REPORT DOCUMENTATION PAGE

Form Approved OMB NO. 0704-0188

1. REPORT DATE (DD-MM-YYYY)	2. REPORT TYPE	3. DATES COVERED (From - To)
10-01-2011	Final Report	1-Jun-2005 - 31-May-2008

4. TITLE AND SUBTITLE

Applications of the Schur Basis to Quantum Algorithms

5a. CONTRACT NUMBER

W911NF-05-1-0312

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER

OBXXX1

6. AUTHORS

Isaac Chuang, Aram Harrow

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAMES AND ADDRESSES

Massachusetts Institute of Technology
Office of Sponsored Programs
Bldg. E19-750
Cambridge, MA 02139 -4307

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

U.S. Army Research Office
P.O. Box 12211
Research Triangle Park, NC 27709-2211

10. SPONSOR/MONITOR'S ACRONYM(S)

ARO

11. SPONSOR/MONITOR'S REPORT NUMBER(S)

47954-PH-QC.1

12. DISTRIBUTION AVAILIBILITY STATEMENT

Approved for Public Release; Distribution Unlimited

13. SUPPLEMENTARY NOTES

The views, opinions and/or findings contained in this report are those of the author(s) and should not contrued as an official Department of the Army position, policy or decision, unless so designated by other documentation.

14. ABSTRACT

Quantum computation offers a promising avenue to high performance computing, for certain applications, but depends on development of new quantum algorithms. Thus far, all major quantum algorithms which are exponentially fast compared with classical counterparts are based on the quantum Fourier transform. This project seeks to develop new quantum algorithms, based on a different mathematical transform known as the Schur transform. The Schur transform (or Schur-Weyl duality) arises naturally in many areas of mathematics, chemistry,

15. SUBJECT TERMS

Quantum Computation, Quantum Information, High performance computation, Quantum Algorithms

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	15. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			Isaac Chuang
UU	UU	UU	UU		19b. TELEPHONE NUMBER: 617-253-1692

Report Title

Applications of the Schur Basis to Quantum Algorithms

ABSTRACT

Quantum computation offers a promising avenue to high performance computing, for certain applications, but depends on development of new quantum algorithms. Thus far, all major quantum algorithms which are exponentially fast compared with classical counterparts are based on the quantum Fourier transform. This project seeks to develop new quantum algorithms, based on a different mathematical transform known as the Schur transform. The Schur transform (or Schur-Weyl duality) arises naturally in many areas of mathematics, chemistry, and physics, and many applications of it have now been found in quantum coding and information theory. This project has also led to the efficient quantum circuits for the Schur transform, and a new quantum algorithm for superpolynomial speedups based on quantum circuits.

List of papers submitted or published that acknowledge ARO support during this reporting period. List the papers, including journal references, in the following categories:

(a) Papers published in peer-reviewed journals (N/A for none)

1. D.A.Bacon, I.L. Chuang, A.W. Harrow. “Efficient Quantum Circuits for Schur and Clebsch-Gordan Transforms.” Phys. Rev. Lett., vol. 97, pp. 170502, 2006.
2. D.A.Bacon, I.L. Chuang and A.W. Harrow. “The Quantum Schur Transform: I. Efficient Qudit Circuits.” Proc. 18th ACM-SIAM Symposium on Discrete Algorithms (SODA), pp. 1235–1244, 2007.
3. A.M. Childs, A.W. Harrow and P. Wocjan. “Weak Fourier-Schur sampling, the hidden subgroup problem, and the quantum collision problem,” Proc. 24th Symposium on Theoretical Aspects of Computer Science (STACS 2007), Lecture Notes in Computer Science 4393, pp. 598–609, 2007.
4. A.W. Harrow. “Quantum expanders from any classical Cayley graph expander.” Q. Inf. Comp., vol. 8, no. 8/9, pp. 715-721, 2008. [arXiv:0709.1142]
5. A.W. Harrow and R.A. Low. “Random Quantum Circuits are Approximate 2-designs” Comm. Math. Phys. vol. 291, no. 1, pp. 257–302, 2009. [arXiv:0802.1919]
6. M. B. Hastings and A.W. Harrow. “Classical and Quantum Tensor Product Expanders.” Q. Inf. Comp. vol. 9, pp. 336–360, 2009. [arXiv:0804.0011]
7. S. Hallgren and A.W. Harrow. “Superpolynomial speedups based on almost any quantum circuit.” Proc. of the 35th International Colloquium on Automata, Languages and Programming (ICALP 2008), LNCS 5125, pp. 782–795, 2008. [arXiv:0805.0007]

Number of Papers published in peer-reviewed journals: 7.00

(b) Papers published in non-peer-reviewed journals or in conference proceedings (N/A for none)

Richard Low, Ph.D. Thesis, University of Bristol, Department of Computer Science, 2009

Hyeyoun Chung, S.M. Thesis, MIT, Department of EECS, 2008

Number of Papers published in non peer-reviewed journals: 2.00

(c) Presentations

 Aram Harrow. IBM Research, Yorktown Heights, NY. January 5, 2007. invited seminar.
 Aram Harrow. ACM-SIAM Symposium on Discrete Algorithms (SODA), New Orleans, LA. 9 January 2007. contributed conference talk.
 Aram Harrow. Caltech Institute for Quantum Information. January 23, 2007. invited seminar.
 Aram Harrow. Quantum Information Processing (QIP), Brisbane, Australia. Jan 30, 2007. contributed conference talk.
 Aram Harrow. Quantum Information Processing (QIP), New Delhi, India. December 20, 2007. contributed conference talk.
 Aram Harrow. CWI, Amsterdam, Netherlands. February 22, 2008. invited seminar.
 Richard Low. February 29, 2008. University of Leeds. invited seminar.
 Aram Harrow. LANL Classical and Quantum Information Theory conference, Santa Fe, NM. March 26, 2008. invited conference talk.
 Aram Harrow. Caltech Institute for Quantum Information. April 3, 2008. invited seminar.
 Aram Harrow. Quantum Information and Graph Theory. Perimeter Institute,Waterloo, ON. April 28, 2008. invited conference talk.
 Aram Harrow. Canadian Institute for Advanced Research. June 4, 2008. invited conference talk.

Number of Presentations: 10.00

Non Peer-Reviewed Conference Proceeding publications (other than abstracts):

Number of Non Peer-Reviewed Conference Proceeding publications (other than abstracts): 0

Peer-Reviewed Conference Proceeding publications (other than abstracts):

Number of Peer-Reviewed Conference Proceeding publications (other than abstracts): 0

(d) Manuscripts

Number of Manuscripts: 0.00

Patents Submitted

Patents Awarded

Awards

Graduate Students

NAME	PERCENT_SUPPORTED
Richard Low	1.00
Hyeyoun Chung	0.50
FTE Equivalent:	**1.50**
Total Number:	**2**

Names of Post Doctorates

NAME	PERCENT_SUPPORTED
FTE Equivalent:	
Total Number:	

Names of Faculty Supported

NAME	PERCENT_SUPPORTED	National Academy Member
Isaac Chuang	0.10	No
Aram Harrow	0.10	No
FTE Equivalent:	**0.20**	
Total Number:	**2**	

Names of Under Graduate students supported

NAME	PERCENT_SUPPORTED
FTE Equivalent:	
Total Number:	

Student Metrics

This section only applies to graduating undergraduates supported by this agreement in this reporting period

The number of undergraduates funded by this agreement who graduated during this period:	0.00
The number of undergraduates funded by this agreement who graduated during this period with a degree in science, mathematics, engineering, or technology fields:.....	0.00
The number of undergraduates funded by your agreement who graduated during this period and will continue to pursue a graduate or Ph.D. degree in science, mathematics, engineering, or technology fields:.....	0.00
Number of graduating undergraduates who achieved a 3.5 GPA to 4.0 (4.0 max scale):	0.00
Number of graduating undergraduates funded by a DoD funded Center of Excellence grant for Education, Research and Engineering:	0.00
The number of undergraduates funded by your agreement who graduated during this period and intend to work for the Department of Defense	0.00
The number of undergraduates funded by your agreement who graduated during this period and will receive scholarships or fellowships for further studies in science, mathematics, engineering or technology fields:	0.00

Names of Personnel receiving masters degrees

NAME
Hyeyoun Chung
Total Number: 1

Names of personnel receiving PHDs

NAME
Richard Low
Total Number: 1

Names of other research staff

NAME	PERCENT SUPPORTED
FTE Equivalent:	
Total Number:	

Sub Contractors (DD882)

Inventions (DD882)

Applications of the Schur Basis to Quantum Algorithms

ARO Project Final Technical Report: 09/01/05 to 08/31/08

Massachusetts Institute of Technology

Contract number: W911NF-05-1-0312

December, 2010

Participating organizations: MIT, University of Bristol

Technical Point of Contact:

Prof. Isaac Chuang
77 Massachusetts Ave
Cambridge, MA 02139
Tel: (617) 253-0905
Fax: (617) 253-0053
E-mail: ichuang@mit.edu

Administrative Point of Contact:

MIT CUA
77 Massachusetts Ave
Cambridge, MA 02139
Tel: (617) 253-6830
Fax: (617) 253-4876
E-mail: j_k@mit.edu

Contents

1 Introduction

Many of the most successful quantum algorithms are designed around symmetries, for which group representation theory provides the mathematical foundation. These algorithms traditionally have achieved their speedups with the quantum Fourier transform (QFT), but this is not the only method known to exploit group symmetries. One concept which has been productive in mathematics, chemistry, physics, and recently quantum information theory, is known as Schur (or Schur-Weyl) duality. Early in this project we gave an efficient quantum circuit, which we call the Schur transform by analogy to the QFT, for transforming quantum data between two different forms: the standard computational basis and the Schur basis. This allows quantum computers to efficently compute using the Schur symmetries of quantum information. While this already has applications to quantum communication, one of our main goals is to find algorithmic uses of the transform. We are also looking at ways of using Schur symmetry in a purely mathematical sense to construct quantum algorithms, so that Schur duality would be used in the analysis of the algorithm but its implementation would not explicitly use the Schur transform.

