f}$.

3 Learning a Pauli channel

In this section we describe the basic setup for learning a Pauli channel. Learning the Pauli error rates of a general channel will end up being just a minor extension, discussed in §6.1.

As described in Equation (1), an n-qubit Pauli channel is determined by a probability distribution p on $\{0,1,2,3\}^n$. This probability distribution induces the mixed unitary channel in which σ_C is applied with probability $p(C)$. An $n = 5$ example:

$$p(00321) = 2/10,\ p(01300) = 3/10,\ p(11323) = 2/6,\ p(30000) = 1/6,\ p(C) = 0 \text{ otherwise.}$$

We anthropomorphize by imagining a character Charlie who operates the channel; on receiving a state ρ, Charlie first (secretly) draws $\boldsymbol{C} \sim p$, then outputs the state $\sigma_{\boldsymbol{C}}\rho\sigma_{\boldsymbol{C}}$.

Alice the Learner would like to estimate the probability distribution p via interactions with Charlie. Alice has the ability to prepare n-qubit states, to "query" Charlie (i.e., pass an n-bit state through his channel), and to measure states that she receives back. Her goal is to learn a precise approximation to p (with high probability), while minimizing the number of queries to Charlie.

▶ **Definition 12.** *We say that Alice performs a* nontrivial probe *if she does the following:*

- *She chooses a string $A \in \{1,2,3\}^n$.*
- *She prepares the (unentangled) n-qubit state $|\psi_A\rangle$ in which the jth qubit is $|\chi_+^{A_j}\rangle$.*
- *She passes $|\psi_A\rangle$ through Charlie, obtaining $\sigma_C|\psi_A\rangle$ with probability $p(C)$.*
- *She does a (non-entangled) measurement on the resulting n-qubit state, measuring the jth qubit in the basis $|\chi_\pm^{A_j}\rangle$.*

Continuing our $n = 5$ example, if Alice does a nontrivial probe with the string $A = 31122$, this entails preparing and passing to Charlie the state

$$|\psi_{31123}\rangle = |\chi_+^3\rangle|\chi_+^1\rangle|\chi_+^1\rangle|\chi_+^2\rangle|\chi_+^2\rangle \quad \Big(= |0\rangle|+\rangle|+\rangle|i\rangle|i\rangle\Big),$$

and then measuring the 5 returned qubits in the bases $|\chi_\pm^3\rangle$, $|\chi_\pm^1\rangle$, $|\chi_\pm^1\rangle$, $|\chi_\pm^2\rangle$, $|\chi_\pm^2\rangle$, respectively.

Now suppose that Charlie drew $C = 00321$ (which occurs with probability 2/10 in our example). Then the state returned to Alice would be

$$\begin{aligned}(\sigma_0 \otimes \sigma_0 \otimes \sigma_3 \otimes \sigma_2 \otimes \sigma_1)|\psi_{31122}\rangle &= (\sigma_0|\chi_+^3\rangle) \otimes (\sigma_0|\chi_+^1\rangle) \otimes (\sigma_3|\chi_+^1\rangle) \otimes (\sigma_2|\chi_+^2\rangle) \otimes (\sigma_1|\chi_+^2\rangle)\\ &= e^{i\theta} \cdot |\chi_+^3\rangle|\chi_+^1\rangle|\chi_-^1\rangle|\chi_+^2\rangle|\chi_-^2\rangle\end{aligned}$$

for some phase $e^{i\theta}$ ($\theta \in \mathbb{R}$) that we did not bother to compute. Now when Alice measures in the bases $|\chi_\pm^3\rangle$, $|\chi_\pm^1\rangle$, $|\chi_\pm^1\rangle$, $|\chi_\pm^2\rangle$, $|\chi_\pm^2\rangle$, her readout will, with probability 1, be

$$|\chi_+^3\rangle|\chi_+^1\rangle|\chi_-^1\rangle|\chi_+^2\rangle|\chi_-^2\rangle.$$

The subscripts $+,+,-,+,-$ here are the 5 bits of information conveyed to Alice by the readout, and we may think of instead labeling them as $\mathtt{00101}$ in accordance with Notation 9. With this relabeling convention, we obtain:

▶ **Fact 13.** Suppose Alice performs a nontrivial probe with string $A \in \{1,2,3\}^n$, and suppose the random string drawn by Charlie is $C \in \{0,1,2,3\}^n$. Then when Alice measures, she obtains the readout $R = A \star C \in \{\mathtt{0},\mathtt{1}\}^n$.

▶ Remark 14. So far we have pictured Alice as first choosing A, and then Charlie as drawing a random C. It is useful now to make a slight shift in perspective: for each interaction between Alice and Charlie, we will equivalently think of *Charlie* as first (secretly) drawing C, and then Alice gaining some partial information about this C by "probing" it using an A of her choice. We emphasize that Alice must make her choice of A without knowing the channel outcome C.

We now describe a trick that Alice may employ in probing the channel:

▶ **Definition 15.** *For a channel distribution p on $\{0,1,2,3\}^n$, and any fixed $B \in \{0,1,2,3\}^n$, define the B-*altered channel distribution *$p^{\oplus B}$ on $\{0,1,2,3\}^n$ via $p^{\oplus B}(C) = p(B \oplus C)$.*

For any string $B \in \{0,1,2,3\}^n$ of her choosing, Alice can effectively simulate access to the B-altered channel: If she wishes to simulate passing $|\phi\rangle$ through the B-altered channel, she could instead simply pass $\sigma_B|\phi\rangle$ through Charlie's actual channel. (This may introduce a "wrong" global phase, but it doesn't matter for any measurement behavior that we consider here.) But in fact, something even simpler is true:

▶ **Observation 16.** *Given $B \in \{0,1,2,3\}^n$, if Alice wants to perform a nontrivial probe of $p^{\oplus B}$ based on string A, she can pass $|\psi_A\rangle$ to Charlie as always. Then, when she measures and obtains $A \star C$, she can "reinterpret" this readout by adding in, mod 2, the string $A \star B \in \{\mathtt{0},\mathtt{1}\}^n$ (which she knows). Recalling Fact 10, this gives her $(A \star B) +_2 (A \star C) = A \star (B \oplus C)$. Thus the reinterpreted readout is indeed distributed as what she would get by probing $p^{\oplus B}$ with A.*

A natural strategy for Alice is to make *random* nontrivial probes. It is easy to see the following:

▶ **Fact 17.** Fix a draw $C \in \{0,1,2,3\}^n$ for Charlie. Now if Alice performs a nontrivial probe with a uniformly random $\boldsymbol{A} \in \{1,2,3\}^n$, then the coordinates of her readout $\boldsymbol{R} = \boldsymbol{A} \star C \in \{\mathtt{0},\mathtt{1}\}^n$ will be independent, with the following distribution for each $1 \le j \le n$:

- If $C_j = 0$ then $\boldsymbol{R}_j$ will be $\mathtt{0}$ with probability 1.
- If $C_j \neq 0$ then $\boldsymbol{R}_j$ will be $\mathtt{0}$ with probability $\frac{1}{3}$ and $\mathtt{1}$ with probability $\frac{2}{3}$.

We can state this more succinctly by introducing some additional terminology:

▶ **Notation 18.** For $B, C \in \{0,1,2,3\}^n$, define the string $C^{\neq B} \in \{\mathtt{0},\mathtt{1}\}^n$ by

$$(C^{\neq B})_j = \begin{cases} \mathtt{1} & \text{if } C_j \neq B_j, \\ \mathtt{0} & \text{if } C_j = B_j. \end{cases}$$

▶ **Definition 19.** *Recall from information theory the so-called Z-*channel with crossover probability *r: it is the binary channel that leaves $\mathtt{0}$ untouched and flips $\mathtt{1}$ to $\mathtt{0}$ with probability r.*

Now Fact 17 can be restated as follows:

▶ **Fact 20.** Fix a draw $C \in \{0,1,2,3\}^n$ for Charlie. Now if Alice performs a random nontrivial probe, her readout is the result of passing $C^{\neq 0^n}$ through a Z-channel with crossover probability $\frac{1}{3}$.

▶ **Observation 21.** *By combining Observation 16 with Fact 20, we obtain the following: Fix a draw $C \in \{0,1,2,3\}^n$ for Charlie and suppose Alice performs a random nontrivial probe. She can then – for any fixed $B \in \{0,1,2,3\}^n$ – interpret her readout as $C^{\neq B}$ passed through a Z-channel with crossover probability $\frac{1}{3}$.* Warning: *these reinterpretations are completely* dependent*; she of course cannot get the result of* independent *channel applications for various B's, unless she makes multiple probes.*

4 Population Recovery

With Observation 21 in hand, we have effectively reduced the problem of learning a Pauli channel to a "Population Recovery"-type problem (with a quantum-free definition). To recap:

there is an unknown probability distribution p on $\{0,1,2,3\}^n$, a learner may request samples, and when a sample C is drawn from p, the learner receives a binary string which can be interpreted as "$C^{\neq B}$ passed through a Z-channel with crossover $\frac{1}{3}$" for any $B \in \{0,1,2,3\}^n$ of the learner's choosing.

In this section we will give a solution to this problem that has optimal sample complexity (except possibly up to a logarithmic factor) using techniques from the field of Population Recovery. Our solution will immediately imply Theorem 1 in the special case where the channel to be learned is indeed a Pauli channel. The case of learning a *general* channel's Pauli error rates is treated in §6.1. We remark that our Pauli channel algorithm only uses nontrivial probes, and thus only involves preparing the states $|0\rangle$, $|+\rangle$, and $|i\rangle$. The other three states $|1\rangle$, $|-\rangle$, and $|-i\rangle$ are only used for the extension to general channels.

4.0.0.1 Idea of our solution

Using known techniques from Population Recovery, one can first reduce to the simpler task of "Individual Recovery" (estimating a single $p(B)$ value) via a coordinate-by-coordinate learning algorithm. Then one can further reduce to just recovering $p(0^n)$, using the altered-channel trick. As for learning $p(0^n)$, we first observe that the replacement of C by $C^{\neq 0^n}$ changes nothing for this problem, so we effectively have the same task just for the $\frac{1}{3}$-crossover Z-channel on binary strings. This is similar to the erasure channel with erasure probability $\frac{1}{3}$, and in fact the known solutions for erasure probability-r [7, 16, 5, 19] *only use the locations of the* 1*'s in the received word.* Thus these known solutions work *equally well* for the Z-channel. Indeed, as noted in [7], the solution is particularly simple when $r \le \frac{1}{2}$ (as it is for us); the full method of "robust local inverses" is not needed, and one can use the "natural inverse" (as we implicitly do in the proof of Theorem 22 below).

4.1 Individual Recovery

Although the proof of the below theorem is self-contained, we remark that it implicitly follows the Individual Recovery routine of [7] for the $\frac{1}{3}$-erasure channel.

▶ **Theorem 22.** *For any fixed $B \in \{0,1,2,3\}^n$, a version of Theorem 1 holds in which the learner only computes an estimate $\widehat{p}(B)$ of $p(B)$ satisfying $|\widehat{p}(B) - p(B)| \le \epsilon_0$ except with probability at most δ_0. The number of samples used is $m = O(1/\epsilon_0^2) \cdot \log(1/\delta_0)$ and the classical post-processing time is $O(mn)$.*

▶ Remark 23. The reader may wish to verify the proof just in the case $B = 0^n$, where it is simpler; the general case then follows from Observation 16.

Proof. Alice obtains m probe/readout pairs $(\boldsymbol{A}, \boldsymbol{R})$, with $\boldsymbol{A} \sim \{1,2,3\}^n$ uniformly random and $\boldsymbol{R} = \boldsymbol{A} \star \boldsymbol{C}$, where $\boldsymbol{C}$ is a random channel outcome drawn from p. The estimate $\widehat{p}(B)$ that Alice will output is the empirical mean of the random variable

$$\boldsymbol{H} = (-1/2)^{|\boldsymbol{A}\star B +_2 \boldsymbol{R}|} = (-1/2)^{\sum_t ((\boldsymbol{A}\star B)+_2\boldsymbol{R})_t} = \prod_{t=1}^{n} (-1/2)^{\boldsymbol{y}_t}, \quad \boldsymbol{y}_t := (\boldsymbol{A}_t \star B_t) +_2 \boldsymbol{R}_t.$$

As seen in Observation 21, for a given outcome $\boldsymbol{C} = C$, the random binary string $(\boldsymbol{A}\star B) +_2 \boldsymbol{R}$ is distributed as $C^{\neq B}$ passed through a Z-channel with crossover probability $\frac{1}{3}$. In particular, its coordinates $\boldsymbol{y}_t$ are independent random variables, with conditional expectation given by

$$\mathbf{E}[(-1/2)^{\boldsymbol{y}_t} \mid \boldsymbol{C} = C] = \begin{cases} (-1/2)^0 = 1 & \text{if } C_t = B_t, \\ \frac{1}{3}(-1/2)^0 + \frac{2}{3}(-1/2)^1 = 0 & \text{if } C_t \neq B_t. \end{cases}$$

Thus

$$\mathbf{E}[\boldsymbol{H} \mid \boldsymbol{C} = C] = \prod_{t=1}^{n} \mathbf{E}\big[(-1/2)^{\boldsymbol{y}_t} \mid \boldsymbol{C} = C\big] = \begin{cases} 1 & \text{if } C = B, \\ 0 & \text{if } C \neq B, \end{cases}$$

and hence indeed $\mathbf{E}[\boldsymbol{H}] = p(B)$. ◀

4.2 Population Recovery

Theorem 22 allows Alice to estimate $p(B)$ for any particular string $B \in \{0,1,2,3\}^n$. But also, for any shorter string $\beta \in \{0,1,2,3\}^\ell$, Alice can estimate the marginal

$$p(\beta) \coloneqq \sum_{\gamma \in \{0,1,2,3\}^{n-\ell}} p(\beta\gamma) = \Pr_{\boldsymbol{C} \sim p}[(\boldsymbol{C}_1, \dots, \boldsymbol{C}_\ell) = \beta],$$

simply by ignoring all data in positions $\ell + 1, \dots, n$. (She is obviously not limited to marginalizing contiguous blocks, but this is all we will need for our purposes.) Alice can thus learn all of p to good ℓ_∞-precision with the straightforward, coordinate-by-coordinate branch-and-prune approach common in Population Recovery (see, e.g., [19, App. A]). We repeat this approach here; the following algorithm achieves our main Theorem 1 for Pauli channels, except for the claim about the running time of the post-processing algorithm:

```
1. Set $\epsilon_0 = \frac{\epsilon}{4}$, $\delta_0 = \frac{4\epsilon\delta}{9n}$ and draw a single batch of $m$ samples, where $m$ is as in Theorem 22.
2. Define "support sets" $\Omega_1 = \{0,1,2,3\}$ and $\Omega_2 = \cdots = \Omega_n = \emptyset$.
3. For round $j = 1 \dots n-1$:
4.   For each prefix $\beta' \in \Omega_j$ and each $b \in \{0,1,2,3\}$:
5.     Run the Individual Recovery algorithm on $\beta \coloneqq \beta' b$ to estimate the marginal $p(\beta)$.
6.     If the estimate is at least $2\epsilon_0 = \frac{\epsilon}{2}$, then place $\beta$ into $\Omega_{j+1}$.
7. Output as $\widehat{p}$ the collection of strings in $\Omega_n$, together with their estimated probabilities.
```

The correctness of the algorithm, that $\|\widehat{p} - p\|_\infty \leq \epsilon$ with failure probability at most δ, is straightforward and is explicitly proven in [19, Lem. 18]. The proof also establishes that when there is no failure, $|\Omega_j| \leq \frac{4}{\epsilon}$ holds for all $1 \leq j \leq n$. Thus for running time purposes (and without impacting the correctness claim) we may have the algorithm abort if ever some Ω_j gets cardinality more than $\frac{4}{\epsilon}$. It only remains to obtain the post-processing running time of $O(mn/\epsilon)$ claimed in Theorem 1.

4.2.0.1 Running time analysis

As it stands, the running time of the above algorithm is $O(mn^2/\epsilon)$, since it may do up to $O(n/\epsilon)$ executions of the $O(mn)$-time Individual Recovery algorithm. But since all executions of the Individual Recovery algorithm are on the same batch of samples, it's not hard to see that information from the jth round of the algorithm can be used to speed up the $(j+1)$st round. More precisely, we show that each round can be done in $O(m/\epsilon)$ time, leading to the overall claimed running time of $O(mn/\epsilon)$.

Let $R \in \{\mathtt{0},\mathtt{1}\}^{m \times n}$ be the measurement outcome bits that the algorithm processes, and let $R_{1\dots j}$ denote the submatrix formed by the first j columns. Also, for $\beta \in \{0,1,2,3\}^j$, let $R^{(\beta)} \in \{\mathtt{0},\mathtt{1}\}^{m \times j}$ be the (hypothetical) matrix whose tth row is the same as $R_{1\dots j}$'s but with $(A_1^t, \dots, A_j^t) \star \beta$ added in mod 2, where A^t is the tth probe string used by Alice. Given β, the algorithm can look up entries of $R^{(\beta)}$ in $O(1)$ time.

Recall that when the algorithm does Individual Recovery on the prefix β, it computes the fraction of rows of $R^{(\beta)}$ that have Hamming weight i, multiplies this number by $(-1/2)^i$, and sums the results. In particular, this estimate can be computed in $O(m)$ time given the vector $h^{(\beta)} \in \mathbb{N}^m$ whose tth entry is the Hamming weight of the tth row of $R^{(\beta)}$ – just add up $(-1/2)^{h_t^{(\beta)}}/m$ across all t.

We can now modify the above Population Recovery algorithm so that whenever a prefix $\beta \in \{0,1,2,3\}^j$ is added into Ω_j, the algorithm retains the vector $h^{(\beta)}$ that went into estimating $p(\beta)$. It is easy to see that in the subsequent round, we can compute each of $h^{(\beta 0)}, h^{(\beta 1)}, h^{(\beta 2)}, h^{(\beta 3)}$ from $h^{(\beta)}$ (and hence the marginal estimates) in $O(m)$ time, and retain them as needed. Thus indeed each round only requires $O(m/\epsilon)$ time, since at most $\frac{4}{\epsilon}$ prefixes are processed in each round.