We report the following major accomplishments over the span of this project timeline, from 09/01/05 to 08/31/08:

- Efficient circuit for the Schur transform devised – *Phys. Rev. Lett.*, vol. 97, pp. 170502, 2006.
- Qudit version of Schur transform devised – *Proc. 18th ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pp. 1235–1244, 2007.
- Schur transform applied to hidden subgroup problem – *Proc. 24th Symposium on Theoretical Aspects of Computer Science (STACS 2007)*, Lecture Notes in Computer Science 4393, pp. 598–609, 2007.
- Quantum expanders developed – *Q. Inf. Comp.*, vol. 8, no. 8/9, pp. 715-721, 2008. [arXiv:0709.1142]
- Analysis of random quantum circuits – *Comm. Math. Phys.* vol. 291, no. 1, pp. 257–302, 2009. [arXiv:0802.1919]
- Study of tensor product expanders – *Q. Inf. Comp.* vol. 9, pp. 336–360, 2009. [arXiv:0804.0011]

- New quantum algorithm for superpolynomial speedups based on quantum circuits – *Proc. of the 35th International Colloquium on Automata, Languages and Programming (ICALP 2008)*, LNCS 5125, pp. 782–795, 2008. [arXiv:0805.0007]

Major publications detailing these results, specifically including those cited here, appear in the appendix of this report.

2 Project motivation and summary of scientific results

The main goal of this project was to develop new quantum algorithms, based on the application of the Schur transform, as a new building block, different from the quantum Fourier transform.

We analyzed a natural strategy for using the Schur transform to solve the hidden subgroup problem (HSP; a common framework for quantum speedups), and found that it failed to give an exponential speedup. Along the way, we gave upper and lower bounds on the complexity of the quantum collision problem, which are tight for oracle complexity and nearly tight for time complexity. (Here we mean a quantum generalization of the classical collision problem from cryptography, in which we want to distinguish a uniform distribution on an unknown N elements from one on an unknown 2N elements.) This work was published in STACS.

We also looked beyond the HSP for problems where quantum computers can exhibit exponential, or at least superpolynomial, speedups over classical computers. Initially, we found a problem which cannot be solved on a classical computer in polynomial time, but which can be solved quantumly using the QFT over the symmetric group, which is closely related to the Schur transform, and for which no previous application was known. By generalizing our construction, we found that these sorts of speedups can in fact be obtained from any efficiently implementable QFT (i.e. over any finite group), or even from most random circuits that are sufficiently long. On the one hand, this shows that the group symmetry was less important than many people initially believed. On the other, it means we have constructed a large class of superpolynomial quantum speedups which look radically unlike the speedups based on the HSP.

One of the building blocks of the Schur transform was a quantum Clebsch-Gordan transform

for the unitary group. If this could be generalized to a quantum Fourier transform on the unitary group, it could have many applications to dealing with unitary symmetries on quantum computers. However, we have not yet been successful at turning classical circuits for the unitary group Fourier transform4 into quantum circuits for the unitary group QFT. The difficulty is that, unlike simpler Fourier transforms, the unitary group Fourier transform involves intermediate steps which, if implemented on a quantum computer, would not preserve the overall normalization of the state. Thus, these transformations would either be physically impossible, or would have an unacceptably high failure rate. This only means that our first approach failed, however, and not that an efficient quantum algorithm is ruled out. We are currently investigating other ways to approach the problem, mostly involving technical changes in how the data is represented, as well as examining different recursive decompositions of the Legendre transform that is at the heart of the problem.

While we have not yet constructed an efficient QFT over the unitary group, we have found one important application that would be made possible by such a QFT. Classically, expander graphs are an extremely useful algorithm tool, with applications in error-correcting codes, network design, probabilistically checkable proofs, pseudorandomness, cryptographic hash functions, and other fields. Only recently, a definition of a quantum expander was proposed, and applications were given to cryptography5 and to condensed matter physics6. However, no efficient implementations of quantum expanders are currently known. We found a method to implement a quantum expander that would be efficient if rotations in high-dimensional irreps of SU(2) could be efficiently simulated on a quantum computer. This task would in turn be efficiently implementable if a QFT over SU(2) could be efficiently carried out on a quantum computer. On the other hand, a direct implementation of rotations in SU(2) irreps was claimed. The method there turns out to be missing some crucial steps, which we have worked to fill in. Doing so gives the only known efficient construction of a quantum expander.

Finally, we also investigated Clebsch-Gordan transforms over groups other than the unitary group, and have constructed explicit efficient circuits for the dihedral and Heisenberg group. Cascading them will allow the construction of circuits that are analogous to the Schur transform, but

with the dihedral or Heisenberg group in place of the unitary group. We have investigated variants of the HSP for which these circuits might be useful.

Our work on superpolynomial speedups also led us to investigate the properties of short random quantum circuits (a.k.a. pseudorandom unitaries), and the extent to which they approximate the behavior of fully random unitary matrices. The superpolynomial speedup mentioned above can be obtained by analyzing the second moment of a family of pseudorandom unitaries (inspired by the techniques in 8), but we expect better constructions and additional applications (efficient methods of randomizing quantum states, or of constructing unknown quantum states from oracles) to arise from studying their higher moments. Ideally, the results would be analogous to the classical case, where polynomial-size random circuits approximate random functions to all orders (in other words, achieving nearly t-wise independence, for any t) as the circuit size increases. For this project, the quantum circuits that are constructed would not explicitly use the Schur transform; instead it is our analysis of the circuit that makes use of Schur duality.

3 List of manuscripts submitted/appearing in print

- *Papers in Refereed Journals and Conferences:*

1. D.A.Bacon, I.L. Chuang, A.W. Harrow. "Efficient Quantum Circuits for Schur and Clebsch-Gordan Transforms." *Phys. Rev. Lett.*, vol. 97, pp. 170502, 2006.
2. D.A.Bacon, I.L. Chuang and A.W. Harrow. "The Quantum Schur Transform: I. Efficient Qudit Circuits." *Proc. 18th ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pp. 1235–1244, 2007.
3. A.M. Childs, A.W. Harrow and P. Wocjan. "Weak Fourier-Schur sampling, the hidden subgroup problem, and the quantum collision problem," *Proc. 24th Symposium on Theoretical Aspects of Computer Science (STACS 2007)*, Lecture Notes in Computer Science 4393, pp. 598–609, 2007.
4. A.W. Harrow. "Quantum expanders from any classical Cayley graph expander." *Q. Inf. Comp.*, vol. 8, no. 8/9, pp. 715-721, 2008. [arXiv:0709.1142]

5. A.W. Harrow and R.A. Low. "Random Quantum Circuits are Approximate 2-designs" *Comm. Math. Phys.* vol. 291, no. 1, pp. 257–302, 2009. [arXiv:0802.1919]

6. M. B. Hastings and A. W. Harrow. "Classical and Quantum Tensor Product Expanders." *Q. Inf. Comp.* vol. 9, pp. 336–360, 2009. [arXiv:0804.0011]

7. S. Hallgren and A.W. Harrow. "Superpolynomial speedups based on almost any quantum circuit." *Proc. of the 35th International Colloquium on Automata, Languages and Programming (ICALP 2008)*, LNCS 5125, pp. 782–795, 2008. [arXiv:0805.0007]

- *Ph.D. and Master's Theses:*

8. Richard Low, Ph.D. Thesis, University of Bristol, Department of Computer Science, 2009

9. Hyeyoun Chung, S.M. Thesis, MIT, Department of EECS, 2008

4 Presentations

- Aram Harrow. IBM Research, Yorktown Heights, NY. January 5, 2007. invited seminar.
- Aram Harrow. ACM-SIAM Symposium on Discrete Algorithms (SODA), New Orleans, LA. 9 January 2007. contributed conference talk.
- Aram Harrow. Caltech Institute for Quantum Information. January 23, 2007. invited seminar.
- Aram Harrow. Quantum Information Processing (QIP), Brisbane, Australia. Jan 30, 2007. contributed conference talk.
- Aram Harrow. National Institute for Informatics, Tokyo, Japan. February 5, 2007. invited seminar.
- Aram Harrow. Univ. of Kwa-Zulu Natal, Durban, S. Africa. April 2, 2007. invited seminar.