5 Multiplicative error

In a practical scenario we would would hope that the "nontrivial error rate" of the Pauli channel,

$$\eta \coloneqq 1 - p(0^n)$$

is very small. This motivates writing p as a mixture distribution, as follows:

$$p: \quad \text{mixing weight } 1-\eta \text{ on } 0^n, \quad \text{mixing weight } \eta \text{ on } p_{\text{err}}, \tag{2}$$

where p_{err} is a distribution on $\{0,1,2,3\}^n \setminus \{0^n\}$. Now a natural goal is to learn with *multiplicative error* ϵ, meaning producing estimates $\widehat{\eta}$, $\widehat{p}_{\text{err}}$ with

$$(1-\epsilon)\eta \leq \widehat{\eta} \leq (1+\epsilon)\eta, \qquad \|\widehat{p}_{\text{err}} - p_{\text{err}}\|_\infty \leq \epsilon.$$

As described in §1, the ideal sample complexity to strive for now is $O(\frac{1}{\epsilon^2 \eta})$.

5.0.0.1 Adaptivity, and a floor on η

Let us make two more technical remarks. First, if η is extraordinarily small (or even 0), we won't want to make $1/\eta$ measurements. Thus we assume the algorithm is given a floor η_0, and when $\eta \leq \eta_0$ we are satisfied just to certify that this is the case. Second, we cannot hope to have (as before) a completely nonadaptive algorithm achieving sample complexity on the order of $1/(\epsilon^2\eta)$ because the algorithm does not know η, or even an approximation to η, in advance. Thus our algorithm will first need to find a preliminary constant-factor approximation η_{est} to η in an online probe-and-measure fashion; then it can proceed nonadaptively.

5.1 Roughly estimating the error rate

Here we describe the (mildly) "adaptive" algorithm that handles the error floor and obtains η_{est}, a factor-5 approximation of η before subsequently finding a good approximation to all the error rates (including $p(0^n) = 1 - \eta$).

▶ **Lemma 24.** *There is a randomized learning algorithm that, given input $0 < \delta_0, \eta_0 < 1$, as well as access to an n-qubit Pauli channel defined by distribution p with nontrivial error rate $\eta = 1 - p(0^n)$:*

- *repeatedly prepares a state, passes it through the Pauli channel, and measures, as in Theorem 1;*

- *halts after some number of repetitions (*always *at most $O(1/\eta_0) \cdot \log(1/\delta_0)$) and outputs either: "$\eta \leq \eta_0$" or else an estimate η_{est} that is within a factor of 5 of η;*
- *runs in classical time that is linear in the number of measurement readouts.*

Except with probability at most δ_0, the algorithm's output is correct and it halts after at most $O(1/\eta) \cdot \log(1/\delta_0)$ repetitions.

Proof. Recall Fact 20: by doing random nontrivial probes, an algorithm can get samples from a random string that is non-0^n with some probability η' between $\frac{2}{3}\eta$ and η. In order to find the factor-5 approximation η_{est} of η, it suffices for the algorithm to estimate η' up to a factor of 3 or else certify $\eta' \leq \eta_0$. This is now a completely standard problem: estimating the bias of an η'-biased coin up to a factor of 3 using on the order of $1/\eta'$ flips, despite not knowing η' in advance. The algorithm is the obvious one: repeatedly flip until getting "heads" (but never more than $O(1/\eta_0)$ times), convert the number of flips $\boldsymbol{G}$ into the estimate $1/\boldsymbol{G}$, then take the median of $O(\log(1/\delta))$ estimates. We omit the straightforward classical analysis. ◀

5.2 Individual Recovery with multiplicative error

We henceforth assume the algorithm from Lemma 24 succeeded and that η_{est} is a factor-5 approximation of the true error rate η. We now describe how the algorithm can do "Individual Recovery" with multiplicative error. A note: the sample complexities are stated in terms of the parameter η; formally, the algorithm does not know η, but it can use $5\eta_{\text{est}}$ (which it knows) in its place, and the $O(\cdot)$ bounds are not affected.

We first show that the algorithm from Theorem 22 already achieves the desired multiplicative-error/sample tradeoff in the case of estimating η:

▶ **Proposition 25.** *Given η_{est} within a factor 5 of $\eta = 1 - p(0^n)$, a version of Theorem 22 holds in which, for $B = 0^n$, the estimate $\widehat{p}(0^n)$ satisfies $|\widehat{p}(0^n) - p(0^n)| \leq \epsilon\eta$ except with failure probability at most δ_0, and the number of samples used is $m = O(\frac{1}{\epsilon^2\eta}) \cdot \log(1/\delta_0)$.*

▶ Remark 26. The success event here is equivalent to the estimate $\widehat{\eta} = 1 - \widehat{p}(0^n)$ satisfying the inequality $(1-\epsilon)\eta \leq \widehat{\eta} \leq (1+\epsilon)\eta$.

Proof. The algorithm used is the same as the one in Theorem 22 (with $B = 0^n$); only the analysis changes. Recall that the algorithm's estimate is the empirical mean of $\boldsymbol{H} = (-1/2)^{|\boldsymbol{R}|}$, a random variable whose true mean is $p(0^n) = 1 - \eta$. Equivalently we may consider the random variable $\overline{\boldsymbol{H}} = 1 - \boldsymbol{H}$, which has true mean η and which is supported in $[0, 2]$. But now a standard multiplicative Chernoff bound shows that the empirical mean $\widehat{\eta}$ of $\overline{\boldsymbol{H}}$ after $O(1/(\epsilon^2\eta)) \cdot \log(1/\delta_0)$ samples indeed satisfies $(1-\epsilon)\eta \leq \widehat{\eta} \leq (1+\epsilon)\eta$. ◀

▶ **Proposition 27.** *A trivial modification of Theorem 22 also achieves, for any $B \neq 0^n$, an estimate $\widehat{p}(B)$ satisfying $|\widehat{p}(B) - p(B)| \leq \epsilon\eta$ except with failure probability at most δ_0, using $m = O(\frac{1}{\epsilon^2\eta}) \cdot \log(1/\delta_0)$ samples.*

Proof. Rather than empirically estimating the mean of $\boldsymbol{H} = (-1/2)^{|\boldsymbol{A}\star B +_2 \boldsymbol{R}|}$, the algorithm instead empirically estimates the mean of $\boldsymbol{H}' = \boldsymbol{H} - (-1/2)^{|\boldsymbol{A}\star B|}$, a random variable bounded in $[-2, 2]$. (Note that Alice knows B and also each probe string $\boldsymbol{A}$, hence can compute $(-1/2)^{|\boldsymbol{A}\star B|}$ herself.) It is easy to see that $\mathbf{E}[(-1/2)^{|\boldsymbol{A}\star B|}] = 0$ using $B \neq 0^n$. Thus $\boldsymbol{H}'$ remains an unbiased estimator for $p(B)$; i.e., $\mathbf{E}[\boldsymbol{H}'] = p(B)$. But furthermore note that $\boldsymbol{H}'$ is almost always 0; specifically, whenever the channel outcome $\boldsymbol{C}$ is 0^n (probability $1-\eta$), we have $\boldsymbol{R} = \boldsymbol{A} \star 0^n = 0^n$ and hence $\boldsymbol{H}' = (-1/2)^{|\boldsymbol{A}\star B|} - (-1/2)^{|\boldsymbol{A}\star B|} = 0$. Thus using $|\boldsymbol{H}'| \leq 2$

we trivially conclude $\mathbf{E}[(\boldsymbol{H}')^2] \leq 4\eta$. But now it follows from the Bernstein inequality (see, e.g., [21, Ch. 2, Prop. 2.4]) that to estimate the mean of a random variable $\boldsymbol{H}'$ that is bounded in $[-2, 2]$ and has $\mathbf{E}[(\boldsymbol{H}')^2] = s$, it suffices to use $\frac{s+2\gamma/3}{\gamma^2} \ln(2/\delta_0)$ samples to achieve additive error γ except with probability at most δ_0. Thus taking $\gamma = \epsilon\eta$ and using $s \leq 4\eta$ indeed completes the proof. ◀

5.3 Population Recovery with multiplicative error

Combining the results from the previous section on Individual Recovery with the reduction in §4.2 immediately proves our Theorem 2 (in the case of Pauli channels).

6 Further extensions: general channels and measurement noise

6.1 Pauli error rates of general channels

With very minor effort we can now upgrade our algorithm to learn the "Pauli error rates" of a *general* quantum channel, thereby fully establishing our Theorem 1.

We recall the following definitions/facts (see, e.g., [4, Lem. 5.2.4]):

▶ **Definition 28.** *Let Λ denote an arbitrary n-qubit quantum channel. Its* Pauli twirl Λ_P *is the n-qubit quantum channel defined by*

$$\Lambda_P \rho = \mathop{\mathbf{E}}_{\boldsymbol{T} \sim \{0,1,2,3\}^n} [\sigma_{\boldsymbol{T}}^\dagger (\Lambda \sigma_{\boldsymbol{T}} \rho \sigma_{\boldsymbol{T}}^\dagger) \sigma_{\boldsymbol{T}}].$$

The channel Λ_P is itself a Pauli channel; the associated probabilities $p(C)$ are called the Pauli error rates *of Λ.*

▶ **Fact 29.** Suppose we write K_j for the Kraus operators of Λ, so $\Lambda\rho = \sum_j K_j \rho K_j^\dagger$. Further suppose that K_j is represented in the Pauli basis as $K_j = \sum_{C \in \{0,1,2,3\}^n} \alpha_{j,C} \sigma_C$. Then Λ's Pauli error rates are given by $p(C) = \sum_j |\alpha_{j,C}|^2$.

It's easy to see that, given access to a general channel Λ, a learner Alice can simulate access to its Pauli twirl Λ_P: whenever Alice wishes to pass ρ through Λ_P, she instead chooses $\boldsymbol{T} \sim \{0, 1, 2, 3\}^n$ uniformly at random, passes $\sigma_{\boldsymbol{T}} \rho \sigma_{\boldsymbol{T}}^\dagger$ through Λ, and replaces the channel output τ with $\sigma_{\boldsymbol{T}}^\dagger \tau \sigma_{\boldsymbol{T}}$.

In our context of learning Pauli error rates, this simulation becomes particularly simple. Recall that our algorithm for Pauli channels only ever passes pure states of the form $|\chi_+^{A_1}\rangle|\chi_+^{A_2}\rangle \cdots |\chi_+^{A_n}\rangle$ through the channel, for $A \in \{1, 2, 3\}^n$. Further, the channel output is always measured in the associated Pauli bases, the jth qubit of the output measured in the basis $|\chi_\pm^{A_n}\rangle$. The effect of simulating the Pauli twirl with $\sigma_{\boldsymbol{T}}$ is simply to replace the input $|\chi_+^{A_j}\rangle$ to qubit j with the input $|\chi_{(-1)^{A_j \star \boldsymbol{T}_j}}^{A_j}\rangle$, and to add $A \star \boldsymbol{T}$ to the measurement outcomes. Thus we may deduce the full version of Theorem 2 (concerning learning Pauli error rates of general channels) from the already-established special case of learning Pauli channels.

6.2 Tolerating measurement errors

It is also straightforward to see that our algorithm can tolerate a mild form of measurement error. Suppose that we have an imperfect 1-qubit measuring device that is used to implement the three Pauli-basis measurements. More precisely, we assume it has the following property: When applied to a qubit in a Pauli eigenvalue state, the measuring device "fails" (say, reads out "?") with probability ν, and otherwise behaves ideally. Here ν is a parameter that we assume is known to the learner through estimation, and that measurement failures are independent events.

As discussed in the paragraph just preceding §4.1, our algorithm for estimating any $p(B)$ is effectively performing the standard "Individual Recovery algorithm" for the *binary erasure channel* with erasure probability $\frac{1}{3}$. (Recall that we actually have the Z-channel with crossover probability $\frac{1}{3}$ applied to the binary string $C^{\neq B}$, but that the erasure channel algorithm only uses the locations of the $\mathtt{1}$'s in the received string, and thus works equally well for the Z-channel.) The effect of measuring device failures is to replace the erasure probability $\frac{1}{3}$ with $r \coloneqq \nu + (1-\nu)\frac{1}{3}$. So long as $r \leq \frac{1}{2}$, the standard recovery algorithm for probability r-erasures works just as well [7]: the only change needed is that the factor "$(-1/2)$" appearing in Theorem 22's definition of $\boldsymbol{H}$ needs to be replaced by $-r/(1-r)$. (Note that this quantity has magnitude bounded by 1 if and only if $r \leq \frac{1}{2}$.) But the condition $r \leq \frac{1}{2}$ is equivalent to $\nu \leq \frac{1}{4}$, and this justifies our Theorem 3.

(In fact, for erasure probability $\frac{1}{2} < r < 1$, much more sophisticated algorithms [5, 19] can succeed at Individual Recovery, at the expense of increasing the sample complexity from the order of $1/\epsilon^2$ to the order of $1/\epsilon^{2r/(1-r)}$; but for simplicity, we ignore pursuing this extension.)

7 An alternative, Fourier approach

Here we give an alternative algorithm for learning Pauli channels, using a perspective from Boolean Fourier analysis; see [18, Chaps. 1, 3] for background and notation.

For Pauli channels, the $\mathbb{F}_2$-Fourier transform relates the error rates and the channel eigenvalues. The Pauli operators themselves are the eigenvectors of a Pauli channel, and we can easily compute the eigenvalue associated to σ_A using the relation $\sigma_A\sigma_C = (-1)^{A\star C}\sigma_C\sigma_A$ via

$$\sigma_A \mapsto \sum_{C\in\{0,1,2,3\}^n} p(C)\cdot \sigma_C\sigma_A\sigma_C^\dagger = \sum_{C\in\{0,1,2,3\}^n} p(C)\cdot(-1)^{\sum_{i=1}^n (A\star C)_i}\sigma_A = \lambda_A\sigma_A,$$

so that $\lambda_A = \mathbf{E}_{\boldsymbol{C}\sim\{0,1,2,3\}^n}\left[2^{2n}p(\boldsymbol{C})\cdot(-1)^{\sum_{i=1}^n(A\star\boldsymbol{C})_i}\right]$. This clearly resembles an $\mathbb{F}_2$-Fourier transform.

To make this connection more explicit, in the remainder of this section we will identify the elements of $\{0,1,2,3\}$ with their base-2 representations in $\mathbb{F}_2^2$. Let us use overline to denote the swapping operation on two bits; i.e., $\overline{a_1a_2} = a_2a_1$ for $a_1, a_2 \in \mathbb{F}_2$. We extend the notation n-fold to vectors $A \in \mathbb{F}_2^{2n} \cong (\mathbb{F}_2^2)^n$. (Equivalently, we have $\overline{0}=0$, $\overline{1}=2$, $\overline{2}=1$, $\overline{3}=3$, and we extend the notation coordinate-wise to $A\in\{0,1,2,3\}^n$.) Now define the *symplectic dot product*

$$\langle A, C\rangle = \overline{A}\cdot C = \sum_{i=1}^n (A\star C)_i \mod 2,$$

where $A\cdot C$ denotes the usual dot product on $\mathbb{F}_2^{2n}$. A Pauli channel eigenvalue is now equivalently written in two ways as

$$\lambda_A = \mathop{\mathbf{E}}_{\boldsymbol{C}\sim\mathbb{F}_2^{2n}}\left[2^{2n}p(\boldsymbol{C})\cdot(-1)^{\langle A,\boldsymbol{C}\rangle}\right] = \mathop{\mathbf{E}}_{\boldsymbol{C}\sim\mathbb{F}_2^{2n}}\left[2^{2n}p(\boldsymbol{C})\cdot(-1)^{\overline{A}\cdot\boldsymbol{C}}\right].$$

Let us write φ for the probability *density* (vis-a-vis the uniform distribution) associated to p; i.e., $\varphi(C) = 2^{2n}p(C)$. Then the Fourier transform $f = \tilde{\varphi}$ is given by

$$f(A) = \tilde{\varphi}(A) = \mathop{\mathbf{E}}_{\boldsymbol{C}\sim\mathbb{F}_2^{2n}}[\varphi(\boldsymbol{C})(-1)^{A\cdot\boldsymbol{C}}] = \mathop{\mathbf{E}}_{\boldsymbol{C}\sim p}[(-1)^{A\cdot\boldsymbol{C}}] = \lambda_{\overline{A}}. \tag{3}$$

Observe that f (and equivalently λ) are functions $f : \mathbb{F}_2^{2n} \to [-1, 1]$ and that $p = \tilde{f}$. Such group character averages were considered in the context of quantum noise estimation in [12]. While we can talk interchangeably about the Fourier coefficients of the density φ and the channel eigenvalues λ (as they are related by $f(A) = \lambda_{\overline{A}}$), we will focus on f in what follows.

We see from Equation (3) that

$$f(A) = \mathop{\mathbf{E}}_{\boldsymbol{C}\sim p}[(-1)^{\langle \overline{A}, C\rangle}] = \mathop{\mathbf{E}}_{\boldsymbol{C}\sim p}[(-1)^{\sum_t (\overline{A}\star C)_t}], \tag{4}$$

and as we now describe this means Alice can straightforwardly estimate $f(A)$ for any A of her choosing.

Let's extend Definition 12 of "nontrivial probe" to allow not just for $A \in \{1, 2, 3\}^n$ but any $A \in \{0, 1, 2, 3\}^n$; we omit the adjective "nontrivial" in this more general case. To handle coordinates j where $A_j = 0$, Alice can simply put any qubit $|\chi\rangle$ into the jth position of her state $|\psi_A\rangle$, ignore the jth position coming out of the channel, and automatically treat the jth readout bit as $\mathtt{0}$. In this way, Fact 13 still holds: for any probe $A \in \{0, 1, 2, 3\}^n$ and any string $C \in \{0, 1, 2, 3\}^n$ drawn by Charlie, the readout is $R = A \star C \in \{\mathtt{0}, \mathtt{1}\}^n$. It follows that Alice can empirically estimate the right-hand side of Equation (4) by repeatedly probing the channel with $\overline{A}$ and averaging the following function of $\boldsymbol{R}$, the readout: $(-1)^{\sum_t \boldsymbol{R}_t}$. This yields $f(A)$ to additive precision ϵ with confidence at least $1 - \delta$, using $O(1/\epsilon^2) \cdot \log(1/\delta)$ probes; we refer to this as "efficient estimation".