- Richard Low. Informal Quantum Information Gathering (IQING 5), Innsbruck, Austria. April 12, 2007. contributed conference talk.
- Richard Low. Taming the Quantum World, Waterloo, ON, Canada. June 4, 2007. contributed conference talk.
- Aram Harrow. Taming the Quantum World, Waterloo, ON, Canada. June 27, 2007. invited conference talk.
- Aram Harrow. Taming the Quantum World, Waterloo, ON, Canada. June 29, 2007. invited conference talk.
- Aram Harrow. Quantum Information Processing (QIP), New Delhi, India. December 20, 2007. contributed conference talk.
- Aram Harrow. CWI, Amsterdam, Netherlands. February 22, 2008. invited seminar.
- Richard Low. February 29, 2008. University of Leeds. invited seminar.
- Aram Harrow. LANL Classical and Quantum Information Theory conference, Santa Fe, NM. March 26, 2008. invited conference talk.
- Aram Harrow. Caltech Institute for Quantum Information. April 3, 2008. invited seminar.
- Aram Harrow. Quantum Information and Graph Theory. Perimeter Institute, Waterloo, ON. April 28, 2008. invited conference talk.
- Aram Harrow. Canadian Institute for Advanced Research. June 4, 2008. invited conference talk.
- Aram Harrow. University of Cambridge. June 24, 2008. invited seminar.
- Aram Harrow. 35th International Colloquium on Automata, Languages and Programming (ICALP 2008). Reyjavik, Iceland. July 10, 2008. contributed conference talk.
- Richard Low. Aug 28, 2008. Asian Conference on Quantum Information Science (AQIS). Seoul, S. Korea. contributed conference talk.

5 Scientific personnel supported and honors/awards/degrees

Isaac Chuang (Prof., MIT)

Aram Harrow (Prof., U. of Bristol)

Richard Low (Ph.D. awarded, 2010)

Hyeyoun Chung (S.B. awarded 2008)

A Reference papers

A.1 CMP Paper

Commun. Math. Phys. 291, 257–302 (2009)
Digital Object Identifier (DOI) 10.1007/s00220-009-0873-6

Communications in
Mathematical Physics

Random Quantum Circuits are Approximate 2-designs

Aram W. Harrow, Richard A. Low

Department of Computer Science, University of Bristol, Bristol, U.K.
E-mail: a.harrow@bristol.ac.uk; low@cs.bris.ac.uk

Received: 20 August 2008 / Accepted: 1 May 2009
Published online: 17 July 2009 – © Springer-Verlag 2009

Abstract: Given a universal gate set on two qubits, it is well known that applying random gates from the set to random pairs of qubits will eventually yield an approximately Haar-distributed unitary. However, this requires exponential time. We show that random circuits of only polynomial length will approximate the first and second moments of the Haar distribution, thus forming approximate 1- and 2-designs. Previous constructions required longer circuits and worked only for specific gate sets. As a corollary of our main result, we also improve previous bounds on the convergence rate of random walks on the Clifford group.

1. Introduction: Pseudo-Random Quantum Circuits

There are many examples of algorithms that make use of random states or unitary operators (e.g. [5,28]). However, exactly sampling from the uniform Haar distribution is inefficient. In many cases, though, only pseudo-random operators are required. To quantify the extent to which the pseudo-random operators behave like the uniform distribution, we use the notion of *k-designs* (often referred to as *t*-designs). A *k*-design has k^{th} moments equal to those of the Haar distribution. For most uses of random states or unitaries, this is sufficient. Constructions of exact *k*-designs on states are known (see [3] and references therein) and some are efficient. Ambainis and Emerson [3] introduced the notion of approximate state *k*-designs, which can be implemented efficiently for any *k*. However, the known constructions of unitary *k*-designs are inefficient to implement. Approximate unitary 2-designs have been considered [10,14,18], although the approaches are specific to 2-designs.

We consider a general class of random circuits where a series of two-qubit gates are chosen from a universal gate set. We give a framework for analysing the k^{th} moments of these circuits. Our conjecture, based on an analogous classical result [23], is that a random circuit on n qubits of length $\text{poly}(n, k)$ is an approximate k-design. While we do not prove this, we instead give a tight analysis of the $k = 2$ case. We find that in a

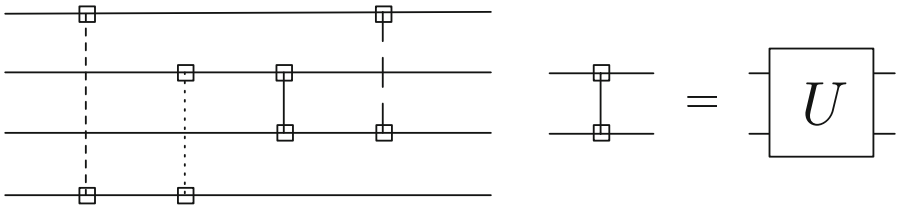

Fig. 1. An example of a random circuit. Different lines indicate a different gate is applied at each step

broad class of natural random circuit models (described in Sect. 1.1), a circuit of length $O(n(n+\log 1/\epsilon))$ yields an ϵ-approximate 2-design. Our definition of an approximate k-design is in Sect. 2.2. Our results also apply to an alternative definition of an approximate 2-design from [10], for which we show random circuits of length $O(n(n+\log 1/\epsilon))$ yield ϵ-approximations, thus extending the results of that paper to a larger class of circuits. Moreover, our results also apply to random stabiliser circuits, meaning that a random stabiliser circuit of length $O(n(n+\log 1/\epsilon))$ will be an ϵ-approximate 2-design. This both simplifies the construction and tightens the efficiency of the approach of [14], which constructed ϵ-approximate 2-designs in time $O(n^6(n^2+\log 1/\epsilon))$ using $O(n^3)$ elementary quantum gates.

1.1. Random Circuits. The random circuit we will use is the following. Choose a 2-qubit gate set that is universal on $U(4)$ (or on the stabiliser subgroup of $U(4)$). One example of this is the set of all one qubit gates together with the controlled-NOT gate. Another is simply the set of all of $U(4)$. Then, at each step, choose a random pair of qubits and apply a gate from the universal set chosen uniformly at random. For the $U(4)$ case, the distribution will be the Haar measure on $U(4)$. One such circuit is shown in Fig. 1 for $n = 4$ qubits. This is based on the approach used in Refs. [9,26] but our analysis is both simpler and more general.

Since the universal set can generate the whole of $U(2^n)$ in this way, such random circuits can produce any unitary. Further, since this process converges to a unitarily invariant distribution and the Haar distribution is unique, the resulting unitary must be uniformly distributed amongst all unitaries [15]. Therefore this process will eventually converge to a Haar distributed unitary from $U(2^n)$. This is proven rigourously in Lemma 3.2. However, a generic element of $U(2^n)$ has 4^n real parameters, and thus to even have $\Omega(4^{-n})$ fidelity with the Haar distribution requires $\Omega(4^n)$ 2-qubit unitaries. We address this problem by considering only the lower-order moments of the distribution and showing these are nearly the same for random circuits as for Haar-distributed unitaries. This claim is formally described in Theorem 2.2.

Our paper is organised as follows. In Sect. 2 we define unitary k-designs and explain how a random circuit could be used to construct a k-design. In Sect. 3 we work out how the state evolves after a single step of the random circuit. We then extend this to multiple steps in Sect. 4 and prove our general convergence results. A key simplification will be (following [26]) to map the evolution of the second moments of the quantum circuit onto a classical Markov chain. We then prove a tight convergence result for the case where the gates are chosen from $U(4)$ in Sect. 5. This section contains most of the technical content of the paper. Using our bounds on mixing time we put together the proof that random circuits yield approximate unitary 2-designs in Sect. 6. Section 7 concludes with some discussion of applications.

2. Preliminaries

2.1. Pauli expansion. Much of the following will be done in the Pauli basis. The Pauli operators will be taken as $\{\sigma_0, \sigma_1, \sigma_2, \sigma_3\}$ and defined to be

$$\sigma_0 = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, \quad \sigma_1 = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \quad \sigma_2 = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}, \quad \sigma_3 = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}.$$

If $|\psi\rangle \in \mathbb{C}^{2^n}$ is a state on n qubits then we write $\psi = |\psi\rangle\langle\psi|$. We can expand ψ in the Pauli basis as

$$\psi = 2^{-n/2} \sum_p \gamma(p)\sigma_p, \tag{2.1}$$

where $\sigma_p = \sigma_{p_1} \otimes \ldots \otimes \sigma_{p_n}$ for the string $p = p_1 \cdots p_n$. Inverting, the coefficients $\gamma(p)$ are given by

$$\gamma(p) = 2^{-n/2} \mathrm{tr}\, \sigma_p \psi. \tag{2.2}$$

It is easy to show that the coefficients $\gamma(p)$ are real and, with the chosen normalisation, the squares sum to $\mathrm{tr}\, \psi^2$, which is 1 for pure ψ. In general

$$\sum_p \gamma^2(p) \leq 1$$

with equality if and only if ψ is pure. Note also that $\mathrm{tr}\, \psi = 1$ is equivalent to $\gamma(0) = 2^{-n/2}$.