We now see that Alice has (noisy) query access to $f : \mathbb{F}_2^{2n} \to [-1, 1]$, and her goal is to estimate the large values of $p = \tilde{f}$. This task is highly reminiscent of the task solved by the Goldreich–Levin learning algorithm [9]. The minor differences are that Goldreich–Levin typically assumes *perfect* query access to some $f : \mathbb{F}_2^2 \to \{-1, 1\}$, and has the normalization that $\sum_C \tilde{f}(C)^2 = 1$, rather than our normalization of $\sum_C \tilde{f}(C) = \sum_C p(C) = 1$. Still, if one "unrolls" the Goldreich–Levin algorithm in this context, one gets almost the same solution for learning Pauli channels as described in §4.2: reduction from Population Recovery to Individual Recovery.

7.1 The Goldreich–Levin approach

In a typical exposition of the Goldreich–Levin algorithm (e.g. [18, Ch. 3.5], which we'll follow), one assumes Alice has perfect query access to an $f : \mathbb{F}_2^n \to \{-1, 1\}$. Herein we sketch the alterations to this exposition that are needed for learning Pauli channels.

One basic subroutine in the Goldreich–Levin algorithm (akin to "Individual Recovery") is using query access to f to efficiently estimate $\tilde{f}(B)$ for various B. This is done (see [18, Prop. 3.30]) via straightforward empirical estimation:

$$\tilde{f}(B) = \mathop{\mathbf{E}}_{\boldsymbol{A}\sim\mathbb{F}_2^{2n}}[f(\boldsymbol{A})(-1)^{\boldsymbol{A}\cdot B}]. \tag{5}$$

Recall that in our setting, Alice can only access $f(\boldsymbol{A})$ by empirically estimating it via Equation (4). Inserting this into the above, we get

$$\tilde{f}(B) = \mathop{\mathbf{E}}_{\boldsymbol{A}\sim\mathbb{F}_2^{2n}}\mathop{\mathbf{E}}_{\boldsymbol{C}\sim p}[(-1)^{\sum_t (\overline{\boldsymbol{A}}\star\boldsymbol{C})_t + \boldsymbol{A}\cdot B}].$$

Thus as needed in Goldreich–Levin, Alice can efficiently estimate this for any B of her choosing by picking uniformly random $\boldsymbol{A} \in \{0, 1, 2, 3\}^n$, probing the channel with $\overline{\boldsymbol{A}}$, and averaging the following function of $\boldsymbol{R}$, the readout: $(-1)^{\sum_t \boldsymbol{R}_t + \boldsymbol{A}\cdot B}$. Indeed the reader will note that this method is almost the same as the one used in Theorem 22! The essential difference is that $\boldsymbol{A}$ is uniform on $\{0,1,2,3\}^n$ rather than $\{1,2,3\}^n$, which effectively makes the "crossover probability" $\frac{1}{2}$ instead of $\frac{1}{3}$, and hence the factor $(-1/2) = -\frac{1/3}{1-1/3}$ becomes $(-1) = -\frac{1/2}{1-1/2}$. Note that this difference implies that the Goldreich–Levin approach does not immediately tolerate measurement failures as in §6.2.

As mentioned earlier, Goldreich–Levin typically assumes $f : \mathbb{F}_2^n \to \{-1,1\}$ and hence we have $\sum_{C \in \mathbb{F}_2^n} \tilde{f}(C)^2 = 1$; its goal is to find all B with $|\tilde{f}(B)| \geq \epsilon$, knowing that there are automatically at most $1/\epsilon^2$ such B. It accomplishes this via a "branch-and-prune" strategy that relies on the ability to estimate $\sum_{C' \in \mathbb{F}_2^{n-k}} \tilde{f}(\beta, C')^2$ for any prefix $\beta \in \mathbb{F}_2^k$. In our setup, with $f : \mathbb{F}_2^{2n} \to [-1,1]$, we instead know a priori that $p = \tilde{f}$ satisfies $\sum_C \tilde{f}(C) = 1$, and our goal is to find all B with $|\tilde{f}(B)| \geq \epsilon$. Thus the search is even easier than in Goldreich–Levin, as the same branch-and-prune strategy works with non-squared Fourier coefficients. Following the strategy gives the same Population-to-Individual Recovery algorithm as in §4.2.

7.2 Final remarks

As mentioned earlier, the techniques used in the previous works [8, 10, 11] on Pauli channel estimation are Fourier-based. The paper [8] achieves SPAM tolerance, and manages to trade some measurement complexity for channel-reuse; on the other hand, its bounds have a dependency on the channel eigenvalue gap $\Delta = \min_{A \neq 0^n}\{1 - |\lambda_A|\}$, which may be arbitrarily small. As shown in the previous section, one can recover our (SPAM-less) Pauli estimation results via the Fourier approach with no dependence on Δ and without assumptions about the noise or the support.

We believe that it is possible to obtain a common generalization of the results in [8] and the present paper that achieves the best of both worlds via this Fourier approach: SPAM-robust and efficient Pauli channel estimation with no dependence on Δ. We leave this for future work.

References

1 Frank Ban, Xi Chen, Adam Freilich, Rocco Servedio, and Sandip Sinha. Beyond trace reconstruction: Population recovery from the deletion channel. In *Proceedings of the 60th Annual IEEE Symposium on Foundations of Computer Science*, pages 745–768, 2019.

2 Frank Ban, Xi Chen, Rocco A. Servedio, and Sandip Sinha. Efficient Average-Case Population Recovery in the Presence of Insertions and Deletions. In Dimitris Achlioptas and László A. Végh, editors, *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques (APPROX/RANDOM 2019)*, volume 145 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 44:1–44:18, Dagstuhl, Germany, 2019. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik.

3 Lucia Batman, Russell Impagliazzo, Cody Murray, and Ramamohan Paturi. Finding heavy hitters from lossy or noisy data. In *Proceedings of the 16th Annual International Conference on Approximation Algorithms for Combinatorial Optimization Problems*, pages 347–362, 2013.

4 Christoph Dankert. *Efficient Simulation of Random Quantum States and Operators.* PhD thesis, University of Waterloo, 2015.

5 Anindya De, Ryan O'Donnell, and Rocco Servedio. Sharp bounds for population recovery. *Theory of Computing*, 16(6):1–20, 2020.

6 Anindya De, Michael Saks, and Sijian Tang. Noisy population recovery in polynomial time. In *Proceedings of the 57th Annual IEEE Symposium on Foundations of Computer Science*, pages 675–684, 2016.

7 Zeev Dvir, Anup Rao, Avi Wigderson, and Amir Yehudayoff. Restriction access. In *Proceedings of the 3nd Annual Innovations in Theoretical Computer Science*, pages 19–33, 2012.

8 Steven Flammia and Joel Wallman. Efficient estimation of Pauli channels. *ACM Transactions on Quantum Computing*, 1(1):1–32, 2020.

9 Oded Goldreich and Leonid Levin. A hard-core predicate for all one-way functions. In *Proceedings of the 21st Annual ACM Symposium on Theory of Computing*, pages 25–32, 1989.

10 Robin Harper, Steven T. Flammia, and Joel J. Wallman. Efficient learning of quantum noise. *Nature Physics*, 16(12):1184–1188, August 2020.

11 Robin Harper, Wenjun Yu, and Steven T. Flammia. Fast estimation of sparse quantum noise. *PRX Quantum*, 2(1):010322, February 2021.

12 Jonas Helsen, Xiao Xue, Lieven M. K. Vandersypen, and Stephanie Wehner. A new class of efficient randomized benchmarking protocols. *npj Quantum Information*, 5(1):71, 2019.

13 E. Knill. Quantum computing with realistically noisy devices. *Nature*, 434(7029):39–44, March 2005.

14 Shachar Lovett and Jiapeng Zhang. Improved noisy population recovery, and reverse Bonami–Beckner inequality for sparse functions. In *Proceedings of the 47th Annual ACM Symposium on Theory of Computing*, pages 137–142, 2015.

15 Shachar Lovett and Jiapeng Zhang. Noisy population recovery from unknown noise. In *Conference on Learning Theory*, pages 1417–1431, 2017.

16 Ankur Moitra and Michael Saks. A polynomial time algorithm for lossy population recovery. In *Proceedings of the 54th Annual IEEE Symposium on Foundations of Computer Science*, pages 110–116, 2013.

17 Shyam Narayanan. Improved algorithms for population recovery from the deletion channel. In *Proceedings of the 2021 ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 1259–1278. Society for Industrial and Applied Mathematics, 2021.

18 Ryan O'Donnell. *Analysis of Boolean functions.* Cambridge University Press, 2014.

19 Yury Polyanskiy, Ananda Theertha Suresh, and Yihong Wu. Sample complexity of population recovery. In Satyen Kale and Ohad Shamir, editors, *Proceedings of the 2017 Conference on Learning Theory*, volume 65 of *Proceedings of Machine Learning Research*, pages 1589–1618, Amsterdam, Netherlands, July 2017. PMLR.

20 Barbara Terhal. Quantum error correction for quantum memories. *Reviews of Modern Physics*, 87(2):307, 2015.

21 Martin Wainwright. *High-dimensional statistics: A non-asymptotic viewpoint.* Cambridge University Press, 2019.

22 Joel Wallman and Joseph Emerson. Noise tailoring for scalable quantum computation via randomized compiling. *Physical Review A*, 94(5):052325, 2016.

23 Avi Wigderson and Amir Yehudayoff. Population recovery and partial identification. *Machine Learning*, 102(1):29–56, 2016.

Quantum Probability Oracles & Multidimensional Amplitude Estimation

Joran van Apeldoorn ✉

Institute for Information Law, University of Amsterdam, The Netherlands
QuSoft, Centrum Wiskunde & Informatica, Amsterdam, The Netherlands

Abstract

We give a multidimensional version of amplitude estimation. Let p be an n-dimensional probability distribution which can be sampled from using a quantum circuit U_p. We show that all coordinates of p can be estimated up to error ε per coordinate using $\widetilde{\mathcal{O}}\left(\frac{1}{\varepsilon}\right)$ applications of U_p and its inverse. This generalizes the normal amplitude estimation algorithm, which solves the problem for $n = 2$. Our results also imply a $\widetilde{\mathcal{O}}\left(n/\varepsilon\right)$ query algorithm for ℓ_1-norm (the total variation distance) estimation and a $\widetilde{\mathcal{O}}\left(\sqrt{n}/\varepsilon\right)$ query algorithm for ℓ_2-norm. We also show that these results are optimal up to logarithmic factors.

2012 ACM Subject Classification Theory of computation → Design and analysis of algorithms; Theory of computation → Quantum computation theory

Keywords and phrases quantum algorithms, amplitude estimation, monte carlo

Digital Object Identifier 10.4230/LIPIcs.TQC.2021.9

Funding *Joran van Apeldoorn*: Supported by the Dutch Research Council (NWO/OCW), as part of the Quantum Software Consortium programme (project number 024.003.037).

Acknowledgements I would like to thank Ronald de Wolf, András Gilyén, Ashley Montanaro, and Srinivasan Arunachalam for insightful discussions and comments. I would also like to thank the TQC2021 reviewers for their useful comments.

1 Introduction

A central challenge when working with random processes is the estimating of the probability of some event occurring from a bunch of samples. An example from classical computer science is Monte Carlo methods, which try and estimate a value that is hard to compute using a random sampling process. To estimate the probability p of an event occurring using classical samples we can simply sample many times and use the fraction of the outcomes where the event occurred as our estimate. It follows from the Chernoff bound that $\mathcal{O}\left(\frac{\ln(1/\delta)}{\varepsilon^2}\right)$ samples suffice to get an ε accurate estimate with failure probability at most δ. In fact, it can be shown that this is optimal for classical samples [3].

If we however have access to "quantum samples", that is a unitary that prepares a state that upon measuring would return 1 with probability p, than we can improve the number of "samples" needed. The *amplitude estimation* algorithm by Brassard et al. [2] show that $\mathcal{O}\left(\frac{\ln(1/\delta)}{\varepsilon}\right)$ applications of the unitary and it's inverse suffice. This already lays the ground work for numerous general speedups, including for many Monte Carlo methods [9].

Sometimes estimating a single probability is not enough, and we are actually interested in finding a full (discrete) probability distribution. We write $\Delta^n := \{x \in \mathbb{R}^n : x \geq 0 \wedge \|x\|_1 = 1\}$ for the set of all probability distributions on n elements. Let $p \in \Delta^n$, if we take $\mathcal{O}\left(\frac{\ln(n/\delta)}{\varepsilon^2}\right)$ classical samples than for each element p_i we get an estimate $\tilde{p}_i$ such that $|p_i - \tilde{p}_i| \leq \varepsilon$ with error probability at most δ/n. Hence by the union bound over all $i \in n$ it follows that $\|p - \tilde{p}\|_\infty \leq \varepsilon$ with probability at least $1 - \delta$.

16th Conference on the Theory of Quantum Computation, Communication and Cryptography (TQC 2021).
Editor: Min-Hsiu Hsieh; Article No. 9; pp. 9:1–9:11
Leibniz International Proceedings in Informatics
LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

This paper considers the problem of recovering an estimate for a distribution $p \in \Delta^n$ using "quantum samples":

▶ **Definition 1** (Quantum probability oracle). *Let $p \in \Delta^n$ be a probability distribution. We say that O_p is a* quantum probability oracle *for p if*

$$O_p \left|0\right\rangle = \sum_{i=1}^{n} \sqrt{p_i} \left|i\right\rangle\!\left|\psi_i\right\rangle$$

for some quantum states $\left|\psi_1\right\rangle, \dots, \left|\psi_n\right\rangle$. That is, applying O_p to the $\left|0\right\rangle$ state and measuring the first register is the same as sampling from p.

Throughout the paper we will assume that if we can apply O_p, then we can also apply O_p^{-1}, and we can do both in a controlled way. Note that this is the case if O_p comes from a randomized classical or quantum algorithm.

We generalize the result of amplitude estimation to n-dimensional distributions, showing that an ε-ℓ_∞-estimate can be obtained with $\widetilde{\mathcal{O}}\left(\frac{1}{\varepsilon}\right)$ queries to a quantum probability oracle. We do so using a multidimensional version of quantum phase estimation, in a similar manner as the quantum gradient estimation algorithm by Jordan [6, 4]. In fact, we consider estimating the gradient of the function $f(x) = \langle x, p\rangle$.

We also consider ℓ_1-norm (or total variation distance) and ℓ_2-norm estimates. We get $\widetilde{\mathcal{O}}\left(\frac{n}{\varepsilon}\right)$ and $\widetilde{\mathcal{O}}\left(\frac{\sqrt{n}}{\varepsilon}\right)$ query algorithms respectively using norm equivalence. In the second part of the paper we give lower bounds that matches the upper bounds up to logarithmic factors for ℓ_1-norm and ℓ_2-norm. An ℓ_∞-norm lower bound follows from known lower bounds on amplitude estimation. We end the paper with some open questions.

Table 1 Comparison of known classical sampling bounds and our quantum results for estimating a distribution $p \in \Delta^n$ up to ε error in a certain norm. Here the $\widetilde{\Theta}(\cdots)$ hides polylogarithmic factors in n and $1/\varepsilon$. *The ℓ_2-norm quantum lower bound only holds when $\varepsilon < 1/(3\sqrt{n})$.

	Known Classical	Quantum
ℓ_∞	$\widetilde{\Theta}\left(\frac{1}{\varepsilon^2}\right)$ LB:[3] UB:Chernoff	$\widetilde{\Theta}\left(\frac{1}{\varepsilon}\right)$ LB: [1][1] UB: Theorem 9
ℓ_2	$\widetilde{\Theta}\left(\frac{1}{\varepsilon^2}\right)$ LB:[3] UB:[7]	$\widetilde{\mathcal{O}}\left(\min(\frac{\sqrt{n}}{\varepsilon}, \frac{1}{\varepsilon^2})\right)$, $\Omega\left(\frac{\sqrt{n}}{\varepsilon}\right)^\star$ LB: Corollary 12 UB: Corollary 10
ℓ_1	$\widetilde{\mathcal{O}}\left(\frac{n}{\varepsilon^2}\right)$ UB:[7]	$\widetilde{\Theta}\left(\frac{n}{\varepsilon}\right)$ LB: Lemma 11 UB: Corollary 10

2 Upper bound

We show our main result in two steps. First we prove the base result, Theorem 5, which has an almost optimal query complexity but lacks in a few other areas. We then add several improvements to obtain our main result, Theorem 9.

[1] A lower bound on normal amplitude estimation follows from the lower bound on parity given in [1].

2.1 Main algorithm

In this section we will show how to obtain an ε-ℓ_∞-approximation of $p \in \Delta^n$ using $\mathcal{O}\left(\frac{\ln(n)}{\varepsilon}\right)$ queries to a quantum probability oracle for p. To do so we consider the linear function $f : [0,1]^n \to [0,1] : x \mapsto \langle x, p\rangle$. We will show how to construct a specific type of oracle for this function, use known results to convert this to a phase roacle for the function, and then apply multidimensional phase estimation to obtain the gradient p.

We will use the following two oracle definitions:

▶ **Definition 2** (Oracles for functions). *Let $f : D \to [0,1]$ be a $[0,1]$ valued function from a discrete domain D. A probability oracle for the function f is a unitary U_f that acts as*

$$U_f\,|x\rangle|0\rangle|0\rangle = |x\rangle\left(\sqrt{f(x)}\,|1\rangle|\psi_1^x\rangle + \sqrt{1-f(x)}\,|0\rangle|\psi_0^x\rangle\right).$$

A phase oracle for the function f is a unitary U_f that acts as

$$U_f\,|x\rangle = e^{\mathbf{i}f(x)}\,|x\rangle\,.$$

We start by constructing a probability oracle for $f(x) = \langle x, p\rangle$

▶ **Lemma 3.** *Let U_p be a quantum probability oracle for a distribution $p \in \Delta^n$. Let $k \geq 1$ be an integer and let $D = \left\{0, \frac{1}{2^k}, \ldots, \frac{2^k-1}{2^k}\right\}$ be a discretization of $[0,1]$. Then a probability oracle $U_{\tilde{f}}$ for a function $\tilde{f}$ can be constructed such that $\tilde{f}$ is an additive μ-approximation of $f(x) : D^n \to [0,1] : x \mapsto \langle x, p\rangle$ using 2 queries to U_p and $\widetilde{\mathcal{O}}\left(n\mathrm{polylog}\left(1/\mu\right)\right)$ two-qubit gates. The gate count can be improved to* $\mathrm{polylog}\left(n/\mu\right)$ *when the input is stored in a QRAM*[2].

Proof. We start in a state $|x\rangle|0\rangle|0\rangle|0\rangle|0\rangle$. First we apply $U_f \otimes I$ to obtain

$$|x\rangle\left(\sum_{i=1}^{n}\sqrt{p_i}\,|i\rangle|\psi_i\rangle\right)|0\rangle|0\rangle\,.$$

Now, for each $i \in [n]$ we do the following conditioned on i being in the second register:

1. In the last register, compute an approximation of $2\arcsin\left(\sqrt{x_i}\right)/\pi$ such that the approximation is in $[0,1)$.
2. Conditioned on the first bit of the approximation rotate the second to last register from $|0\rangle$ to $|1\rangle$ over an angle $\pi/4$.
3. Continue for the other bits: rotate over an angle $\pi/8$ conditioned on the second bit, then $\pi/16$, and so on.
4. Uncompute the last register.

Note that we can approximate the arcsin very efficiently, only introducing a logarithmic overhead in terms of the precision. In the end the second to last register will be rotated over an angle very close to $2\arcsin\left(x_i\right)/\pi \times \pi/2 = \arcsin\left(x_i\right)$. We finish the analysis as if the angle was exact. We end up with (after dropping the last register which is now $|0\rangle$ again)

$$|x\rangle\sum_{i=1}^{n}\sqrt{p_i}\,|i\rangle|\psi_i\rangle\left(\sqrt{x_i}\,|1\rangle + \sqrt{1-x_i}\,|0\rangle\right) = |x\rangle\sum_{i=1}^{n}\sqrt{p_i x_i}\,|i\rangle|\psi_i\rangle|1\rangle + \ldots|0\rangle\,.$$

[2] A QRAM allows us to store values in such a way that we can recover them conditioned on an index register using a single QRAM query. While a physical QRAM requires many gates to build, the implementation can likely be highly parallel in a similar manner to classical RAM. When we consider a model with a QRAM we abstract the details of the QRAM away, and count a QRAM query as a single gate, similar to how a classical RAM query is normally counted as a single operation for a classical computer.

The ℓ_2-norm of the $|1\rangle$ part of te state is $\sqrt{\sum_{i=1}^{n} \sqrt{p_i x_i}^2} = \sqrt{\langle x, p\rangle}$. We conclude that the state can be written as

$$\sqrt{\langle x, p\rangle}\, |x\rangle |\psi_{x,0}\rangle |0\rangle + \sqrt{1 - \langle x, p\rangle}\, |x\rangle |\psi_{x,1}\rangle |1\rangle\,,$$

and hence we have implemented a probability oracle for f.

The steps taken for each i can be performed at the same time when x is stored in a QRAM, as this allows us to query x_i in superposition on $|i\rangle$. ◀

For our purposes we will require a phase oracle, not a probability oracle. Luckily, in [4] it was shown that a phase oracle can be constructed from a probability oracle with minimal overhead:

▶ **Lemma 4.** *[4, Corollary 4.1 (Rephrased)] Let U_f be a probability oracle for a function $f : D \to [0, 1]$ acting on q qubits. Let $T > 0$. An phase oracle with η-additive error for $T \cdot f(x)$ can be constructed using $\mathcal{O}(|T| + \log(1/\eta))$ applications of U_f and its inverse, and $\mathcal{O}(q|T| + q\log(1/\eta))$ two-qubit gates.*

We could directly apply quantum gradient calculation [6, 4] now to obtain p, but since we have a linear function the result can be obtained using a slightly simpler proof, so we include it for completeness.

▶ **Theorem 5.** *Let $p \in \Delta^n$ and let U_p be a quantum probability oracle acting on q qubits for p. Let $\varepsilon > 0$. An approximation $\tilde{p}$ such that $\|p - \tilde{p}\|_\infty \leq \varepsilon$ can be found with error probability at most δ using $\mathcal{O}(\ln(n/\delta)/\varepsilon)$ applications of U_p and $\widetilde{\mathcal{O}}(\ln(\delta)qn/\varepsilon)$ two-qubit gates. The gatecount can be improved to $\widetilde{\mathcal{O}}(\ln(\delta)q(n + 1/\varepsilon))$ by using a QRAM.*

Proof. Let $k = \lceil \log(4/\varepsilon) \rceil$. Consider the following algorithm:

1. Start in a n-register all zero state, where each register as k qubits:

$$|0^k\rangle \dots |0^k\rangle$$

2. Apply Hadamard gates to all qubits to obtain

$$\bigotimes_{i=1}^{n} \left(\frac{1}{\sqrt{2^k}} \sum_{x_i=0}^{2^k-1} |x_i\rangle \right) = \frac{1}{2^{kn/2}} \sum_{x \in \{0, 2^k-1\}^n} |x\rangle$$

3. Make a phase query for an 1/6-approximation of $f(x) = \langle x, p\rangle$ using Lemma 3 with $\mu < 1/(96\varepsilon)$ and Lemma 4 with $T = 2^k$ and $\eta \leq 1/12$ to obtain a state 1/6-close in ℓ_2-norm to

$$\begin{aligned} \frac{1}{2^{kn/2}} \sum_{x \in \{0, 2^k-1\}^n} e^{\mathbf{i}\langle x, p\rangle} |x\rangle &= \frac{1}{2^{kn/2}} \sum_{x \in \{0, 2^k-1\}^n} e^{\mathbf{i}\sum_i x_i p_i} |x\rangle \\ &= \frac{1}{2^{kn/2}} \sum_{x \in \{0, 2^k-1\}^n} \left(\prod_{i=1}^{n} e^{\mathbf{i}x_i p_i} \right) |x\rangle \\ &= \bigotimes_{i=1}^{n} \left(\frac{1}{\sqrt{2^k}} \sum_{x_i=0}^{2^k-1} e^{\mathbf{i}x_i p_i} |x_i\rangle \right) \end{aligned}$$

4. Apply the k-qubit inverse QFT to each of the n registers and measure each register.

Note that this algorithm applies U_p a total of $\mathcal{O}\left(\frac{1}{\varepsilon}\right)$ times as per Lemma 4. The gate cost of the phase oracle implementation is $\widetilde{\mathcal{O}}\left(qn/\varepsilon\right)$ (or $\widetilde{\mathcal{O}}\left(q\ln(n)/\varepsilon\right)$ when using QRAM), and the n inverse QFTs require $\mathcal{O}\left(n\ln^2(1/\varepsilon)\right)$ gates[3].

If we ignore the ℓ_2-error due to the imperfect phase oracle than it would follow from the analysis of phase estimation that we end up with a vector $\tilde{p}$ such that $|p_i - \tilde{p}_i| \leq 4/2^k \leq \varepsilon$ with probability at least $5/6$ per coordinate. Since we incurred at most $1/6$-ℓ_2-norm error we conclude that $|p_i - \tilde{p}_i| \leq \varepsilon$ with probability at least $2/3$ per coordinate. By repeating $\mathcal{O}\left(\ln(n/\delta)\right)$ times and taking the coordinate wise median, the error probability can be reduced to δ/n. Taking the union bound we get the result from the theorem. ◀

2.2 Improvements and tweaks

In this section we give three improvements on the main algorithm. We start by removing the dependence on n, leaving only the implicit dependence via q in the gate-complexity. We then show how to get a better query bound when only considering part of a distribution. Finally we show that the algorithm can be tweaked to always return an estimate from Δ^n.

2.2.1 Removing the dependence on n

While the main algorithm requires few queries, the time complexity grows linear in n. Since the classical algorithm has no dependence linear dependence on n we would hope the same for the quantum algorithm.

The high gate count in the quantum algorithm is due to the fact that we consider all coordinates of p, even those with very small or 0 entries. However, to get an ε-ℓ_∞-approximation we can ignore all coordinates where the probability is less than ε. This leaves at most $1/\varepsilon$ coordinates to run the algorithm on. To find relevant coordinates we simply use classical samples:

▶ **Lemma 6.** *Let $p \in \Delta^n$, and $\varepsilon, \delta \in (0, 1/3)$. $\mathcal{O}\left(\ln(n/\delta)/\varepsilon\right)$ classical samples suffice to, with error probability at most δ, find all $i \in [n]$ such that $p_i \geq \varepsilon$.*

Proof. Consider a single entry i such that $p_i \geq \varepsilon$. After T samples the probability that we have not seen i yet is at most $(1-\varepsilon)^T$. Letting $T = \frac{\ln(\delta\varepsilon)}{\ln(1-\varepsilon)} = \mathcal{O}\left(\frac{\ln(1/(\delta\varepsilon))}{\varepsilon}\right)$ ensures that this error probability is at most $\delta\varepsilon$. Union bounding over the at most $1/\varepsilon$ coordinates gives the result from the lemma. ◀

The lemma shows that the number of coordinates we have to consider in our main algorithm is independent from n. As we can simply look at the inner product on those entries, we only get a dependence on n implicitly as $q \geq \log(n)$. In fact, the classical algorithm can be improved using the same method.

2.2.2 Learning part of the distribution

Often we will not be interested in all coordinates of p, the method from the previous section is an example, but there might be other cases as well. One example is a binary distribution $(p, 1-p)$, where we only need to estimate the first entry. If we know a $p_{\max}$ such that $p \leq p_{\max}$, then amplitude estimation [2] requires $\mathcal{O}\left(\frac{\sqrt{p_{\max}}}{\varepsilon}\right)$ applications of U_p.

[3] The square can be removed by approximating the QFT using standard techniques.

Similarly, if we know that $p_i \leq p_{\max}$ for all i, then the classical algorithm can be improved by a factor p_{max}. Sadly our main algorithm can not be improved by $\sqrt{p_{max}}$ to our knowledge, but we may get a dependence on the sum of the entries in the part of p that we want to estimate.

▶ **Lemma 7.** *Let $p \in \Delta^n$, $\varepsilon \in (0, 1/3)$ and let $S \subseteq [n]$. Let $p_{mt} \geq \sum_{i \in S} p_i$ be the* $\mathbf{m}$*aximal* $\mathbf{t}$*otal probability on S. We can construct a quantum probability oracle for a distribution $p'' \in \Delta^{n+2}$ using $\mathcal{O}\left(\sqrt{1/p_{mt}}\right)$ applications of U_p, membership queries for S, and two-qubit gates such that a estimating p'' up to $\mathcal{O}(\varepsilon/p_{mt})$-$\ell_\infty$-error gives an ε-ℓ_∞ error estimate of p.*

Proof. The main idea is to amplify the probabilities by a factor of $a = \Theta(1/p_{mt})$ using $\mathcal{O}(\sqrt{a})$ iterations of amplitude amplification. This allows us to take $\varepsilon' = \varepsilon \cdot a$ as a larger error tolerance. However, we need to be careful as we do not know the original ℓ_2 norm of the "good" part of the state, and hence we do not know the exact amplification that $\mathcal{O}(\sqrt{a})$ iterations of amplitude amplification would give, only that it is $\Theta(1/p_{mt})$.

We consider a new distribution p' with dimension $n + 2$. The first n coordinates are equal to $p/2$, while the last to coordinates are $p_{mt}/2$ and $(1 - p_{mt})/2$. We can construct a quantum probability oracle $U_{p'}$ for p' using a single controlled application of U_p.

Using amplitude amplification we can create an quantum probability oracle $U_{p''}$ for a distribution p'' that is equal to ap' on the indices in $S \cup \{n + 1\}$ for some unknown $a \in \Theta(1/p_{mt})$. This requires $\mathcal{O}(\sqrt{a})$ applications of $U_{p'}$ and membership queries for S.

Note that $p''_{n+1} = ap_{mt}/2 = \Theta(1)$, and in particulair let L be a (constant) lower bound so $p''_{n+1} \geq L$. Now, let $\tilde{p}''$ be a $\frac{\varepsilon L}{8p_{mt}}$-$\ell_\infty$-estimate of p''. It follows that $\tilde{p}''_{n+1}$ is an $(1 \pm \frac{\varepsilon}{8p_{mt}})$ multiplicative estimate of p''_{n+1}, and hence it gives such a multiplicative estimate $\tilde{a}$ of a.

Let $\tilde{p}_i = 2\tilde{p}''_i/\tilde{a}$. We know that $\tilde{p}''_i = p''_i + e_1$ for some error term e_1 with $|e_1| \leq L\varepsilon/8p_{mt}$. We also know that $\tilde{a} = a(1 + e_2)$ for some error term e_2 with $|e_2| \leq \varepsilon/8p_{mt}$. Hence we know that

$$\begin{aligned}
\tilde{p}_i &= \frac{2\tilde{p}''_i}{\tilde{a}} \\
&= \frac{2(p''_i + e_1)}{a(1 + e_2)} \\
&= \frac{2(\frac{ap_i}{2} + e_1)}{a(1 + e_2)} \\
&= \frac{p_i + 2e_1/a}{(1 + e_2)} \\
&= (p_i + 2e_1/a)(1 + e_3) \\
&= p_i + 2e_1/a + p_i e_3 + 2e_1 e_3/a
\end{aligned}$$

where $|e_3| \leq 2|e_2| \leq \varepsilon/4p_{mt}$. We can therefore bound the final error by

$$\begin{aligned}
|2e_1/a + p_i e_3 + 2e_1 e_3/a| &\leq |2e_1/a| + |p_i e_3| + |2e_1 e_3/a| \\
&\leq 2\frac{L\varepsilon}{4p_{mt}a} + p_{mt}\frac{\varepsilon}{4p_{mt}} + 2\frac{L\varepsilon^2}{32p_{mt}^2 a} \\
&\leq 2\frac{L\varepsilon}{8L} + \frac{\varepsilon}{4} + 2\frac{L\varepsilon}{64L} \\
&\leq \varepsilon
\end{aligned}$$

Where we used that $\varepsilon \leq p_{mt}$, as otherwise the problem is trivial, as well as $\frac{1}{p_{mt}a} \leq \frac{1}{2L}$. ◀

2.2.3 Returning a probability distribution

Our main algorithm does not always return a $\tilde{p} \in \Delta^n$, all we are promised is that $\|p - \tilde{p}\|_\infty \leq \varepsilon$. The following Lemma shows that we can always convert such a $\tilde{p}$ into a good approximation inside Δ^n.

▶ **Lemma 8.** *Let $p \in \Delta^n$ and let $\tilde{p}$ be such that $\|p - \tilde{p}\|_\infty \leq \varepsilon/8$. Then a $\min(n, 8/\varepsilon)$-sparse $\tilde{p}'$ can be constructed from $\tilde{p}$ such that $\tilde{p}' \in \Delta^n$ and $\|p - \tilde{p}\|_\infty \leq \varepsilon$.*

Proof. Let $\tilde{p}'$ be defined by setting all elements in $\tilde{p}$ that are below $\varepsilon/4$ to zero and all elements above 1 to 1, this introduces at most $\varepsilon/4$ extra error in ℓ_∞-norm so $\|p - \tilde{p}'\|_\infty \leq \varepsilon/2$.

Now, for an element in $\tilde{p}'$ to be non-zero, the corresponding element of p should be at least $\varepsilon/8$, hence $\tilde{p}'$ has at most $8/\varepsilon$ non-zero elements. Let $k \leq \min(n, 8/\varepsilon)$ be the number of non-zero elements in $\tilde{p}'$. Let S be the sum of the entries in $\tilde{p}'$, so

$$\max(0, 1 - n\varepsilon/2) \leq S \leq 1 + k\varepsilon/4.$$

If $S = 1$ then $\tilde{p}' \in \Delta^n$ so we are done.

If $S > 1$, then we decrease each of the non-zero elements by $(S - 1)/k \leq \varepsilon/4$. This introduces at most $\varepsilon/4$ extra error, so the total error is less than $\varepsilon/2 + \varepsilon/4$. Now all elements are non-negative and they sum to 1.

If $S < 1$ and there is an element larger than $1 - \varepsilon/4$, return the distribution that is 1 on the corresponding index and 0 everywhere else. Otherwise we consider two cases, $n \leq 8/\varepsilon$ and $n > 8/\varepsilon$. For the first case, the ℓ_1-norm error in $\tilde{p}'$ is at most $n\varepsilon/2$, so $1 - S$ is at most $n\varepsilon/2$. Hence, by increasing each coordinate by at most $\varepsilon/2$ we can ensure that the resulting vector is in Δ^n. For the second case we pick $2/\varepsilon$ entries in $\tilde{p}'$, giving preference to the non-zero entries, and increase the picked entries by $\frac{\varepsilon(1-S)}{2} \leq \varepsilon/2$.

Finally, we note that this construction can be implemented in time linear in the input or output sparsity, whichever is larger, times $\log(1/\varepsilon)$. ◀

2.2.4 Putting it all together

We can now combine these improvements with our base algorithm to get the following result as a corollary.