This notation is extended to states on nk qubits by treating γ as a function of k strings from $\{0, 1, 2, 3\}^n$. Thus a state ρ on nk qubits is written as

$$\rho = 2^{-nk/2} \sum_{p_1, \ldots, p_k} \gamma_0(p_1, \cdots, p_k)\sigma_{p_1} \otimes \ldots \otimes \sigma_{p_k}. \tag{2.3}$$

2.2. k-designs. We will say that a k-design is efficient if the effort required to sample a state or unitary from the design is polynomial in n and k. Note that we do not require the number of states to be polynomial because, even for approximate unitary designs, an exponential number of unitaries is required. Rather, the number of random bits needed to specify an element of the design should be poly(n, k).

2.2.1. State designs A (state) k-design is an ensemble of states such that, when one state is chosen from the ensemble and copied k times, it is indistinguishable from a uniformly random state. This is a way of quantifying the pseudo-randomness of the state and is a quantum analogue of k-wise independence. Hayashi et al. [20] give an inefficient construction of k-designs for any n and k.

The state k-design definition we use is due to Ref. [3]:

Definition 2.1. *An ensemble of quantum states $\{p_i, |\psi_i\rangle\}$ is a state k-design if*

$$\sum_i p_i \left(|\psi_i\rangle\langle\psi_i|\right)^{\otimes k} = \int_\psi \left(|\psi\rangle\langle\psi|\right)^{\otimes k} d\psi, \tag{2.4}$$

where the integration is taken over the left invariant Haar measure on the unit sphere in $\mathbb{C}^d$, normalised so that $\int_\psi d\psi = 1$.

It is well known that the above integral is equal to $\frac{\Pi_{+k}}{\binom{k+d-1}{k}}$, where Π_{+k} is the projector onto the symmetric subspace of k d-dimensional spaces. For a rigourous proof, see Ref. [16] and for a less precise proof, but from a quantum information perspective, see Ref. [7].

2.2.2. Unitary designs A unitary k-design is, in a sense, a stronger version of a state design. Just as applying a Haar-random unitary to an arbitrary pure state results in a uniformly random pure state, applying a unitary chosen from a unitary k-design to an arbitrary pure state should result in a state k-design. Another way to say this is that the state obtained by acting $U^{\otimes k}$, where U is drawn from a unitary k-design on $U(d)$, on any d^k-dimensional state should be indistinguishable from the case where U is drawn uniformly from $U(d)$. Formally, we have:

Definition 2.2. *Let $\{p_i, U_i\}$ be an ensemble of unitary operators. Define*

$$\mathcal{G}_W(\rho) = \sum_i p_i U_i^{\otimes k} \rho (U_i^\dagger)^{\otimes k} \tag{2.5}$$

and

$$\mathcal{G}_H(\rho) = \int_U U^{\otimes k} \rho (U^\dagger)^{\otimes k} dU. \tag{2.6}$$

Then the ensemble is a unitary k-design iff $\mathcal{G}_W = \mathcal{G}_H$.

Unitary designs can also be defined in terms of polynomials, so that if p is a polynomial with degree k in the matrix elements of U and k in the matrix elements of U^*, then averaging p over a unitary k-design should give the same answer as averaging over the Haar measure. To see the equivalence with Definition 2.2 note that averaging a monomial over our ensemble can be expressed as $\langle i_1, \ldots, i_k | \mathcal{G}_W(|j_1, \ldots, j_k\rangle\langle j'_1, \ldots, j'_k|) | i'_1, \ldots, i'_k \rangle$, and so if $\mathcal{G}_W = \mathcal{G}_H$ then any polynomial of degree k will have the same expectation over both distributions.

2.3. Approximate k-designs.

2.3.1. Approximate state designs Numerous examples of exact efficient state 2-design constructions are known (e.g. [8]) but general exact constructions are not efficient in n and k. Approximate state designs were first introduced by Ambainis and Emerson [3] and they constructed efficient approximate state k-designs for any k. Aaronson [1] also gives an efficient approximate construction.

We define approximate state designs as follows.

Definition 2.3. *An ensemble of quantum states $\{p_i, |\psi_i\rangle\}$ is an ϵ-approximate state k-design if*

$$(1-\epsilon)\int_\psi (|\psi\rangle\langle\psi|)^{\otimes k}\, d\psi \le \sum_i p_i\, (|\psi_i\rangle\langle\psi_i|)^{\otimes k} \le (1+\epsilon)\int_\psi (|\psi\rangle\langle\psi|)^{\otimes k}\, d\psi. \tag{2.7}$$

In [3], a similar definition was proposed but with the additional requirement that the ensemble also forms a 1-design (exactly), i.e.

$$\sum_i p_i |\psi_i\rangle\langle\psi_i| = \int_\psi |\psi\rangle\langle\psi| d\psi.$$

This requirement was necessary there only so that a suitably normalised version of the ensemble would form a POVM. We will not use it.

By taking the partial trace one can show that a k-design is a k'-design for $k' \leq k$. Thus approximate k-designs are always at least approximate 1-designs.

2.3.2. Approximate unitary designs It was shown in Ref. [4] that a quantum analogue of a one time pad requires $2n$ bits to exactly randomise an n qubit state. However, in Ref. [5] it was shown that $n + o(n)$ bits suffice to do this approximately. Translated into k-design language, this says an exact unitary 1-design requires 2^{2n} unitaries but can be done approximately with $2^{n+o(n)}$. So approximate designs can have fewer unitaries than exact designs. Here, we are interested in improving the efficiency of implementing the unitaries. There are no known efficient exact constructions of unitary k-designs; it is hoped that our approach will yield approximate unitary designs efficiently.

We will require approximate unitary k-designs to be close in the diamond norm [24]:

Definition 2.4. *The diamond norm of a superoperator T,*

$$||T||_\diamond = \sup_d ||T \otimes id_d||_\infty = \sup_d \sup_{X \neq 0} \frac{||(T \otimes id_d)X||_1}{||X||_1},$$

where id_d is the identity channel on d dimensions.

Operationally, the diamond norm of the difference between two quantum operations tells us the largest possible probability of distinguishing the two operations if we are allowed to have them act on part of an arbitrary, possibly entangled, state. In the supremum over ancilla dimension d, it can be shown that d never needs to be larger than the dimension of the system that T acts upon. The diamond norm is closely related to completely bounded norms (cb-norms), in that $||T||_\diamond$ is the cb-norm of $T^\dagger$ and can also be interpreted as the $L_1 \to L_1$ cb-norm of T itself [11,27].

We can now define approximate unitary k-designs.

Definition 2.5. *$\mathcal{G}_W$ is an ϵ-approximate unitary k-design if*

$$||\mathcal{G}_W - \mathcal{G}_H||_\diamond \leq \epsilon, \tag{2.8}$$

where $\mathcal{G}_W$ and $\mathcal{G}_H$ are defined in Definition 2.2.

In Ref. [10], they consider approximate twirling, which is implemented using an approximate 2-design. They give an alternative definition of closeness which is more convenient for this application:

Definition 2.6 ([10]). *Let $\{p_i, U_i\}$ be an ensemble of unitary operators. Then this ensemble is an ϵ-approximate twirl if*

$$\max_\Lambda \left|\left| \mathbb{E}_W W(\Lambda(W^\dagger \rho W))W^\dagger - \mathbb{E}_U U(\Lambda(U^\dagger \rho U))U^\dagger \right|\right|_\diamond \leq \frac{\epsilon}{d^2}, \tag{2.9}$$

where the first expectation is over W chosen from the ensemble and the second is the Haar average. The maximisation is over channels Λ and d is the dimension (2^n in our case).

Our results work for both definitions with the same efficiency.

2.4. Random Circuits as k-designs. If a random circuit is to be an approximate k-design then Eq. 2.8 must be satisfied where the U_i are the different possible random circuits. We can think of this as applying the random circuit not once but k times to k different systems.

Suppose that applying t random gates yields the random circuit W. If $W^{\otimes k}$ acts on an nk-qubit state ρ, then following the notation of Eq. 2.8, the resulting state is

$$\rho_W := W^{\otimes k}\rho(W^\dagger)^{\otimes k} = 2^{-nk/2}\sum_{p_1,\ldots,p_k}\gamma_0(p_1,\ldots,p_k)W\sigma_{p_1}W^\dagger\otimes\cdots\otimes W\sigma_{p_k}W^\dagger. \tag{2.10}$$

For this to be a k-design, the expectation over all choices of random circuit should match the expectation over Haar-distributed $W \in U(2^n)$.

We are now ready to state our main results. Our results apply to a large class of gate sets which we define below:

Definition 2.7. *Let $\mathcal{E} = \{p_i, U_i\}$ be a discrete ensemble of elements from $U(d)$. Define an operator $G_\mathcal{E}$ by*

$$G_\mathcal{E} := \sum_i p_i U_i^{\otimes k}\otimes(U_i^*)^{\otimes k}. \tag{2.11}$$

More generally, we can consider continuous distributions. If μ is a probability measure on $U(d)$ then we can define G_μ by analogy as

$$G_\mu := \int_{U(d)} d\mu(U)U^{\otimes k}\otimes(U^*)^{\otimes k}. \tag{2.12}$$

Then $\mathcal{E}$ (or μ) is k-copy gapped if $G_\mathcal{E}$ (or G_μ) has only $k!$ eigenvalues with absolute value equal to 1.

For any discrete ensemble $\mathcal{E} = \{p_i, U_i\}$, we can define a measure $\mu = \sum_i p_i\delta_{U_i}$. Thus, it suffices to state our theorems in terms of μ and G_μ.

The condition on G_μ in the above definition may seem somewhat strange. We will see in Sect. 3 that when $d \geq k$ there is a $k!$-dimensional subspace of $(\mathbb{C}^d)^{\otimes 2k}$ that is acted upon trivially by any G_μ. Additionally, when μ is the Haar measure on $U(d)$ then G_μ is the projector onto this space. Thus, the k-copy gapped condition implies that vectors orthogonal to this space are shrunk by G_μ.

We will see that G_μ is k-copy gapped in a number of important cases. First, we give a definition of universality that can apply not only to discrete gates sets, but to arbitrary measures on $U(4)$.

Definition 2.8. *Let μ be a distribution on $U(4)$. Suppose that for any open ball $S \subset U(4)$ there exists a positive integer ℓ such that $\mu^{\star\ell}(S) > 0$. Then we say μ is universal [for $U(4)$].*

Here $\mu^{\star\ell}$ is the ℓ-fold convolution of μ with itself; i.e.

$$\mu^{\star\ell} = \int \delta_{U_1\cdots U_\ell}d\mu(U_1)\cdots d\mu(U_\ell).$$

When μ is a discrete distribution over a set $\{U_i\}$, Definition 2.8 is equivalent to the usual definition of universality for a finite set of unitary gates.

Theorem 2.1. *The following distributions on $U(4)$ are k-copy gapped:*

(i) *Any universal gate set. Examples are $U(4)$ itself, any entangling gate together with all single qubit gates, or the gate set considered in [26].*
(ii) *Any approximate (or exact) unitary k-design on 2 qubits, such as the uniform distribution over the 2-qubit Clifford group, which is an exact 2-design.*

Proof.

(i) This is proven in Lemma 3.2.
(ii) This follows straight from Definition 2.2. □

Theorem 2.2. *Let μ be a 2-copy gapped distribution and W be a random circuit on n qubits obtained by drawing t random unitaries according to μ and applying each of them to a random pair of qubits. Then there exists C (depending only on μ) such that for any $\epsilon > 0$ and any $t \geq C(n(n + \log 1/\epsilon))$, $\mathcal{G}_W$ is an ϵ-approximate unitary 2-design according to either Definition 2.5 or Definition 2.6.*

To prove Theorem 2.2, we show that the second moments of the random circuits converge quickly to those of a uniform Haar distributed unitary. For W a circuit as in Theorem 2.2, write $\gamma_W(p_1, p_2)$ for the Pauli coefficients of $\rho_W = W^{\otimes 2}\rho\left(W^\dagger\right)^{\otimes 2}$. Then write $\gamma_t(p_1, p_2) = \mathbb{E}_W \gamma_W(p_1, p_2)$ where W is a circuit of length t. Then we have

Lemma 2.1. *Let μ and W be as in Theorem 2.2. Let the initial state be ρ with $\gamma_0(p, p) \geq 0$ and $\sum_p \gamma_0(p, p) = 1$ (for example the state $|\psi\rangle\langle\psi| \otimes |\psi\rangle\langle\psi|$ for any pure state $|\psi\rangle$). Then there exists a constant C (possibly depending on μ) such that for any $\epsilon > 0$,*

(i)

$$\sum_{\substack{p_1, p_2 \\ p_1 p_2 \neq 00}} \left(\gamma_t(p_1, p_2) - \delta_{p_1 p_2} \frac{1}{2^n(2^n + 1)}\right)^2 \leq \epsilon \tag{2.13}$$

for $t \geq Cn \log 1/\epsilon$.

(ii)

$$\sum_{\substack{p_1, p_2 \\ p_1 p_2 \neq 00}} \left|\gamma_t(p_1, p_2) - \delta_{p_1 p_2} \frac{1}{2^n(2^n + 1)}\right| \leq \epsilon \tag{2.14}$$

for $t \geq Cn(n + \log 1/\epsilon)$ or, when μ is the uniform distribution on $U(4)$ or its stabiliser subgroup, $t \geq Cn \log \frac{n}{\epsilon}$.

We can then extend this to all states by a simple corollary:

Corollary 2.1. *Let μ, W and γ_W be as in Lemma 2.1. Then, for any initial state $\rho = \frac{1}{2^n}\sum_{p_1, p_2} \gamma_0(p_1, p_2)\sigma_{p_1} \otimes \sigma_{p_2}$, there exists a constant C (possibly depending on μ) such that for any $\epsilon > 0$,*

(i)

$$\sum_{\substack{p_1, p_2 \\ p_1 p_2 \neq 00}} \left(\gamma_t(p_1, p_2) - \delta_{p_1 p_2} \frac{\sum_{p \neq 0} \gamma_0(p, p)}{4^n - 1}\right)^2 \leq \epsilon \tag{2.15}$$

for $t \geq Cn(n + \log 1/\epsilon)$.

(ii)

$$\sum_{\substack{p_1,p_2\\ p_1p_2\neq 00}} \left| \gamma_t(p_1,p_2) - \delta_{p_1p_2} \frac{\sum_{p\neq 0}\gamma_0(p,p)}{4^n-1} \right| \leq \epsilon \tag{2.16}$$

for $t \geq Cn(n+\log 1/\epsilon)$.

By the usual definition of an approximate design (Definition 2.5), we only need convergence in the 2-norm (Eq. 2.15), which is implied by 1-norm convergence (Eq. 2.16) but weaker. However, Definition 2.6, which requires the map to be close to the twirling operation, requires 1-norm convergence (i.e. Eq. 2.16). Thus, Theorem 2.2 for Definition 2.5 follows from Corollary 2.1(i) and Theorem 2.2 for Definition 2.6 follows from Corollary 2.1(ii). Theorem 2.2 is proved in Sect. 6 and Corollary 2.1 in Sect. 4.

We note that, in the course of proving Lemma 2.1, we prove that the eigenvalue gap (defined in Sect. 4.3) of the Markov chain that gives the evolution of the $\gamma(p,p)$ terms is $O(1/n)$. It is easy to show that this bound is tight for some gate sets.

Related work. Here we summarise the other efficient constructions of approximate unitary 2-designs.

- The uniform distribution over the Clifford group on n qubits is an exact 2-design [14]. Moreover, [14] described how to sample from the Clifford group using $O(n^8)$ classical gates and $O(n^3)$ quantum gates. Our results show that applying $O(n(n+\log 1/\epsilon))$ random two-qubit Clifford gates also achieve an ϵ-approximate 2-design (although not necessarily a distribution that is within ϵ of uniform on the Clifford group).
- Dankert et al. [10] gave a specific circuit construction of an approximate 2-design. To achieve small error in the sense of Definition 2.5, their circuits require the same $O(n(n+\log 1/\epsilon))$ gates that our random circuits do. However, when we use Definition 2.6, the circuits from [10] only need $O(n\log 1/\epsilon)$ gates while the random circuits analysed in this paper need to be length $O(n(n+\log 1/\epsilon))$.
- The closest results to our own are in the papers by Oliveira et al. [9,26], which considered a specific gate set (random single qubit gates and a controlled-NOT) and proved that the second moments converge in time $O(n^2(n+\log 1/\epsilon))$. Our strategy of analysing random quantum circuits in terms of classical Markov chains is also adapted from [9,26]. In Sect. 3, we generalise this approach to analyse the k^{th} moments for arbitrary k.

  The main results of our paper extend the results of [9,26] to a larger class of gate sets and improve their convergence bounds. Some of these improvements have been conjectured by [30], which presented numerical evidence in support of them.

3. Analysis of the Moments

In order to prove our results, we need to understand how the state evolves after each step of the random circuit. In this section we consider just one step and a fixed pair of qubits. Later on we will extend this to prove convergence results for multiple steps with random pairs of qubits drawn at every step. We consider first the Haar distribution over the full unitary group and then will discuss the more general case of any 2-copy gapped distribution.

In this section, we work in general dimension d and with a general Hermitian orthogonal basis $\sigma_0,\ldots,\sigma_{d^2-1}$. Later we will take d to be either 4 or 2^n and the σ_i to be

Pauli matrices. However, in this section we keep the discussion general to emphasise the potentially broader applications.