▶ **Theorem 9.** *Let $p \in \Delta^n$ and let U_p be a quantum probability oracle acting on q qubits for p. Let $\varepsilon > 0$. Let $S \subseteq [n]$ and let p_{mt} be an upperbound on $\sum_{i \in S} p_i$. An $\widetilde{\mathcal{O}}(1/\varepsilon)$-sparse $\tilde{p} \in \Delta^n$ such that $\|p - \tilde{p}\|_\infty \leq \varepsilon$ can be found with error probability at most $\delta > 0$ using $\mathcal{O}\left(\ln(1/\varepsilon\delta)\sqrt{p_{mt}}/\varepsilon\right)$ applications of U_p (and membership queries for S) and $\widetilde{\mathcal{O}}\left(q \ln(\delta)\sqrt{p_{mt}}/\varepsilon^2\right)$ two-qubit gates. The gatecount can be improved to $\widetilde{\mathcal{O}}\left(q \ln(\delta)\sqrt{p_{mt}}/\varepsilon\right)$ using QRAM.*

We note that the query complexity matches that of normal amplitude estimation (the query complexity of which is known to be optimal as it can solve the parity problem for a $1/\varepsilon$-bit long string [1]) up to logarithmic factors.

Using the equivalence of norms we can also get upper bounds on the query complexity for ℓ_ρ estimates.

▶ **Corollary 10.** *Let $p \in \Delta^n$ and let U_p be a quantum probability oracle acting on q qubits for p. Let $\varepsilon > 0$ and $\rho \geq 1$. Let $S \subseteq [n]$ and let p_{mt} be an upperbound on $\sum_{i \in S} p_i$. An $\widetilde{\mathcal{O}}\left(n^{1/\rho}/\varepsilon\right)$-sparse $\tilde{p} \in \Delta^n$ such that $\|p - \tilde{p}\|_\rho \leq \varepsilon$ can be found with error probability at most $\delta > 0$ using $\mathcal{O}\left(\ln(1/\varepsilon\delta)\sqrt{p_{mt}}n^{1/\rho}/\varepsilon\right)$ applications of U_p (and membership queries* *for S) and $\widetilde{\mathcal{O}}\left(q\ln(\delta)\sqrt{p_{mt}}n^{2/\rho}/\varepsilon^2\right)$ two-qubit gates. The gatecount can be improved to $\widetilde{\mathcal{O}}\left(q\ln(\delta)\sqrt{p_{mt}}n^{1/\rho}/\varepsilon\right)$ using QRAM.*

We note that this might not always be optimal, in particular in the low-precision regime. For example, classical sampling can produce an ε-ℓ_2-estimate using $\widetilde{\mathcal{O}}\left(1/\varepsilon^2\right)$ samples as shown by Kamath et al. [7].

3 Lower bounds

In this section we will prove lower bounds on the number of applications of U_p that are required to approximate p in different norms. Since the ℓ_∞-norm bound follows from known lower bounds on amplitude estimation that can be obtained from the lower bound on parity [1], we focus on the ℓ_1 and ℓ_2 norms. We start by proving a lower bound on ℓ_1-norm estimation.

▶ **Lemma 11.** *Let $\varepsilon \in (0, 1/3)$ and $n \geq 2$. Any algorithm that (with success probability at least $2/3$) for every $p \in \Delta^n$ outputs a $\tilde{p}$ for which $\|p - \tilde{p}\|_1 \leq \varepsilon$ using queries to a quantum probability oracle for p, uses at least $\Omega\left(\frac{n}{\varepsilon}\right)$ such queries.*

Proof. We assume that n is even as we can always add an extra zero entry. Let $k = \Theta\left(1/\varepsilon\right)$, where μ will be defined later. Let $x^{(1)}, \cdots, x^{(n/2)} \in \{0, 1\}^k$ be such that for all i we have $|x^{(i)}| \in \{k/2, k/2 + 1\}$. Finding the Hamming weight of a single $x^{(i)}$ solves the majority problem and hence requires $\Omega\left(k\right)$ quantum queries to a standard (binary) oracle for $x^{(i)}$ [1]. We further note that any algorithm that recovers a $n/2$-bit string requires $\Omega\left(n\right)$ quantum queries. Since quantum query complexity is multiplicative under composition [8] it follows that finding all of the $n/2$ Hamming weights requires $\Omega\left(nk\right) = \Omega\left(n/\varepsilon\right)$ quantum queries. Standard techniques can be used to show that finding a constant fraction of the Hamming weights would still require $\Omega\left(n/\varepsilon\right)$ quantum queries, as Grover search can be used to find the "mistakes".

We now reduce this problem to finding an ℓ_1-approximation of a probability distribution. Let $p \in \Delta^n$ be given by $p_i = 2\frac{|x^{(i)}|}{nk}$ for $i \leq n/2$ and by $p_i = 2\frac{k-|x^{(i)}|}{nk}$ otherwise. Let $\tilde{p}$ be an ε approximation of p. If $|p_i - \tilde{p}_i| < \frac{1}{nk}$ than we can find $|x^i|$ from $\tilde{p}_i$. As $\tilde{p}$ is an ε-ℓ_1-norm estimate, it can only be off more than $1/kn = \Theta\left(\varepsilon/n\right)$ on a small constant fraction of the indices, allowing us to find the Hamming weight for all the others.

Finally we show how to implement a quantum probability oracle for p. We can sample from p using a classical algorithm as follows:

1. Pick a uniformly random $i \in [n/2]$.
2. Pick a uniformly random $j \in [k]$.
3. If $x^{(i)}_j = 1$ return i, if $x^{(i)}_j = 0$ return $i + n/2$.

By replacing the uniformly random picks by the creation of a uniform superposition we get a quantum probability oracle for p.

We conclude that $\Omega\left(n/\varepsilon\right)$ queries to a quantum probability oracle for p are required to obtain an ε-ℓ_1-approximation. ◀

As a corollary we get a lower bound for ℓ_2-estimates in the high precision regime:

▶ **Corollary 12.** *Let $\varepsilon \in (0, 1/3\sqrt{n})$ and $n \geq 2$. Any algorithm that (with success probability at least $2/3$) for every $p \in \Delta^n$ outputs a $\tilde{p} \in \Delta^n$ for which $\|p - \tilde{p}\|_2 \leq \varepsilon$ using queries to a quantum probability oracle for p, uses at least $\Omega\left(\frac{1}{\varepsilon}\right)$ such queries.*

Proof. This follows from the fact that $\|p - \tilde{p}\|_1 \leq \sqrt{n}\,\|p - \tilde{p}\|_2$ combined with Lemma 11. ◀

4 Open questions

Estimating the expectation value of stochastic variables

We can identify a stochastic variable over a finite probability distribution $p \in \Delta^n$ with a vector $a \in \mathbb{R}^n$. Here a_i is the value of the stochastic variable on outcome i. Hence, the expectation value of the stochastic variable is equal to $\langle a, p \rangle$. If we have m stochastic variables $a^{(1)}, \ldots, a^{(m)}$ then we can write these as the rows of a matrix $A \in \mathbb{R}^{m \times n}$. This leads to the following problem:

Let $A \in [-1,1]^{m \times n}$ be a known matrix, let $\varepsilon > 0$ be an error parameter, and let $p \in \Delta^n$ be a unknown probability distribution, accessible via a quantum probability oracle. Output a vector $\tilde{q} \in \mathbb{R}^m$ such that $\|Ap - q\|_\infty \leq \varepsilon$.

Here we take $A \in [-1,1]^{m \times n}$ for normalization purposes.

Classically this problem can be solved using $\mathcal{O}\left(\frac{\ln(m/\delta)}{\varepsilon^2}\right)$ samples. The argument is similar as before: each expectation value can be estimated with error probability δ/m, and union bounding gives the result. However, our quantum algorithm does not generalize as easily. One way to solve the problem is to apply amplitude estimation n times, but this would use $\widetilde{\mathcal{O}}\left(n \ln(1/\delta)/\varepsilon\right)$ applications of U_p. In fact, we can proof the following lower bound:

▶ **Lemma 13.** *Let $\varepsilon \in (0, 1/(3\sqrt{n}))$ and let n be a positive integer power of two. There exists a matrix $A \in \{-1,1\}^{n \times n}$, such that any algorithm that for every $p \in \Delta^n$ (with success probability at least $2/3$) outputs a $\tilde{q} \in \Delta^n$, for which $\|Ap - \tilde{q}\|_\infty \leq \varepsilon$, uses at least $\Omega\left(\frac{\sqrt{n}}{\varepsilon}\right)$ queries to a quantum probability oracle for p.*

Proof. We let $A \in \{-1,1\}^{n \times n}$ be $\sqrt{n} H^{\otimes \log(n)}$, the rescaled n-fold Hadamard, so $\frac{1}{\sqrt{n}} A$ is unitary. Now let $p \in \Delta^n$ be an unknown probability distribution given by a quantum probability oracle. Let $\mathcal{A}$ be an algorithm that uses T queries to a quantum probability oracle for p, and outputs an estimate $\tilde{q}$ such that $\|Ap - \tilde{q}\|_\infty \leq \varepsilon$. This ℓ_∞−norm estimate also gives an ℓ_2-norm estimate $\|Ap - \tilde{q}\|_2 \leq \sqrt{n}\varepsilon$. Applying the unitary $\frac{1}{\sqrt{n}} A^T$ gives

$$\frac{1}{\sqrt{n}} \left\|A^T A p - A^T \tilde{q}\right\|_2 \leq \sqrt{n}\varepsilon,$$

and using that $A^T A = nI$ we get

$$\left\|np - A^T \tilde{q}\right\|_2 \leq n\varepsilon.$$

So $\left\|p - \frac{1}{n} A^T \tilde{q}\right\|_2 \leq \varepsilon$, hence from q we can recover an ε-approximation of p in ℓ_2-norm, which, by Corollary 12 requires at least $\Omega\left(\frac{\sqrt{n}}{\varepsilon}\right)$ queries to a quantum probability oracle for p. ◀

We note that the proof, combined with the $\widetilde{\mathcal{O}}\left(\frac{\ln(m/\delta)}{\varepsilon^2}\right)$ classical algorithm for estimating the expectation value of stochastic variables, gives an alternative proof to that of [7] of the fact that $\widetilde{\mathcal{O}}\left(\ln(n/\delta)/\varepsilon^2\right)$ samples suffice for an ε-ℓ_2-estimate.

Although the lower bound is disappointing, it still leaves open the possibility of an improvement over applying amplitude estimation n times. In particular, when $A = I$ the problem is simply that of ℓ_∞-norm estimation, and hence we know that there is an improved algorithm. Slightly more general, if A can be decomposed as $A = RC$ for matrices R and C such that R has a maximal row sum of r, and C has a maximal column sum of c, then the problem can be solved with $\widetilde{\mathcal{O}}\left(\frac{rc}{\varepsilon}\right)$ queries, by first applying C/c as a leaky Markov chain step, estimating the result in infty norm up to error ε/b, and then applying R. It is however unclear for which matrices a good decomposition exists.

Improvements for partial distributions

While our improved algorithm from Theorem 9 works better when the total probability of seeing a sample we are interested in is low, there is still a discrepancy between the classical dependence on p_{max} and the quantum dependence on $\sqrt{p_{mt}}$.

Lower bound for low precision ℓ_2-norm estimates

Our lower bound for ℓ_2-norm estimates only works for the high precision ($\varepsilon \in \mathcal{O}\left(1/\sqrt{n}\right)$) regime. A $\Omega\left(\frac{1}{\varepsilon}\right)$ lower bound for the $\varepsilon > \frac{1}{\sqrt{n}}$ regime follows from the lower bound on amplitude estimation, but it is an open question whether this may be improved to $\Omega\left(\frac{1}{\varepsilon^2}\right)$.

Circuit depth

Recent work by Giurgica-Tiron et al. [5] addresses a big disadvantage of amplitude estimation on near term hardware: the circuit depth. While classical probabilities can be estimated by a highly parallel system of logarithmic depth using $\widetilde{\mathcal{O}}\left(1/\varepsilon^2\right)$ processors, quantum amplitude estimation is inherently sequential and takes depth $\widetilde{\mathcal{O}}\left(1/\varepsilon\right)$. Giurgica-Tiron et al. give algorithms that interpolate between these two cases, keeping the depth times the number of oracle queries constant at $\widetilde{\mathcal{O}}\left(1/\varepsilon^2\right)$. It would be interesting to achieve a similar trade-off in the multidimensional case.

Applications

A natural question is of course that of applications. Since the algorithm works when samples from p are generated by a quantum algorithm, inherently quantum outputs like that of the HHL algorithm, Hamiltonian simulation, or quantum Gibbs sampling might be a good fit. Our new methods allow a lower dependence on the error ε when performing quantum state tomography on the resulting states than the classical method of simply measuring does.

Another application might lie in distribution learning theory, or more broadly learning theory in general. Here we are given an unknown distribution and are asked to learn certain properties of the distribution. Our estimation algorithm might serve as a new tool to design quantum improvements in this area.

References

1 R. Beals, H. Buhrman, R. Cleve, M. Mosca, and R. de Wolf. Quantum lower bounds by polynomials. *Journal of the ACM*, 48(4):778–797, 2001. Earlier version in FOCS'98. doi:10.1145/502090.502097.

2 Gilles Brassard, Peter Hoyer, Michele Mosca, and Alain Tapp. Quantum amplitude amplification and estimation. *Contemporary Mathematics*, 305, 2002. arXiv:arXiv:quant-ph/0005055.

3 Paul Dagum, Richard Karp, Michael Luby, and Sheldon Ross. An optimal algorithm for monte carlo estimation. *SIAM Journal on Computing*, 29(5):1484–1496, 2000. doi:10.1137/s0097539797315306.

4 András Gilyén, Srinivasan Arunachalam, and Nathan Wiebe. *Optimizing quantum optimization algorithms via faster quantum gradient computation*, pages 1425–1444. SIAM, 2019. doi:10.1137/1.9781611975482.87.

5 Tudor Giurgica-Tiron, Iordanis Kerenidis, Farrokh Labib, Anupam Prakash, and William Zeng. Low depth algorithms for quantum amplitude estimation, 2020. arXiv:2012.03348.

6 Stephen P. Jordan. Fast quantum algorithm for numerical gradient estimation. *Phys. Rev. Lett.*, 95:050501, July 2005. doi:10.1103/PhysRevLett.95.050501.

7 Sudeep Kamath, Alon Orlitsky, Dheeraj Pichapati, and Ananda Theertha Suresh. On learning distributions from their samples. In *Proceedings of The 28th Conference on Learning Theory*, volume 40 of *Proceedings of Machine Learning Research*, pages 1066–1100, Paris, France, 2015. PMLR. URL: `http://proceedings.mlr.press/v40/Kamath15.html`.

8 Shelby Kimmel. Quantum adversary (upper) bound. *Chicago Journal of Theoretical Computer Science*, 19(1):1–14, 2013. `doi:10.4086/cjtcs.2013.004`.

9 Ashley Montanaro. Quantum speedup of monte carlo methods. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 471(2181):20150301, 2015. `doi:10.1098/rspa.2015.0301`.

Quantum Logarithmic Space and Post-Selection

François Le Gall
Graduate School of Mathematics, Nagoya University, Japan

Harumichi Nishimura
Graduate School of Informatics, Nagoya University, Japan

Abuzer Yakaryılmaz[1]
Center for Quantum Computer Science, University of Latvia, Rīga, Latvia
QWorld Association, Tallinn, Estonia

Abstract

Post-selection, the power of discarding all runs of a computation in which an undesirable event occurs, is an influential concept introduced to the field of quantum complexity theory by Aaronson (Proceedings of the Royal Society A, 2005). In the present paper, we initiate the study of post-selection for space-bounded quantum complexity classes. Our main result shows the identity PostBQL = PL, i.e., the class of problems that can be solved by a bounded-error (polynomial-time) logarithmic-space quantum algorithm with post-selection (PostBQL) is equal to the class of problems that can be solved by unbounded-error logarithmic-space classical algorithms (PL). This result gives a space-bounded version of the well-known result PostBQP = PP proved by Aaronson for polynomial-time quantum computation. As a by-product, we also show that PL coincides with the class of problems that can be solved by bounded-error logarithmic-space quantum algorithms that have no time bound.

2012 ACM Subject Classification Theory of computation → Quantum computation theory

Keywords and phrases computational complexity, space-bounded quantum computation, post-selection

Digital Object Identifier 10.4230/LIPIcs.TQC.2021.10

Related Version *Full Version*: https://arxiv.org/abs/2105.02681

Funding *François Le Gall*: Supported by JSPS KAKENHI grants Nos. JP19H04066, JP20H05966, JP20H00579, JP20H04139, JP21H04879 and by the MEXT Quantum Leap Flagship Program (MEXT Q-LEAP) grants No. JPMXS0118067394 and JPMXS0120319794.
Harumichi Nishimura: Supported by JSPS KAKENHI grants Nos. JP19H04066, JP20H05966, JP21H04879 and by the MEXT Q-LEAP grants No. JPMXS0120319794.
Abuzer Yakaryılmaz: Supported by ERC Advanced Grant MQC and ERDF project Nr. 1.1.1.5/19/A/005 "Quantum computers with constant memory".

Acknowledgements We thank the anonymous reviewers of TQC 2021 for helpful comments.

1 Introduction

Post-selection. Post-selection is the power of discarding all runs of a computation in which an undesirable event occurs. This concept was introduced to the field of quantum complexity theory by Aaronson [1]. While unrealistic, post-selection turned out to be an extremely useful tool to obtain new and simpler proofs of major results about classical computation, and also prove new results about quantum complexity classes. The most celebrated result is arguably the identity PostBQP = PP proved by Aaronson [1], which shows that the class of

[1] Part of this research was done while Yakaryılmaz was visiting Kyoto University in November 2016 and March 2017.

problems that can be solved by a bounded-error polynomial-time quantum algorithm with post-selection (PostBQP) is equal to the class of problems that can be solved by unbounded-error polynomial-time classical algorithms (PP), and thus makes possible to bridge quantum complexity classes and classical complexity classes.