Fix an orthonormal basis for $d \times d$ Hermitian matrices: $\sigma_0, \ldots, \sigma_{d^2-1}$, normalised so that $\mathrm{tr}\,\sigma_p\sigma_q = d\,\delta_{p,q}$. Let σ_0 be the identity. We need to evaluate the quantity

$$\mathbb{E}_U \left(U^{\otimes k} \sigma_{p_1} \otimes \cdots \otimes \sigma_{p_k} (U^\dagger)^{\otimes k} \right) =: T(\mathbf{p}), \tag{3.1}$$

where the expectation is over Haar distributed $U \in U(d)$. We will need this quantity in two cases. Firstly, for $d = 2^n$, these are the moments obtained after applying a uniformly distributed unitary so we know what the random circuit must converge to. Secondly, for $d = 4$, this tells us how a random $U(4)$ gate acts on any chosen pair.

Call the quantity in Eq. 3.1 $T(\mathbf{p})$ (we use **bold** to indicate a k-tuple of coefficients; take $\mathbf{p} = (p_1, \ldots, p_k)$) and write it in the σ_p basis as

$$T(\mathbf{p}) = \sum_{\mathbf{q}} \hat{G}(\mathbf{q}; \mathbf{p}) \sigma_{q_1} \otimes \cdots \otimes \sigma_{q_k}. \tag{3.2}$$

Here, $\hat{G}(\mathbf{q}; \mathbf{p})$ is the coefficient in the Pauli expansion of $T(\mathbf{p})$ and we define $\hat{G}$ as the matrix with entries equal to $\hat{G}(\mathbf{q}; \mathbf{p})$. We have left off the usual normalisation factor because, as we shall see, with this normalisation $\hat{G}$ is a projector. Inverting this, we have

$$\begin{aligned} \hat{G}(\mathbf{q}; \mathbf{p}) &= d^{-k} \mathrm{tr} \left(\sigma_{q_1} \otimes \cdots \otimes \sigma_{q_k} T(\mathbf{p}) \right) \\ &= d^{-k} \mathbb{E}_U \mathrm{tr} \left((\sigma_{q_1} \otimes \cdots \otimes \sigma_{q_k}) U^{\otimes k} (\sigma_{p_1} \otimes \cdots \otimes \sigma_{p_k}) (U^\dagger)^{\otimes k} \right). \end{aligned} \tag{3.3}$$

Note that $\hat{G}$ is real since T and the basis are Hermitian.

We can gain all the information we need about the Haar integral in Eq. 3.1 with the following observations:

Lemma 3.1. *$T(\mathbf{p})$ commutes with $U^{\otimes k}$ for any unitary U.*

Proof. Follows from the invariance of the Haar measure on the unitary group.

Corollary 3.1. *$T(\mathbf{p})$ is a linear combination of permutations from the symmetric group S_k.*

Proof. This follows from Schur-Weyl duality (see e.g. [16]).

From this, we can prove that $\hat{G}$ is a projector and find its eigenvectors.

Theorem 3.1. *$\hat{G}$ is symmetric, i.e. $\hat{G}(\mathbf{q}; \mathbf{p}) = \hat{G}(\mathbf{p}; \mathbf{q})$.*

Proof. Follows from the invariance of the trace under cyclic permutations.

Theorem 3.2. *P_π is an eigenvector of $\hat{G}$ with eigenvalue 1 for any permutation operator P_π i.e.*

$$\sum_{\mathbf{q}} \hat{G}(\mathbf{p}; \mathbf{q}) \mathrm{tr} \left(\sigma_{q_1} \otimes \cdots \otimes \sigma_{q_k} P_\pi \right) = \mathrm{tr} \left(\sigma_{p_1} \otimes \cdots \otimes \sigma_{p_k} P_\pi \right).$$

Further, any vector orthogonal to this set has eigenvalue 0.

Proof. For the first part,

$$\begin{aligned}
&\sum_{\mathbf{q}} \hat{G}(\mathbf{p};\mathbf{q})\mathrm{tr}\left(\sigma_{q_1}\otimes\ldots\otimes\sigma_{q_k}P_\pi\right)\\
&\quad= d^{-k}\sum_{\mathbf{q}}\mathbb{E}_U\mathrm{tr}\left(\sigma_{q_1}U\sigma_{p_1}U^\dagger\right)\ldots\mathrm{tr}\left(\sigma_{q_k}U\sigma_{p_k}U^\dagger\right)\mathrm{tr}\left(\sigma_{q_1}\otimes\ldots\otimes\sigma_{q_k}P_\pi\right)\\
&\quad= d^{-k}\mathrm{tr}\left(P_\pi\mathbb{E}_U\sum_{q_1}\mathrm{tr}\left(\sigma_{q_1}U\sigma_{p_1}U^\dagger\right)\sigma_{q_1}\otimes\ldots\otimes\sum_{q_k}\mathrm{tr}\left(\sigma_{q_k}U\sigma_{p_k}U^\dagger\right)\sigma_{q_k}\right) \qquad (3.4)
\end{aligned}$$

Writing $U^\dagger\sigma_p U$ in the σ_p basis, we find

$$\frac{1}{d}\sum_q \mathrm{tr}\left(\sigma_q U\sigma_p U^\dagger\right)\sigma_q = U\sigma_p U^\dagger.$$

Therefore Eq. 3.4 becomes

$$\mathrm{tr}\left(P_\pi\mathbb{E}_U U^\dagger\sigma_{p_1}U\otimes\ldots\otimes U^\dagger\sigma_{p_k}U\right)=\mathrm{tr}\left(\sigma_{p_1}\otimes\ldots\otimes\sigma_{p_k}P_\pi\right).$$

For the second part, consider any vector v which is orthogonal to the permutation operators (we can neglect the complex conjugate because P_π is real in this basis), i.e.

$$\sum_{\mathbf{q}}\mathrm{tr}\left(\sigma_{q_1}\otimes\cdots\otimes\sigma_{q_k}P_\pi\right)v(\mathbf{q})=0 \qquad (3.5)$$

for any permutation π. Then

$$\sum_{\mathbf{q}}\hat{G}(\mathbf{p};\mathbf{q})v(\mathbf{q})=d^{-k}\sum_{\mathbf{q}}\mathrm{tr}\left(\sigma_{q_1}\otimes\cdots\otimes\sigma_{q_k}T(\mathbf{p})\right)v(\mathbf{q})$$

which is zero since $T(\mathbf{p})$ is a linear combination of permutations and v is orthogonal to this by Eq. 3.5. □

Theorem 3.3. $\hat{G}^2=\hat{G}$, *i.e.* $\sum_{\mathbf{q}'}\hat{G}(\mathbf{p};\mathbf{q}')\hat{G}(\mathbf{q}';\mathbf{q})=\hat{G}(\mathbf{p};\mathbf{q})$.

Proof. Using Eq. 3.3,

$$\sum_{\mathbf{q}'}\hat{G}(\mathbf{p};\mathbf{q}')\hat{G}(\mathbf{q}';\mathbf{q})=\sum_{\mathbf{q}'}\hat{G}(\mathbf{p};\mathbf{q}')d^{-k}\mathrm{tr}\left(\sigma_{q_1'}\otimes\cdots\otimes\sigma_{q_k'}T(\mathbf{q})\right).$$

From Corollary 3.1, $T(\mathbf{q})$ is a linear combination of permutations. This implies, using Theorem 3.2 that

$$\begin{aligned}
\sum_{\mathbf{q}'}\hat{G}(\mathbf{p};\mathbf{q}')d^{-k}\mathrm{tr}\left(\sigma_{q_1'}\otimes\ldots\otimes\sigma_{q_k'}T(\mathbf{q})\right)&=d^{-k}\mathrm{tr}\left(\sigma_{p_1}\otimes\cdots\otimes\sigma_{p_k}T(\mathbf{q})\right)\\
&=\hat{G}(\mathbf{p};\mathbf{q})
\end{aligned}$$

as required. □

Corollary 3.2. $\hat{G}$ *is a projector so has eigenvalues* 0 *and* 1.

We now evaluate $\hat{G}$ and T for the cases of $k=1$ and $k=2$ since these are the cases we are interested in for the remainder of the paper.

3.1. $k = 1$. The $k = 1$ case is clear: the random unitary completely randomises the state. Therefore all terms in the expansion are set to zero apart from the identity i.e.

$$T(p) = \begin{cases} \sigma_0 & p = 0 \\ 0 & p \neq 0. \end{cases} \tag{3.6}$$

3.2. $k = 2$. For $k = 2$, there are just two permutation operators, identity I and swap $\mathcal{F}$. Therefore there are just two eigenvectors with non-zero eigenvalue ($n > 1$). In normalised form, taking them to be orthogonal, their components are

$$f_1(q_1, q_2) = \delta_{q_1 0}\delta_{q_2 0},$$
$$f_2(q_1, q_2) = \frac{1}{d^2 - 1}\delta_{q_1 q_2}(1 - \delta_{q_1 0}).$$

We will now prove three properties of $\hat{G}$ that we need:

1. $\hat{G}(p_1, p_2; q_1, q_2) = 0$ if $p_1 \neq p_2$ or $q_1 \neq q_2$.