Space-bounded quantum complexity classes. The study of space-bounded quantum Turing machines was initiated by Watrous [16]. Watrous showed in particular that any quantum Turing machine running in space s can be simulated by an unbounded-error probabilistic Turing machine running in space $O(s)$. This result implies the identity $\mathsf{PQL} = \mathsf{PL}$, where PQL denotes the class of problems that can be solved by unbounded-error logarithmic-space quantum Turing machines, and PL denotes the class of problems that can be solved by unbounded-error logarithmic-space classical Turing machines. The main open question of the field is whether bounded-error quantum Turing machines can be simulated space-efficiently by *bounded-error* classical Turing machines.

A major step towards establishing the superiority of space-bounded quantum Turing machines over space-bounded classical (bounded-error) Turing machines has been the construction by Ta-Shma [14] of logarithmic-space quantum algorithms for inverting well-conditioned matrices (it is unknown how to perform the same task classically in logarithmic space). While Ta-Shma's quantum algorithm used intermediate measurements, a version of this quantum algorithm without measurement was later constructed by Fefferman and Lin [6] (see also [5] for a related result on space-efficient error reduction for unitary quantum computation). Very recent works [7, 8] have further showed that many other problems from linear algebra involving well-conditioned matrices can be solved as well in logarithmic space by quantum algorithms, and additionally showed that intermediate measurements can be removed from any space-bounded quantum computation.

Our results. In view of the impact of the concept of post-selection to quantum complexity theory and in view of the surge of recent activities on space-bounded quantum complexity classes, a natural question is investigating the power of post-selection for space-bounded quantum complexity classes. To our knowledge, this question has not been investigated so far in the literature (while the notion of post-selection was previously studied in quantum automata theory [21]). In this paper, we tackle this question and obtain the following result (here PostBQL denotes the class of problems that can be solved by a bounded-error polynomial-time logarithmic-space quantum Turing machine that uses post-selection – see Section 2 for a formal definition):

▶ **Theorem 1** (Main Theorem). $\mathsf{PostBQL} = \mathsf{PL}$.

This result thus gives a space-bounded version of the result $\mathsf{PostBQP} = \mathsf{PP}$ mentioned above for polynomial-time complexity classes. This enables us to bridge quantum complexity classes and classical complexity classes for space-bounded computation as well, and thus suggests that post-selection may become a useful tool to analyze space-bounded (quantum and classical) computation as well. Actually, as a by-product of our main result, we also obtain the fact that PL coincides with the class of problems that can be solved by bounded-error logarithmic-space quantum algorithms that has no time bound (namely, the bounded-error logarithmic-space quantum algorithms are as computationally powerful as the unbounded-error ones under no time restriction).

We additionally present several results about logarithmic-space quantum computation with post-selection in Section 4.

Overview of our techniques. As for the result PostBQP = PP proved by Aaronson [1], the nontrivial part of the proof of our main theorem is the simulation of a probabilistic machine by a post-selecting quantum simulation machine. The simulation technique given in [1] requires a polynomial amount of qubits, and thus cannot be used in our setting since we are limited to a logarithmic amount of qubits. Therefore, we propose a different simulation, which is composed of three parts. First, we show how to simulate the computation of a logarithmic-space probabilistic Turing machine by a logarithmic-width probabilistic circuit K (Section 3.1). Note that the computation process of K is represented by a mixture $\sum_j p_j C_j$, which means that the configuration is in C_j with probability p_j. (It can be written as $\sum_j p_j |C_j\rangle\langle C_j|$ when the mixed state formalism [11] is used.) Here, we can assume that there are unique accepting and rejecting configurations C_a and C_r. Thus, the final mixture of K can be represented in the form of $pC_a + (1-p)C_r$, where $p > 1/2$ if the input is a yes-instance, and $p < 1/2$ if it is a no-instance. Second, we give a simulation of the probabilistic circuit K by a logarithmic-space quantum Turing machine M with post-selection (Section 3.2). Note that this simulation is done in a *coherent* manner. Namely, if the mixture of K at some step is $\sum_j p_j C_j$, the quantum state of M at the corresponding simulation step should be the normalized state of $\sum_j p_j |C_j\rangle$. Thus, M produces the normalized state of $|\psi\rangle = p|C_a\rangle + (1-p)|C_r\rangle$ as the final outcome. In fact, we use the power of post-selection for this simulation, and the final outcome can be obtained after post-selection with an exponentially small probability. Then, the third part is fairly similar to the approach used in [1]: using polynomial number of states constructed from the same number of copies of $|\psi\rangle$, we use repetition and post-selection to increase the success probability of the simulation (Section 3.3).

2 Preliminaries

2.1 Space-bounded probabilistic Turing machines

A classical space-bounded Turing machine has an input tape and a work tape. Both tapes are infinite and their cells are indexed by integers, each of which contains the blank symbol (#) unless it is overwritten with a different symbol. The input tape has a read-only head and the work tape has a read/write head. Each head can access a single cell in each time step and, after each transition, it can stay on the same cell, move one cell to the right, or move one cell to the left.

The input alphabet is denoted Σ and the work tape alphabet is denoted Γ, none of which contains the blank symbol. Moreover, $\tilde{\Sigma} = \Sigma \cup \{\#\}$ and $\tilde{\Gamma} = \Gamma \cup \{\#\}$. For a given string x, $|x|$ represents the length of x.

Formally, a (space-bounded) probabilistic Turing machine (PTM) M is a 7-tuple

$$M = (S, \Sigma, \Gamma, \delta, s_i, s_a, s_r),$$

where S is the set of states, $s_i \in S$ is the initial state, $s_a \in S$ and $s_r \in S$ ($s_a \neq s_r$) are the accepting and rejecting states, respectively, and δ is the transition function described below.

At the beginning of the computation, the given input, say $x \in \Sigma^*$, is placed on the input tape between the first cell and the $|x|$-th cell, the input tape head and the work tape head are placed on the cells indexed by 0s, and the state is set to s_i. In each step, M evolves with respect to the transition function and the computation is terminated after entering s_a or s_r. In the former (latter) case, the decision of "acceptance" ("rejection") is made. It must be guaranteed that the input tape head never visits the cells indexed by -1 and $|x|+2$. The formal definition of δ is as follows:

$$\delta : S \times \tilde{\Sigma} \times \tilde{\Gamma} \times S \times \tilde{\Gamma} \times \{-1,0,1\} \times \{-1,0,1\} \rightarrow \left\{0, \frac{1}{2}, 1\right\}.$$

Suppose that M is in $s \in S$ and reads $\sigma \in \tilde{\Sigma}$ and $\gamma \in \tilde{\Gamma}$ on the input and work tapes, respectively. Then, in one step, the new state is set to $s' \in S$, the symbol $\gamma' \in \tilde{\Gamma}$ is written on the cell under the work tape head, and the positions of the input and work tape heads are respectively updated with respect to $d_i \in \{-1,0,1\}$ and $d_w \in \{-1,0,1\}$, with probability

$$\delta(s, \sigma, \gamma, s', \gamma', d_i, d_w),$$

where the input (work) tape head moves one cell to the left if $d_i = -1$ ($d_w = -1$) and one cell to the right if $d_i = 1$ ($d_w = 1$). Remark that any transition with zero probability is never implemented. To be a well-formed PTM, for each triple (s, σ, γ),

$$\sum_{s' \in S, \gamma' \in \tilde{\Gamma}, d_i \in \{-1,0,1\}, d_w \in \{-1,0,1\}} \delta(s, \sigma, \gamma, s', \gamma', d_i, d_w) = 1.$$

For a given input $x \in \Sigma^*$, M can follow more than one computation path. A computation path either halts with a decision or runs forever. A halting path is called accepting (rejecting) if the decision of "acceptance" ("rejection") is made on this path. The accepting (rejecting) probability of M on x is the cumulative sum over all accepting (rejecting) paths.

A language L is said to be recognized by PTM M with unbounded error if and only if any $x \in L$ is accepted by M with probability more than $1/2$ and any $x \notin L$ is accepted with probability less than $1/2$. A language L is said to be recognized by PTM M with error bound $\varepsilon < 1/2$ if and only if any $x \in L$ is accepted by M with probability at least $1 - \varepsilon$ and any $x \notin L$ is rejected with probability at least $1 - \varepsilon$. When $\varepsilon > 0$ is a constant (independent of the input), it is said that L is recognized by M with bounded error. As a special case, if all non-members of L are accepted with probability 0, then it is called one-sided bounded-error. A PTM making only deterministic transitions (i.e., such that the range of the transition function is $\{0,1\}$) is a deterministic Turing machine (DTM).

The range of the transition function can also be defined as $[0,1] \cap \mathbb{Q}$, and thus the PTM, called rational valued PTM, can make more than one transition with rational valued probabilities in each step. Remark that all results presented in this paper are also followed for rational valued PTMs. A nondeterministic Turing machine (NTM) can be defined as a rational valued PTM and a language is said to be recognized by a NTM if and only if for any member there is at least one accepting path and for any non-member there is no accepting path (or equivalently any member is accepted with nonzero probability and any non-member is accepted with zero probability).

A language is recognized by a machine in (expected) time $t(n)$ and space $s(n)$ if the machine, on a given input x, runs no more than (expected) $t(|x|)$ time steps and visits no more than $s(|x|)$ different cells on its work tape with non-zero probability.

The class PL (L and NL) is the set of languages recognized by unbounded-error PTMs (DTMs and NTMs) in logarithmic space (with no time restriction). It is shown that each of these classes coincides with the subclass such that the running time of the corresponding machines is polynomially bounded (note that the proof is nontrivial for PL [10]).

The class BPL (RL) is the set of languages recognized by bounded-error PTMs (one-sided bounded-error PTMs) *in polynomial time* and logarithmic space. On contrary to the above three classes $\mathsf{PL}, \mathsf{L}, \mathsf{NL}$, it is unknown that these two classes are the same as their corresponding classes such that the underlying machines have no time restriction, which we denote by $\mathsf{BPL}(\infty)$ ($\mathsf{RL}(\infty)$).

Any language L is in $\mathsf{C_{=}L}$ [2] if and only if there exists a polynomial-time logarithmic-space PTM M such that any $x \in L$ is accepted by M with probability $\frac{1}{2}$ and any $x \notin L$ is accepted by M with probability other than $\frac{1}{2}$.

2.2 Turing machines with post-selection

A postselecting PTM (PostPTM) has the ability to discard some predetermined outcomes and then makes its decision with the rest of the outcomes, which is guaranteed to happen with non-zero probability (see [1, 21]). Formally, a PostPTM is a modified PTM with three halting states. A PTM has the accepting state s_a and the rejecting state s_r as the halting states. A PostPTM has an additional halting state s_n called the non-postselecting halting state. In this paper, we require that a PostPTM must halt its computation absolutely, i.e., there is no infinite loop.

For a given input x, let $p_{acc,M}(x)$ ($p_{rej,M}(x)$ and $p_{npost,M}(x)$) be the probability of PostPTM M ending in s_a (s_r and s_n). Since M halts absolutely, we know that

$$p_{acc,M}(x) + p_{rej,M}(x) + p_{npost,M}(x) = 1.$$

Due to post-selection, we discard the probability $p_{npost,M}(x)$ and then make a normalization on $p_{acc,M}(x)$ and $p_{rej,M}(x)$ for the final decision. Thus, the input x is accepted (rejected) by M with probability

$$\tilde{p}_{acc,M}(x) := \frac{p_{acc,M}(x)}{p_{acc,M}(x) + p_{rej,M}(x)} \quad \left(\tilde{p}_{rej,M}(x) := \frac{p_{rej,M}(x)}{p_{acc,M}(x) + p_{rej,M}(x)}\right).$$

The postselecting counterparts of BPL and RL are $\mathsf{PostBPL}$ and PostRL, respectively. (For instance, L is in $\mathsf{PostBPL}$ if and only if there are a polynomial-time logarithmic-space PostPTM M and a constant $\varepsilon < 1/2$ such that $\tilde{p}_{acc,M}(x)$ is at least $1-\varepsilon$ when x is in L, and $\tilde{p}_{rej,M}(x)$ is at least $1-\varepsilon$ when x is not in L). Let $\mathsf{PostEPL}$ denote the class of languages recognized with no error (or exactly) by polynomial-time logarithmic-space PostPTMs (i.e., L is in $\mathsf{PostEPL}$ if and only if there is a polynomial-time logarithmic-space PostPTM M such that $p_{acc,M}(x) > 0$ and $p_{rej,M}(x) = 0$ when x is in L, and $p_{acc,M}(x) = 0$ and $p_{rej,M}(x) > 0$ when x is not in L).

2.3 Space-bounded quantum Turing machines and complexity classes

The initial quantum Turing machine (QTM) models (e.g., [4, 3, 16]) were defined fully quantum. While quantum circuits have been used more widely in literature, QTMs are still the main computational models when investigating space bounded complexity classes. However, their definitions have been modified since 90s (e.g., [17, 15, 14, 6]). The main modifications are that the computation is governed classically and the quantum part can be seen like a quantum circuit. This paper follows these modifications. To be more precise, our QTM is a PTM augmented with a quantum tape. Here, the quantum tape is designed like a quantum circuit, i.e., it contains a qubit (or qudit) in each tape cell and it can have more than one tape head so that a quantum gate can be applied to a few qubits at the same time.

We remark that the result given in this paper can also be obtained by the other space-bounded QTMs defined in literature [18, 17, 20, 15, 14], where algebraic numbers are used as transition values. The main advantage of the aforementioned modifications in QTMs is to simplify the proofs and the descriptions of quantum algorithms.

Formally, a (space-bounded) QTM M is a 9-tuple

$$M = (S, \Sigma, \Gamma, \delta_q, \delta_c, s_i, s_a, s_r, \Omega),$$

where, different from the PTMs, the transition function is composed by two functions δ_q and δ_c that are responsible for the transitions on quantum and classical parts, respectively, and Ω is the set of contents of a classical register storing quantum measurement outcomes. (Similarly to the PTMs, S is the set of internal states, Σ is the input alphabet, Γ is the work tape alphabet, and s_i, s_a, and s_r are respectively the initial state, the accepting state, and the rejecting state.) As the physical structure, M additionally has a quantum tape with l heads, and the classical register storing a value in $\Omega = \{1, \ldots, m\}$, where $l, m > 0$ are constants (independent of the input given to M). The quantum tape heads are numbered from 1 to l. For simplicity, we assume that the quantum tape contains only qubits (with states $|0\rangle$ and $|1\rangle$) in its cells. Each cell is set to $|0\rangle$ at the beginning of the computation. For a given input $x \in \Sigma^*$, the classical part is initialized as described for PTMs. The l tape heads on the quantum tape are placed on the qubits numbered from 0 to $l-1$.

The overall computation of M is governed classically. Each transition of M has two phases, quantum and classical, which alternate. We define the transition functions δ_q and δ_c different from the transition functions of PTMs. Suppose that M is in $s \in S$ and reads $\sigma \in \tilde{\Sigma}$ and $\gamma \in \tilde{\Gamma}$, respectively. For each triple (s, σ, γ), $\delta_q(s, \sigma, \gamma)$ can be the identity operator, a projective measurement (in the computational basis), or a unitary operator. If it is the identity operator, the quantum phase is skipped by setting the value of the classical register to 1 (in Ω). If the quantum operator is unitary, then the corresponding unitary operator is applied to the qubits under the heads on the quantum tape, and the value in the classical register is set to 1 (in Ω). If it is a measurement operator, then the corresponding projective measurement is done on the qubits under the heads on the quantum tape, and the measurement outcome, represented by an integer between 1 and $m' \leq m$ (in Ω), is written in the classical register, where m' is the total number of all possible measurement outcomes of the measurement operator.

After the quantum phase, the classical phase is implemented. For each quadruple $(s, \sigma, \gamma, \omega)$, δ_c returns the new state, the symbol written on the work tape, and updates of all heads, where $\omega \in \Omega$.

The termination of the computation of M is the same as the PTMs, i.e., done by entering the accepting state s_a or the rejecting state s_r. One time step corresponds to a single transition. We add the number of qubits visited with non-zero probability during the computation (as well as the number of cells visited on the classical work tape) to the space usage.

Remark that any QTM using superoperators can be simulated by a QTM using unitary operators and measurements with negligible memory and time overheads, i.e., by using extra quantum and classical states, any superoperator can be implemented by unitary operators and measurements in constant steps (e.g. [11, 13]).

Since the computation of the QTM defined above is controlled classically, a postselecting QTM (PostQTM) can be defined similar to PostPTMs: the PostQTM has an additional classical halting state s_n, and any computation that ends in s_n is discarded when calculating the overall accepting and rejecting probability on the given input.

The quantum counterparts of BPL ($\mathsf{BPL}(\infty)$), RL, PL, NL, $\mathsf{PostBPL}$, PostRL, and $\mathsf{PostEPL}$ are BQL ($\mathsf{BQL}(\infty)$), RQL, PQL, NQL[2], $\mathsf{PostBQL}$, $\mathsf{PostRQL}$, and $\mathsf{PostEQL}$, respectively, where QTMs use algebraic numbers as transition amplitudes.

The following relations on logarithmic space quantum and classical complexity classes

[2] Note that NQL is the quantum counterpart of NL based on the criterion by the accepting probabilities of the underlying machine, not the certificate-based counterpart (QMAL).

are already known [9, 16, 17, 7]:

$$\mathsf{L} \subseteq \mathsf{NL} = \mathsf{coNL} \subseteq \mathsf{coC_{=}L} = \mathsf{NQL} \subseteq \mathsf{PL} = \mathsf{PQL}.$$

$$\mathsf{L} \subseteq \mathsf{BPL} \subseteq \mathsf{BQL} \subseteq \mathsf{PL} = \mathsf{PQL}.$$

3 Main Result

In this section, our main theorem (PostBQL = PL) is proved. We start with the easy inclusion.

▶ **Theorem 2.** PostBQL ⊆ PQL = PL

Proof. Any polynomial-time logarithmic-space PostQTM M can be easily converted to a polynomial-time logarithmic-space QTM M' such that M' enters the accepting and rejecting states with equal probability when M enters the non-postselecting halting state. Thus the balance between accepting and rejecting probabilities is preserved, and the language recognized by M with bounded-error is recognized by M' with unbounded error. ◀

In the rest of this section, we give the proof of the following inclusion.