   Proof. Consider the function $f(q_1, q_2) = \delta_{q_1 a}\delta_{q_2 b}$ with $a \neq b$. This function has zero overlap with the eigenvectors f_1 and f_2 so it goes to zero when acted on by $\hat{G}$. Therefore $\hat{G}(p_1, p_2; a, b) = 0$. The claim follows from the symmetry property (Theorem 3.1). □

   With this we will write $\hat{G}(p; q) \equiv \hat{G}(p_1, p_2; q_1, q_2)$.
2. $\hat{G}(p; 0) = \delta_{p0}$.

   Proof. Let $\hat{G}$ act on eigenvector f_1. □
3. $\hat{G}(p; a) = \frac{1}{d^2-1}$ for $a, p \neq 0$.

   Proof. Let $\hat{G}$ act on the input δ_{qa}. This has zero overlap with f_1 and overlap $\frac{1}{d^2-1}$ with f_2. □

Therefore we have

$$\hat{G}(p_1, p_2; q_1, q_2) = \begin{cases} 0 & p_1 \neq p_2 \text{ or } \mathrm{q}_1 \neq \mathrm{q}_2 \\ 1 & p_1 = p_2 = q_1 = q_2 = 0 \\ \frac{1}{d^2-1} & p_1 = p_2 \neq 0, q_1 = q_2 \neq 0 \end{cases}. \tag{3.7}$$

Since $T(p_1, p_2) = \sum_{q_1, q_2} \hat{G}(p_1, p_2; q_1, q_2)\sigma_{q_1} \otimes \sigma_{q_2}$, we have

$$T(p_1, p_2) = \begin{cases} 0, & p_1 \neq p_2, \\ \sigma_0 \otimes \sigma_0, & p_1 = p_2 = 0, \\ \frac{1}{d^2-1}\sum_{p' \neq 0} \sigma_{p'} \otimes \sigma_{p'} & p_1 = p_2 \neq 0. \end{cases} \tag{3.8}$$

Therefore the terms $\sigma_{p_1} \otimes \sigma_{p_2}$ with $p_1 \neq p_2$ are set to zero. Further, the sum of the diagonal coefficients $\gamma(p, p)$ is conserved. This allows us to identify this with a probability distribution (after renormalising) and use Markov chain analysis. To see this, write again the starting state

$$\rho = \frac{1}{d}\sum_{q_1, q_2} \gamma_0(q_1, q_2)\sigma_{q_1} \otimes \sigma_{q_2}$$

with the state after application of any unitary W,

$$\rho_W = \frac{1}{d}\sum_{q_1,q_2} \gamma_W(q_1,q_2)\sigma_{q_1}\otimes\sigma_{q_2} = 2^{-n}\sum_{q_1,q_2}\gamma(q_1,q_2)\left(W\sigma_{q_1}W^\dagger\right)\otimes\left(W\sigma_{q_2}W^\dagger\right).$$

Then

$$\begin{aligned}\sum_q \gamma_W(q,q) &= \frac{1}{d}\sum_q \text{tr}\left(\sigma_q\otimes\sigma_q\rho_W\right)\\ &= \text{tr}\left(\mathcal{F}\rho_W\right)\\ &= \frac{1}{d}\sum_{q_1,q_2}\gamma(q_1,q_2)\text{tr}\left(\mathcal{F}\left(W\sigma_{q_1}W^\dagger\right)\otimes\left(W\sigma_{q_2}W^\dagger\right)\right)\\ &= \frac{1}{d}\sum_{q_1,q_2}\gamma(q_1,q_2)\text{tr}\left(\sigma_{q_1}\sigma_{q_2}\right)\\ &= \sum_q \gamma(q,q)\end{aligned}$$

as required, where $\mathcal{F}$ is the swap operator and we have used Lemmas A.2 and A.1.

3.3. Moments for general universal random circuits. We now consider universal distributions μ that in general may be different from the uniform (Haar) measure on $U(d)$. Our main result in this section will be to show that a universal distribution on $U(4)$ is also 2-copy gapped. In fact, we will phrase this result in slightly more general terms and show that a universal distribution on $U(d)$ is also k-copy gapped for any k. Universality (Definition 2.8) generalises in the obvious way to $U(d)$, whereas when we say that μ is k-copy gapped, we mean that

$$\|G_\mu - G_{U(d)}\|_\infty < 1, \tag{3.9}$$

where $G_? = \mathbb{E}_U U^{\otimes k}\otimes(U^*)^{\otimes k}$, with the expectation taken over μ for G_μ or over the Haar measure for $G_{U(d)}$.

The reason Eq. 3.9 represents our condition for μ to be k-copy gapped is as follows: Observe that $\hat{G}$ and G are unitarily related, so the definition of k-copy gapped could equivalently be given in terms of $\hat{G}$. We have shown above that $\hat{G}_{U(d)}$ (and thus $G_{U(d)}$) has all eigenvalues equal to 0 or 1; i.e. is a projector. By contrast, G_μ may not even be Hermitian. However, we will prove below that all eigenvectors of $G_{U(d)}$ with eigenvalue 1 are also eigenvectors of G_μ with eigenvalue 1. Thus, Eq. 3.9 will imply that $\lim_{t\to\infty}(\hat{G}_\mu)^t = \hat{G}_{U(d)}$, just as we would expect for a gapped random walk.

We would like to show that Eq. 3.9 holds whenever μ is universal. This result was proved in [6] (and was probably known even earlier) when μ had the form $(\delta_{U_1}+\delta_{U_2})/2$. Here we show how to extend the argument to any universal μ.

Lemma 3.2. *Let μ be a distribution on $U(d)$. Then all eigenvectors of $G_{U(d)}$ with eigenvalue 1 are eigenvectors of G_μ with eigenvalue one. Additionally, if μ is universal then μ is k-copy gapped for any positive integer k (cf. Eq. 3.9).*

In particular, if $k = 2$ this lemma implies that μ is 2-copy gapped (cf. Theorem 2.1).

Proof. Let $V \cong \mathbb{C}^d$ be the fundamental representation of $U(d)$, where the action of $U \in U(d)$ is simply U itself. Let V^* be its dual representation, where U acts as U^*. The operators G_μ and $G_{U(d)}$ act on the space $V^{\otimes k} \otimes (V^*)^{\otimes k}$. We will see that $G_{U(d)}$ is completely determined by the decomposition of $V^{\otimes k} \otimes (V^*)^{\otimes k}$ into irreducible representations (irreps). Suppose that the multiplicity of (r_λ, V_λ) in $V^{\otimes k} \otimes (V^*)^{\otimes k}$ is m_λ, where the V_λ's are the irrep spaces and $r_\lambda(U)$ the corresponding representation matrices. In other words

$$V^{\otimes k} \otimes (V^*)^{\otimes k} \cong \bigoplus_\lambda V_\lambda \otimes \mathbb{C}^{m_\lambda}, \tag{3.10}$$

$$U^{\otimes k} \otimes (U^*)^{\otimes k} \sim \sum_\lambda |\lambda\rangle\langle\lambda| \otimes r_\lambda(U) \otimes I_{m_\lambda}. \tag{3.11}$$

Here $\sim$ indicates that the two sides are related by conjugation by a fixed (U-independent) unitary.

Let $\lambda = 0$ denote the trivial irrep: i.e. $V_0 = \mathbb{C}$ and $r_0(U) = 1$ for all U. We claim that $\mathbb{E}_U r_\lambda(U) = 0$ whenever $\lambda \neq 0$ and the expectation is taken over the Haar measure. To show this, note that $\mathbb{E}_U r_\lambda(U)$ commutes with $r_\lambda(V)$ for all $V \in U(d)$ and thus, by Schur's Lemma, we must have $\mathbb{E}_U r_\lambda(U) = cI$ for some $c \in \mathbb{C}$. However, by the translation-invariance of the Haar measure we have $cI = \mathbb{E}_U r_\lambda(U) = \mathbb{E}_U r_\lambda(UV) = c\, r_\lambda(V)$ for all $V \in U(d)$. Since $\lambda \neq 0$, we cannot have $r_\lambda(V) = I$ for all V and so it must be that $c = 0$.

Thus, if we write $G_{U(d)}$ and G_μ using the basis on the RHS of Eq. 3.11, we have

$$G_{U(d)} = |0\rangle\langle 0| \otimes I_{m_0}, \tag{3.12}$$

where $|0\rangle\langle 0|$ is a projector onto the trivial irrep. On the other hand,

$$G_\mu = |0\rangle\langle 0| \otimes I_{m_0} + \sum_{\lambda \neq 0} |\lambda\rangle\langle\lambda| \otimes \left(\int r_\lambda(U) d\mu(U) \right) \otimes I_{m_\lambda}. \tag{3.13}$$

Thus, every eigenvector of $G_{U(d)}$ with eigenvalue one is also fixed by G_μ. For the remainder of the space, the direct sum structure means that

$$\|G_{U(d)} - G_\mu\|_\infty = \max_{\substack{\lambda \neq 0 \\ m_\lambda \neq 0}} \left\| \int r_\lambda(U) d\mu(U) \right\|_\infty. \tag{3.14}$$

Note that this maximisation only includes λ with $\dim V_\lambda > 1$. This is because non-trivial one-dimensional irreps of $U(d)$ have the form $\det U^m$ for some non-zero integer m. Under the map $U \mapsto e^{i\phi} U$, such irreps pick up a phase of $e^{im\phi}$. However, $U^{\otimes k} \otimes (U^*)^{\otimes k}$ is invariant under $U \mapsto e^{i\phi} U$. Thus $V^{\otimes k} \otimes (V^*)^{\otimes k}$ cannot contain any non-trivial one-dimensional irreps.