▶ **Theorem 3.** PL ⊆ PostBQL.

As described in Section 1, the proof of Theorem 3 consists of three parts, each of which will be given in the next three subsections. We start by giving an overview of the first part. Let L be a language in PL. Then there exists a PTM M recognizing L with unbounded error such that M on input x halts in $|x|^k$ steps by using at most $d\log(|x|)$ space for some fixed positive integers d and k.

Without loss of generality, we can assume that M always splits into two paths in every step, the work tape alphabet of M has only two symbols 0 and 1, and M halts only when the work tape contains only blanks and both tape heads are placed on the 0-th cells, i.e., there exist a single accepting and a single rejecting configurations. Let m be the number of internal states.

We fix x as the given input with length $|x| = n$. Any configuration of M is represented by a 4-tuple of binary strings

$$(s, h_{\text{in}}, w, h_{\text{wk}}),$$

where s is the internal state, h_{in} is the position of the input head, w is the content of the work tape, and h_{wk} is the position of the work tape. (We also assume that w is always a binary string, which does not contain any blank symbol.) The set of all configurations is denoted by C^x, i.e., $C^x = \{C_1, \ldots, C_N\}$ for some N polynomial in n. The length of any configuration is

$$l = \lceil \log m \rceil + \lceil \log n \rceil + \lceil d \log n \rceil + \lceil \log(d \log n) \rceil \in O(\log n).$$

Based on C^x, we define a stochastic matrix P_x, called *the configuration matrix*, whose columns and rows are indexed by configurations and its (j,i)-th entry represents the probability going from C_i to C_j. Then, the whole computation of M on x can be traced by an N-dimensional column vector, called *configuration vector*:

$$v_{l+1} = P_x v_l,$$

where $1 \leq l \leq n^k$ and v_l represents the probability distribution of the configurations after the l-th step. Here, v_0 is the initial configuration vector having a single nonzero entry, that is

1, corresponding to the initial configuration, and v_{n^k} is the final configuration vector having at most two nonzero entries that keep the overall accepting and rejecting probabilities:

$$v_{n^k} = P_x^{n^k} v_0.$$

Since the computation is split into two paths in each step with equal probability, the overall accepting (A) and rejecting probabilities (R) are respectively of the forms

$$\frac{A'}{2^{n^k}} \text{ and } \frac{R'}{2^{n^k}},$$

where $0 \leq A', R' \leq 2^{n^k}$, $A' + R' = 2^{n^k}$, and $A' \neq 2^{n^k-1}$.

We present a simulation of the above matrix-vector multiplication in logarithmic space. It is clear that keeping all entries of a single configuration vector separately requires polynomial space in n. On the other hand, a single configuration can be kept in logarithmic space. Therefore, we keep a mixture of configurations as a single summation for any time step. In other words, we can keep v_i as

$$v_i[1]C_1 + v_i[2]C_2 + \cdots + v_i[n^k]C_{n^k},$$

where each coefficient $v_i[j]$ represents the probability of being in the corresponding configuration C_j. The transition from v_i to v_{i+1} can be obtained in a single step by applying P_x. However, in our simulation, we can do this in n^k sub-steps. The idea is as follows: In the j-th sub-step, we check whether our mixture has C_j or not. If it exists, then C_j is evolved to C_j' and C_j'' that are the configurations obtained from C_j in a single step when the outcome of the coin is respectively heads or tails. In this way, from the mixture corresponding to v_i, we obtain the next mixture:

$$v_{i+1}[1]C_1 + v_{i+1}[2]C_2 + \cdots + v_{i+1}[n^k]C_{n^k}.$$

Then, the final mixture is

$$AC_a + RC_r,$$

where C_a and C_r are the accepting and rejecting configurations, respectively.

We present the details of this simulation in the following subsection.

3.1 Probabilistic circuit

In this subsection, it is shown that we can construct, in deterministic logarithmic space, a logarithmic-width and polynomial-depth probabilistic circuit $K_{M,x}$ that simulates M on x.

Note that a logarithmic-space DTM can easily output each element of C^x. Moreover, for any $C_j \in C^x$, it can also easily output two possible next configurations C_j' and C_j'' such that M switches from C_j to C_j' if the result of the coin flip is heads and it switches from C_j to C_j'' if the result of the coin flip is tails.

A logarithmic-space DTM D described below can output the desired probabilistic circuit $K_{M,x}$ with width $l + 3$ where (i) the first bit is named as *the random bit* that is used for coin flip, (ii) the second and third bits are named as *the block control bit* and *the configuration control bit* that are used to control the transition between the configurations in each time step, and (iii) the rest of the bits hold a configuration of M on x.

The circuit $K_{M,x}$ consists of n^k *blocks*, and D outputs the n^k blocks. Each block corresponds to a single time step of M on x:

$$block_1, block_2, \ldots, block_{n^k},$$

where each block is identical, i.e., each block implements the transition matrix P_x operating on configurations. Remark that, after $block_i$, we have the mixture representing v_i.

Before each block, the random bit is set to 0 or 1 with equal probability and the block control bit is set to 1. As long as the block control bit is 1, the configurations are checked one by one in the block. Once it is set to 0, the remaining configurations are skipped.

Any block is composed by N *parts* where each part corresponds to a single configuration:

$$part_1, part_2, \ldots, part_N.$$

Here, $part_j$ implements the transitions from C_j in a single step. In $part_j$, we do the following items:

1. If *the block control bit* is 0, then SKIP the remaining items. Otherwise, CONTINUE.
2. SET *the configuration control bit* to 1 (here we assume that M is in C_j).
3. It checks whether M is in C_j.
   - If M is not in C_j, SET *the configuration control bit* to 0 and SKIP the remaining items. (Remark that the block control bit is still 1 in this case, and thus the next configuration C_{j+1} will be checked in $part_{j+1}$.)
   - Otherwise (i.e., if M indeed is in C_j), CONTINUE.
4. SWITCH from C_j to C'_j if *the random bit* is 0 and SWITCH from C_j to C''_j if *the random bit* is 1.
5. SET *the block control bit* to 0.

After all n^k blocks, D outputs the last block called *decision block*. In the last block, it is checked whether the last configuration is C_a or C_r. If it is C_a (resp. C_r), then the first bit of the decision block is set to 1 (resp. 0).

For the above operations, we can use some gates operating on no more than four bits that are the first three bits and one bit from the rest in each time. With l sequential gates, we can determine whether we are in C_j or not. Similarly, with l sequential gates, we can implement the transition from C_j to C'_j and, with another l sequential gates, we can implement the transition from C_j to C''_j. Here, using l sequential gates allows us to keep the size of any gate no more than 4 bits as shown in Fig. 1 (where $H_1, \ldots, H_l$ denote the l sequential gates).

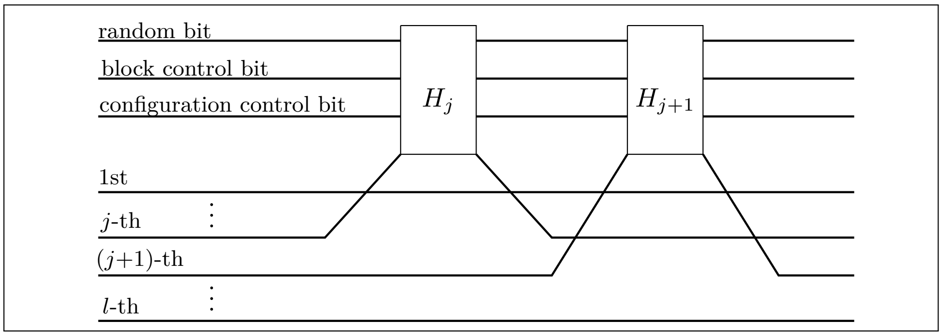


Figure 1 Two sequential gates operating on the first three bits and one bit from the rest.

When physically implementing the above circuit $K_{M,x}$, before each block, the circuit will be in a single configuration, and during executing the block, only the part corresponding to this configuration will be active, and thus the circuit will switch to one of the two possible next configurations. After *the decision block*, we will observe the first bit as 1 and 0 with probabilities A and R, respectively.

Remark that the set of all possible gates which can be used in the above circuit is finite and independent of the input x. The only probabilistic gate is a single bit operator implementing a fair coin toss. The rest of gates are deterministic and basically they are controlled operators with maximum dimension of 16.

Before continuing with the quantum part, we make further simplifications on $K_{M,x}$. As 2-bit AND and OR gates[3] and 1-bit NOT gate form a universal gate set for classical circuits, each deterministic gate (operating at most 4 bits) can be replaced by some finite numbers of NOT, AND, OR, and some 1-bit resetting gates with help of a few extra auxiliary bits used for intermediate calculations, which are appended to the bottom part of the circuit. Let $G = \{G_0, G_1, \ldots, G_t\}$ be the new set of our gates, where G_0 implements the fair coin by outputting the values 0 and 1 with equal probability, and the values are used by the deterministic gates whenever it is needed.

We denote the simplified circuit as $K'_{M,x}$ or shortly as K'. Let l' be the width of K' (note that $l' = l + O(1) = O(\log n)$). Thus, we have K' such that the probability of observing 1 (resp. 0) on the first bit is A (resp. R).

3.2 QTM part

In this subsection, we give a logarithmic-space postselecting QTM that simulates the computation of K' in a *coherent* manner, as described in Section 1.

A logarithmic-space (postselecting) QTM can trace the computation of K' on its quantum tape by help of its classical part. Since the circuit K' is deterministic logarithmic-space constructible, the classical part of the QTM helps to create the parts of K' on the quantum tape whenever it is needed. Moreover, any mixture of the configurations in K' is kept in a pure state of l' qubits (described below).

The QTM uses $l' + 2$ active qubits on the quantum tape for tracing K' on the input. The last two qubits are auxiliary, and the first l' qubits are used to keep the probabilistic state of K'. We consider the quantum tape as a logarithmic-width quantum circuit simulating K'.

For each gate of K', say G_j, we apply a unitary gate (operator) operating on at most 4 qubits, say U_j. Therefore, we use 4 tape heads on the quantum tape.

During the simulation, the first l' qubits are always kept in a superposition and after each unitary operator the last qubit or the last two qubits are always measured. If the outcome is 0 or 00, then the computation continues. Otherwise, the computation is terminated in the non-postselecting state.

In the probabilistic circuit K', G_0 is applied on the first qubit. For each G_0, we apply

$$U_0 = \frac{1}{2}\begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & 1 & -1 \\ 1 & -1 & -1 & 1 \end{pmatrix}$$

on the first and the last qubits, measure the last qubit, and continue if $|0\rangle$ is observed:

$$U_0|00\rangle = \frac{1}{2}|00\rangle + \frac{1}{2}|01\rangle + \frac{1}{2}|10\rangle + \frac{1}{2}|11\rangle \xrightarrow{\text{post-selection}} \frac{1}{\sqrt{2}}|0\rangle + \frac{1}{\sqrt{2}}|1\rangle.$$

$$U_0|10\rangle = \frac{1}{2}|00\rangle - \frac{1}{2}|01\rangle + \frac{1}{2}|10\rangle - \frac{1}{2}|11\rangle \xrightarrow{\text{post-selection}} \frac{1}{\sqrt{2}}|0\rangle + \frac{1}{\sqrt{2}}|1\rangle.$$

[3] We assume that these gates are represented by 4 × 4 matrices.

Thus, coin-flipping operator can be easily implemented.

For the other operators (including the ones given below), we use the techniques given in [12]. For any G_j $(1 \leq j \leq t)$, we apply unitary operator U_j acting on four qubits. Before applying U_j, the quantum part is in

$$\sum_{a,b \in \{0,1\}} \alpha_{a,b} |ab00\rangle,$$

since the last two qubits are measured before and any outcome other than $|00\rangle$ is discarded by entering the non-postselecting state. Thus, only $4 \times 4 = 16$ entries of U_j affects the above quantum state. We construct U_j step by step as follows. These 16 entries are set to the corresponding values from G_j. Thus, the probabilistic state, which is kept in the pure state, can be traced exactly up to some normalization factor.

Without loss of generality, we assume that (by reordering the quantum states) these 16 values are placed in the top left corner. Then, U_j is of the form

$$\frac{1}{e} \left(\begin{array}{c|c|c|c} G_j & G'_j & G''_j & 0 \\ \hline * & * & * & * \end{array} \right),$$

where e is the normalization factor and all G_j, G'_j, and G''_j are 4×4 matrices.

The entries of G'_j are set in order to make the first four rows pairwise orthogonal:

$$G'_j = \begin{pmatrix} 1 & 0 & 0 & 0 \\ \gamma_{1,2} & 1 & 0 & 0 \\ \gamma_{1,3} & \gamma_{2,3} & 1 & 0 \\ \gamma_{1,4} & \gamma_{2,4} & \gamma_{3,4} & 0 \end{pmatrix},$$

where the values are set column by column. The values of $\gamma_{1,2}$, $\gamma_{1,3}$, and $\gamma_{1,4}$ are set to the appropriate values such that the first row becomes orthogonal to the second, the third, and the fourth ones, respectively. Similarly, we set the values of the second and third columns. Since G_j is composed by integers, G'_j is also composed by integers.

The entries of G''_j are set in order to make the first four rows with equal length, say e, which is a square of an integer:

$$G''_j = \begin{pmatrix} \gamma_1 & 0 & 0 & 0 \\ 0 & \gamma_2 & 0 & 0 \\ 0 & 0 & \gamma_3 & 0 \\ 0 & 0 & 0 & \gamma_4 \end{pmatrix},$$

where diagonal entries are picked as the square roots of some integers. Remark that the entries of G''_j does not change the pair-wise orthogonality of the first four rows. Moreover, at this point, the first four rows become pair-wise orthonormal (due to normalization factor e). One can easily fill up the rest of the matrix with some arbitrary algebraic numbers in order to have a complete unitary matrix.

Since the set of G depends on the transitions of the PTM M, each U_j can be kept in the description of the QTM.

By using the above quantum operators, we can simulate K' with exponentially small probability. Only note that, due to normalization factors, the computation is terminated in the non-postselecting state with some probabilities after applying each unitary gate.

At the end of the simulation of K', we separate the first qubit from the rest of qubits, each of which is set to $|0\rangle$. Then, we have this unnormalized quantum state in the first qubit:

$$(1 - A)|0\rangle + A|1\rangle = \begin{pmatrix} 1 - A \\ A \end{pmatrix}.$$

The operator $\begin{pmatrix} 1/2 & 3/2 \\ 1/2 & -1/2 \end{pmatrix}$ maps the above quantum state to

$$|\tilde{u}\rangle = \begin{pmatrix} \frac{1}{2}+A \\ \\ \frac{1}{2}-A \end{pmatrix}.$$

Since this operator can be also implemented with post-selection by using an extra qubit, the new unnormalized quantum state is set to $|\tilde{u}\rangle$.

If $A=0$, then the quantum state $|u\rangle$, that is the normalized version of $|\tilde{u}\rangle$, is identical to $|+\rangle = \frac{|0\rangle+|1\rangle}{\sqrt{2}}$. If $A < \frac{1}{2}$, then the quantum state $|u\rangle$ lies between $|+\rangle$ and $|0\rangle$, and thus it is closer to $|+\rangle$ compared to $|-\rangle = \frac{|0\rangle-|1\rangle}{\sqrt{2}}$. If $A > \frac{1}{2}$, then $|u\rangle$ lies between $|0\rangle$ and $|-\rangle$, and thus it is closer to $|-\rangle$ compared to $|+\rangle$.

After making a measurement in $\{|+\rangle, |-\rangle\}$ basis, we can easily distinguish the cases whether A is close to 0 or A is close to 1 with bounded error. In the case of when A is close to $\frac{1}{2}$, the probability of observing these basis states can be very close to each other. In Section 3.3, we use a modified version of the trick used by Aaronson [1] to increase the success probability. Actually, we will need to use the above QTM $O(n^k)$ times sequentially in logarithmic space.

3.3 Executing a series of QTMs

Let p be our integer parameter from the set $\{0,1,\ldots,n^k\}$. For each p, we consider a QTM $M[p]$ as follows. First, we execute the above QTM in Section 3.2, and then transform $|\tilde{u}\rangle$ to

$$|\tilde{u}_p\rangle = \begin{pmatrix} \frac{1}{2}+A \\ \\ 2^{n^k-p}\left(\frac{1}{2}-A\right) \end{pmatrix}$$

in (n^k-p) iterations. In each iteration, we combine the first qubit with another qubit in state $|0\rangle$, apply the quantum operator

$$\frac{1}{2}\begin{pmatrix} 1 & \sqrt{3} & 0 & 0 \\ \sqrt{3} & -1 & 0 & 0 \\ 0 & 0 & 2 & 0 \\ 0 & 0 & 0 & 2 \end{pmatrix},$$

and then the second qubit is measured. If the measurement outcome is $|0\rangle$, then the computation continues. Otherwise, the computation is terminated by entering the non-postselecting state. (By induction, we can easily see that $|\tilde{u}\rangle \xrightarrow{n^k-p \text{ steps}} |\tilde{u}_p\rangle$.) Note that for each p, the QTM $M[p]$ can be done in logarithmic space as the QTM described in Section 3.2 is done in $O(\log n)$ space, and the counter for the iteration for creating $|\tilde{u}\rangle$ needs $O(\log n)$ space as well.

By substituting $A = \frac{A'}{2^{n^k}}$, the quantum state $|\tilde{u}_p\rangle$ can be rewritten as

$$|\tilde{u}_p\rangle = \begin{pmatrix} \frac{1}{2}+\frac{A'}{2^{n^k}} \\ \\ 2^{n^k-p}\left(\frac{1}{2}-\frac{A'}{2^{n^k}}\right) \end{pmatrix} = \begin{pmatrix} \frac{1}{2}+\frac{A'}{2^{n^k}} \\ \\ 2^{n^k-p}\left(\frac{2^{n^k}-2A'}{2^{n^k+1}}\right) \end{pmatrix} = \begin{pmatrix} \frac{1}{2}+\frac{A'}{2^{n^k}} \\ \\ \frac{2^{n^k}-2A'}{2^{p+1}} \end{pmatrix}.$$

It is easy to see that

- when $A < \frac{1}{2}$ $\left(A' < \frac{2^{n^k}}{2} \Rightarrow 2A' < 2^{n^k}\right)$, the normalized state $|u_p\rangle$ of $|\tilde{u}_p\rangle$ lies in the first quadrant, and thus it is closer to $|+\rangle$, and
- when $A > \frac{1}{2}$ $\left(A' > \frac{2^{n^k}}{2} \Rightarrow 2A' > 2^{n^k}\right)$, $|u_p\rangle$ lies in the fourth quadrant, and thus it is closer to $|-\rangle$.

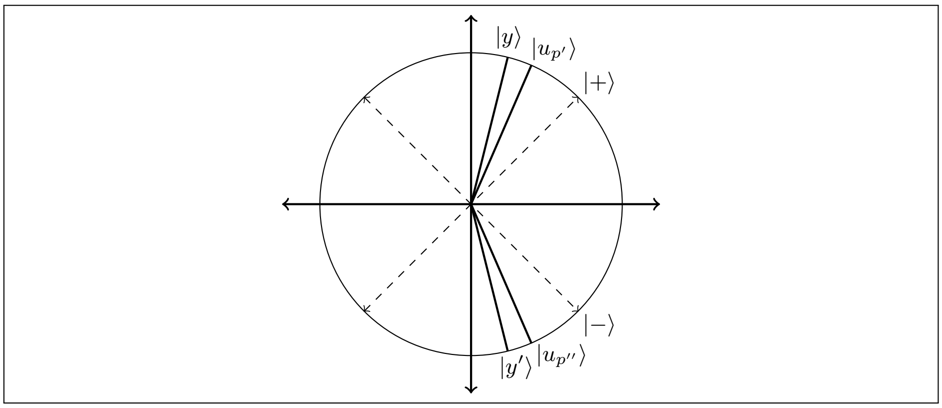


Figure 2 The visualization of $|y\rangle$ and $|u_{p'}\rangle$ when $A < \frac{1}{2}$, and $|y'\rangle$ and $|u_{p''}\rangle$ when $A > \frac{1}{2}$.

Case $A < \frac{1}{2}$: As $A \leq \frac{1}{2} - \frac{1}{2^{n^k}}$ (recall that A is the accepting probability of the PTM M on input x that halts in n^k steps),

$$\frac{2^{n^k} - 2A'}{2^{p+1}} \geq \frac{2}{2^{p+1}}.$$

Thus, there exists a value of p, say p', such that

$$\frac{2^{n^k} - 2A'}{2^{p'+1}} \in [1, 2].$$

Then, since $\frac{1}{2} \leq \frac{1}{2} + \frac{A'}{2^{n^k}}$ and $2 \geq \frac{2^{n^k}-2A'}{2^{p'+1}}$, the quantum state $|y\rangle = \frac{2}{\sqrt{17}}\begin{pmatrix} 1/2 \\ 2 \end{pmatrix}$ lies between $|1\rangle$ and $|u_{p'}\rangle$, and since $\frac{1}{2} \leq \frac{1}{2} + \frac{A'}{2^{n^k}} < 1$ and $2 \geq \frac{2^{n^k}-2A'}{2^{p'+1}} \geq 1$, $|u_{p'}\rangle$ lies between $|y\rangle$ and $|+\rangle$ (see Fig. 2). Thus, the probability of observing $|+\rangle$ after measuring $|u_{p'}\rangle$ in $\{|+\rangle, |-\rangle\}$ basis is always greater than

$$\frac{25}{34} > \frac{7}{10}$$

since $|\langle y|+\rangle|^2 = \frac{25}{34}$.

Case $A > \frac{1}{2}$: The case is similar to the previous case. There exists a value of p, say p'', such that

$$\frac{2^{n^k} - 2A'}{2^{p''+1}} \in [-2, -1].$$

Then, the quantum state $|y'\rangle = \frac{2}{\sqrt{17}}\begin{pmatrix} 1/2 \\ -2 \end{pmatrix}$ lies between $-|1\rangle$ and $|u_{p''}\rangle$ and $|u_{p''}\rangle$ lies between $|y'\rangle$ and $|-\rangle$ (see Fig. 2). Thus the probability of observing $|u_{p''}\rangle$ when measuring in $\{|+\rangle, |-\rangle\}$ basis is always greater than

$$\frac{25}{34} > \frac{7}{10}.$$

Now the overall quantum algorithm is as follows:

1. Prepare counter C to 0. For each $p \in \{0, 1, \dots, n^k - 1\}$, the following steps are implemented.
   a. We execute the above QTM $M[p]$, and make the measurement at the end in $\{|+\rangle, |-\rangle\}$ basis. (Note that the execution can be discarded by entering the non-postselecting state in the procedure of Section 3.2.)
   b. If the measurement result corresponds to $|+\rangle$, then we reset the quantum register to all $|0\rangle$ (note that this is possible using the classical control since all the non-$|0\rangle$ qubits are induced only by post-selection, and thus we know what states they are in), and add $+1$ to C.
   c. If the measurement result corresponds to $|-\rangle$, then we reset the quantum register to all $|0\rangle$, and add -1 to C.
2. If $C = n^k$ (namely, we observe $|+\rangle$ in all executions), then the input is rejected.
3. If $C = -n^k$ (namely, we observe $|-\rangle$ in all executions), then the input is accepted.
4. Otherwise (namely, if we observe the outcomes $|+\rangle$ and $|-\rangle$ at least once in some executions), the computation is terminated in the non-postselecting state.

Note that the overall quantum algorithm is implemented in logarithmic space since the counter is clearly implemented in $O(\log n)$ space, and $M[p]$ is also implemented in $O(\log n)$ space, and each iteration of step 1 is done by the reuse of the classical and quantum registers.

The analysis of the algorithm is as follows:

- When $A < \frac{1}{2}$, the probability of observing $|+\rangle$ is always greater than $|-\rangle$ in each execution and at least once it is $\frac{7}{3}$ times more. Thus, if $x \notin L$, the probability of observing all $|+\rangle$'s is at least $\frac{7}{3}$ times more than the probability of observing all $|-\rangle$'s after all executions.
- When $A > \frac{1}{2}$, the probability of observing $|-\rangle$ is always greater than $|+\rangle$ in each execution and at least once it is $\frac{7}{3}$ times more. Thus, if $x \in L$, the probability of observing all $|-\rangle$'s is at least $\frac{7}{3}$ times more than the probability of observing all $|+\rangle$'s after all executions.

Therefore, after normalizing the final accepting and rejecting postselecting probabilities, it follows that L is recognized by a polynomial-time logarithmic-space postselecting QTM with error bound $\frac{3}{10}$. This completes the proof of Theorem 3. (The error bound can easily be decreased by using the standard probability amplification techniques.)

3.4 Additional result

Additionally, we can show that PostBQL is contained in the class of languages recognized by logarithmic space bounded-error QTMs that halt in *expected exponential time.*

▶ **Theorem 4.** PostBPL $\subseteq$ BPL(exp) *and* PostBQL $\subseteq$ BQL(exp)*, where* BPL(exp) *(*BQL(exp)*) is the class of languages recognized by logarithmic space bounded-error PTMs (QTMs) that halt in expected exponential time.*

Proof. Let M be a polynomial-time logarithmic-space PostPTM. By restarting the whole computation from the beginning instead of entering the non-postselecting state, we can obtain a logarithmic-space exponential-time PTM M' from M, i.e., (i) the restarting mechanism does not require any extra space, and, (ii) since M produces no less than exponentially small halting probability in polynomial time, M' halts with probability 1 in exponential expected time. Both machines recognize the same language with the same error bound since the restarting and postselecting mechanism can be used interchangeably [19, 21], i.e., the accepting and rejecting probabilities by M and M' are the same on every input. Thus, we can conclude that PostBPL $\subseteq$ BPL(exp). In the same way, we can obtain that PostBQL $\subseteq$ BQL(exp). ◀

As $\mathsf{BQL}(\mathsf{exp}) \subseteq \mathsf{BQL}(\infty) \subseteq \mathsf{PQL}$ by definition and Watrous showed $\mathsf{PQL} = \mathsf{PL}$ [17], our main result ($\mathsf{PL} = \mathsf{PostBQL}$) leads to the following equivalence among $\mathsf{BQL}(\mathsf{exp})$, $\mathsf{BQL}(\infty)$ and PL.

▶ **Corollary 5.** $\mathsf{PL} = \mathsf{PQL} = \mathsf{PostBQL} = \mathsf{BQL}(\mathsf{exp}) = \mathsf{BQL}(\infty)$.

We leave open whether $\mathsf{BPL}(\mathsf{exp})$ is contained in $\mathsf{PostBPL}$.

4 Related Results

In this section, we provide several results on logarithmic-space complexity classes with post-selection. The first result is a characterization of NL by logarithmic-space complexity classes.

▶ **Theorem 6.** $\mathsf{NL} = \mathsf{PostEPL} = \mathsf{PostRL}$.

Proof. We start with the first equality $\mathsf{NL} = \mathsf{PostEPL}$. Let $L \in \mathsf{NL}$. Since $\mathsf{NL} = \mathsf{coNL}$ [9], $\overline{L}$ is also in NL. Then, there exist polynomial-time logarithmic-space NTMs N_1 and N_2 recognizing L and $\overline{L}$. Based on N_1 and N_2, we can construct a polynomial-time logarithmic-space PostPTM M such that M executes N_1 and N_2 with equal probability on the given input. Then, M accepts the input if N_1 accepts and rejects the input if N_2 accepts. Any other outcome is discarded by M. Therefore, (i) any $x \in L$ is accepted with nonzero probability and rejected with zero probability by M, and, (ii) any $x \in \overline{L}$ is accepted with zero probability and rejected with nonzero probability by M. Thus, L is recognized by M with no error, and thus $L \in \mathsf{PostEPL}$.

Let $L \in \mathsf{PostEPL}$. Then, there exists a polynomial-time logarithmic-space PostPTM M recognizing L with no error. Based on M, we can construct a polynomial-time logarithmic-space NTM N such that N executes M on the given input and switches to the rejecting state if M ends in the non-postselecting halting state. Thus, N accepts all and only strings in L. Therefore, $L \in \mathsf{NL}$.

Now we are done with equality $\mathsf{NL} = \mathsf{PostEPL}$. It is trivial that $\mathsf{PostEPL} \subseteq \mathsf{PostRL}$. To complete the proof, it is enough to show that $\mathsf{PostRL} \subseteq \mathsf{NL}$. If a language is recognized by a polynomial-time logarithmic-space PostPTM M with one-sided bounded-error, then it is also recognized by a polynomial-time logarithmic-space NTM M' where M' is modified from M such that if M enters the non-postselelecting state, then M' enters the rejecting state. ◀

By using the same argument, we can also obtain the following result on quantum class $\mathsf{PostEQL}$ (note that the first equality comes from $\mathsf{NQL} = \mathsf{coC_{=}L}$ [16, 7]).

▶ **Theorem 7.** $\mathsf{C_{=}L} \cap \mathsf{coC_{=}L} = \mathsf{NQL} \cap \mathsf{coNQL} = \mathsf{PostEQL}$.

As will be seen below, the relation between $\mathsf{PostEQL}$ and $\mathsf{PostRQL}$ seems different from the relation between their classical counterparts since $\mathsf{C_{=}L}$ and $\mathsf{coC_{=}L}$ may be different classes. Remark that it is also open whether NL is a proper subset of $\mathsf{C_{=}L} \cap \mathsf{coC_{=}L}$ or not.

By using the quantum simulation given in Section 3, we can obtain the following result.

▶ **Theorem 8.** $\mathsf{coC_{=}L} = \mathsf{PostRQL}$.

Proof. It is easy to see that $\mathsf{PostRQL} \subseteq \mathsf{NQL}$. Let L be a language in $\mathsf{PostRQL}$ and M be a polynomial-time logarithmic-space PostQTM recognizing L with one-sided bounded-error. By changing the transitions to the non-postselecting state of M to the rejecting state, we can obtain a polynomial-time logarithmic-space NQTM recognizing L, and thus $\mathsf{PostRQL} \subseteq \mathsf{NQL}$. Since $\mathsf{NQL} = \mathsf{coC_{=}L}$ [16, 7], we obtain $\mathsf{PostRQL} \subseteq \mathsf{coC_{=}L}$.

Now we prove the other direction. Let L be in $\mathsf{coC_{=}L}$. Then there exists a polynomial-time logarithmic-space PTM M' that accepts any non-member of L with probability $\frac{1}{2}$ and any member with probability different from $\frac{1}{2}$. Let x be a given input with length n.

We use the simulation given in Section 3. We make the same assumptions on the PTM M' except that M' accepts some string with probability $\frac{1}{2}$ and M' never accepts any string with probability in the following interval

$$\left(\frac{1}{2} - \frac{1}{2^{n^k}}, \frac{1}{2} + \frac{1}{2^{n^k}}\right)$$

for some fixed integer k. This condition is trivial if the running time never exceeds n^k, i.e., the total number of probabilistic branches never exceeds 2^{n^k}.

Then, we construct a polynomial-time logarithmic-space PostQTM as described in Section 3 with the following unnormalized final quantum state:

$$\begin{pmatrix} 2A - 1 \\ 2^{-n^k} \end{pmatrix},$$

where A is the accepting probability of M'. We measure this qubit and accept (reject) the input, if we observe $|0\rangle$ ($|1\rangle$). All the other outcomes are discarded by entering the non-postselecting state.

It is clear that for any non-member of L, A is always equal to $\frac{1}{2}$, and thus the QTM accepts the input with zero probability and rejects the input with some non-zero probability. Therefore, any non-member of L is rejected with probability 1.

On the other hand, for any member, the amplitude of $|0\rangle$ is at least twice of the amplitude of $|1\rangle$, and thus the accepting probability is at least four times more than the rejecting probability. Thus, any member is accepted with probability at least $\frac{4}{5}$. The success probability can be increased by using the standard probability amplification techniques. ◀

References

1 Scott Aaronson. Quantum computing, postselection, and probabilistic polynomial-time. *Proceedings of the Royal Society A*, 461(2063):3473–3482, 2005.

2 Eric Allender and Mitsunori Ogihara. Relationships among PL, #L, and the determinant. *RAIRO Theoretical Informatics and Applications*, 30(1):1–21, 1996.

3 Ethan Bernstein and Umesh Vazirani. Quantum complexity theory. *SIAM Journal on Computing*, 26(5):1411–1473, 1997.

4 David Deutsch. Quantum theory, the Church-Turing principle and the universal quantum computer. *Proceedings of the Royal Society of London A*, 400:97–117, 1985.

5 Bill Fefferman, Hirotada Kobayashi, Cedric Yen-Yu Lin, Tomoyuki Morimae, and Harumichi Nishimura. Space-efficient error reduction for unitary quantum computations. In *Proceedings of the 43rd International Colloquium on Automata, Languages, and Programming*, volume 55 of *LIPIcs*, pages 14:1–14:14, 2016.

6 Bill Fefferman and Cedric Yen-Yu Lin. A complete characterization of unitary quantum space. In *Proceedings of the 9th Innovations in Theoretical Computer Science Conference*, volume 94 of *LIPIcs*, pages 4:1–4:21, 2018.

7 Bill Fefferman and Zachary Remscrim. Eliminating intermediate measurements in space-bounded quantum computation. In *Proceedings of the 53rd Annual ACM Symposium on Theory of Computing, to appear*, 2021. Also available at arXiv:2006.03530.

8 Uma Girish, Ran Raz, and Wei Zhan. Quantum logspace algorithm for powering matrices with bounded nom. In *Proceedings of the 48th International Colloquium on Automata, Languages, and Programming, to appear*, 2021. Also available at arXiv:2006.04880.

9 Neil Immerman. Nondeterministic space is closed under complementation. *SIAM Journal on Computing*, 17(5):935–938, 1988.

10 Hermann Jung. On probabilistic time and space. In *Automata, Languages and Programming*, volume 194 of *LNCS*, pages 310–317. Springer, 1985.

11 Michael A. Nielsen and Isaac L. Chuang. *Quantum Computation and Quantum Information*. Cambridge University Press, 2000.

12 A. C. Cem Say and Abuzer Yakaryılmaz. Computation with narrow CTCs. In *Unconventional Computation*, volume 6714 of *LNCS*, pages 201–211. Springer, 2011.

13 A. C. Cem Say and Abuzer Yakaryılmaz. Quantum finite automata: A modern introduction. In *Computing with New Resources*, volume 8808 of *LNCS*, pages 208–222. Springer, 2014.

14 Amnon Ta-Shma. Inverting well conditioned matrices in quantum logspace. In *44th ACM Annual Symposium on Theory of Computing*, pages 881–890. ACM, 2013.

15 Dieter van Melkebeek and Thomas Watson. Time-space efficient simulations of quantum computations. *Theory of Computing*, 8(1):1–51, 2012.

16 John Watrous. Space-bounded quantum complexity. *Journal of Computer and System Sciences*, 59(2):281–326, 1999.

17 John Watrous. On the complexity of simulating space-bounded quantum computations. *Computational Complexity*, 12(1-2):48–84, 2003.

18 John Watrous. Quantum computational complexity. In *Encyclopedia of Complexity and System Science*. Springer, 2009. Also available at arXiv:0804.3401.

19 Abuzer Yakaryılmaz and A. C. C. Say. Succinctness of two-way probabilistic and quantum finite automata. *Discrete Mathematics and Theoretical Computer Science*, 12(2):19–40, 2010.

20 Abuzer Yakaryılmaz and A. C. Cem Say. Unbounded-error quantum computation with small space bounds. *Information and Computation*, 279(6):873–892, 2011.

21 Abuzer Yakaryılmaz and A. C. Cem Say. Proving the power of postselection. *Fundamenta Informaticae*, 123(1):107–134, 2013.