Now suppose by contradiction that there exists $\lambda \neq 0$ with $m_\lambda \neq 0$ and $\| \int r_\lambda(U) d\mu(U) \|_\infty = 1$. (We do not need to consider the case $\| \int r_\lambda(U) d\mu(U) \|_\infty > 1$, since $\|r_\lambda(U)\|_\infty = 1$ for all U and $\| \cdot \|_\infty$ obeys the triangle inequality.) Indeed, the triangle inequality further implies that there exists a unit vector $|v\rangle \in V_\lambda$ such that

$$\int d\mu(U)\, r_\lambda(U) |v\rangle = \omega |v\rangle,$$

for some $\omega \in \mathbb{C}$ with $|\omega| = 1$.

By the above argument we can assume that $\dim V_\lambda > 1$. Since V_λ is irreducible, it cannot contain a one-dimensional invariant subspace, implying that there exists $U_0 \in U(d)$ such that

$$|\langle v|r_\lambda(U_0)|v\rangle| = 1 - \delta,$$

for some $\delta > 0$. Since $U \mapsto |\langle v|r_\lambda(U)|v\rangle|$ is continuous, there exists an open ball S around U_0 such that $|\langle v|r_\lambda(U)|v\rangle| \leq 1 - \delta/2$ for all $U \in S$. Define $\bar{S} := U(d)\backslash S$.

Now we use the fact that μ is universal to find an ℓ such that $\mu^{\star\ell}(S) > 0$. Next, observe that $\int d\mu^{\star\ell}(U)\, \langle v|r_\lambda(U)|v\rangle = \omega^\ell$. Taking the absolute value of both sides yields

$$\begin{aligned}
1 &= \left| \int_{U(d)} d\mu^{\star\ell}(U)\, \langle v|r_\lambda(U)|v\rangle \right| \\
&\leq \int_{U(d)} d\mu^{\star\ell}(U)\, |\langle v|r_\lambda(U)|v\rangle| \\
&= \int_S d\mu^{\star\ell}(U)\, |\langle v|r_\lambda(U)|v\rangle| + \int_{\bar{S}} d\mu^{\star\ell}(U)\, |\langle v|r_\lambda(U)|v\rangle| \\
&\leq \mu^{\star\ell}(S)\left(1 - \frac{\delta}{2}\right) + \left(1 - \mu^{\star\ell}(S)\right) \\
&< 1,
\end{aligned}$$

a contradiction. We conclude that $\|G_{U(d)} - G_\mu\|_\infty < 1$.

4. Convergence

In Sect. 3 we saw that iterating any universal gate set on $U(d)$ eventually converges to the uniform distribution on $U(d)$. Since the set of all two-qubit unitaries is universal on $U(2^n)$, this implies that random circuits eventually converge to the Haar measure. In this section, we turn to proving upper bounds on this convergence rate, focusing on the first two moments.

Let $\hat{G}^{(ij)}$ be the matrix with $\hat{G}$ (with $d = 4$) acting on qubits i and j and the identity on the others. Then, if the pair (i, j) is chosen at step t, we can find the coefficients at step $t + 1$ by multiplying by $\hat{G}^{(ij)}$. In general, a random pair is chosen at each step. So

$$\gamma_{t+1}(\mathbf{p}) = \sum_{\mathbf{q}} \frac{1}{n(n-1)} \sum_{i \neq j} \hat{G}^{(ij)}(\mathbf{p}; \mathbf{q})\gamma_t(\mathbf{q}), \tag{4.1}$$

where γ_{t+1} are the expected coefficients at step t. We can think of this evolution as repeated application of the matrix

$$P = \frac{1}{n(n-1)} \sum_{i \neq j} \hat{G}^{(ij)}. \tag{4.2}$$

For $k = 2$, the key idea of Oliveira et al. [26] was to map the evolution of the $\gamma(p, p)$ coefficients to a Markov chain. The $\gamma(p_1, p_2)$ coefficients with $p_1 \neq p_2$ just decay as each qubit is chosen and can be analysed directly.

However, we can only map the $\gamma(p, p)$ coefficients to a probability distribution when they are non-negative, which is not the case for general states. Most of the rest of the paper is dedicated to proving Lemma 2.1, which only applies to states with $\gamma(p, p) \geq 0$ and normalised so their sum is 1. Corollary 2.1 then extends this to all states:

Proof (of Corollary 2.1). Lemma 2.1 still applies to the $\gamma(p_1, p_2)$ terms with $p_1 \neq p_2$. Therefore we just need to show how to apply Lemma 2.1 to states that initially have some negative $\gamma(p, p)$ terms.

For the $\gamma(p, p)$ terms, Lemma 2.1 says that the random walk starting with any initial probability distribution converges to uniform in some bounded time t. Let $g_t(p, p; q, q)$ be the coefficients after t steps of the walk starting at a particular point q (i.e. $g_0(p, p; q, q) = \delta_{p,q}$). Now, for any starting state ρ, let the initial coefficients be $\gamma_0(p, p)$. Then, by linearity, we can write the expected coefficients after t steps $\gamma_t(p, p) := \mathbb{E}\gamma_W(p, p)$ as

$$\gamma_t(p, p) = \sum_{q \neq 0} \gamma_0(q, q) g_t(p, p; q, q) \tag{4.3}$$

for $p \neq 0$.

We can now prove convergence rates for the expected coefficients $\gamma_t(p, p)$:

(i) For the 2-norm, we have from Lemma 2.1 that for $t \geq Cn \log 1/\epsilon$,

$$\sum_{p \neq 0} \left(g_t(p, p; q, q) - \frac{1}{4^n - 1} \right)^2 \leq \epsilon \tag{4.4}$$

for any q. Note that the normalisation for the $\gamma(p, p)$ terms with $p \neq 0$ has changed from Lemma 2.1 since we are neglecting the $\gamma(0, 0)$ term here. Now

$$\begin{aligned}
&\sum_{p \neq 0} \left(\gamma_t(p, p) - \frac{\sum_{q \neq 0} \gamma_0(q, q)}{4^n - 1} \right)^2 \\
&= \sum_{p \neq 0} \left(\sum_{q \neq 0} \gamma_0(q, q) \left(g_t(p, p; q, q) - \frac{1}{4^n - 1} \right) \right)^2 \\
&\leq \sum_{q \neq 0} \gamma_0(q, q)^2 \sum_{q' \neq 0} \sum_{p \neq 0} \left(g_t(p, p; q', q') - \frac{1}{4^n - 1} \right)^2 \\
&\leq (4^n - 1)\epsilon \sum_{q \neq 0} \gamma_0(q, q)^2 \\
&\leq 4^n \epsilon \sum_{q_1, q_2} \gamma_0(q_1, q_2)^2 \\
&= 4^n \epsilon \operatorname{tr} \rho^2 \\
&\leq 4^n \epsilon,
\end{aligned}$$

where the first inequality is the Cauchy-Schwarz inequality. Therefore for $t \geq Cn(n + \log 4^n/\epsilon)$, the 2-norm distance from stationarity for the $\gamma(p, p)$ terms is at most ϵ. Choose C' such that $C'n(n + \log 1/\epsilon) \geq Cn(n + \log 4^n/\epsilon)$ to obtain the result.

(ii) For the 1-norm, Lemma 2.1 says that for $t \geq Cn(n + \log 1/\epsilon)$,

$$\sum_{p \neq 0} \left| g_t(q; p, p) - \frac{1}{4^n - 1} \right| \leq \epsilon. \tag{4.5}$$

We can then proceed much as for the 2-norm case:

$$\begin{aligned}
&\sum_{p\neq 0}\left|\gamma_t(p,p)-\frac{\sum_{q\neq 0}\gamma_0(q,q)}{4^n-1}\right| \\
&=\sum_{p\neq 0}\left|\sum_{q\neq 0}\gamma_0(q,q)\left(g_t(p,p;q,q)-\frac{1}{4^n-1}\right)\right| \\
&\leq \sum_{q\neq 0}|\gamma_0(q,q)|\sum_{p\neq 0}\left|g_t(p,p;q,q)-\frac{1}{4^n-1}\right| \\
&\leq \epsilon\sum_{q\neq 0}|\gamma_0(q,q)| \\
&\leq 2^n\epsilon.
\end{aligned}$$

The last inequality follows from $|\sigma_q\otimes\sigma_q|=\sigma_0\otimes\sigma_0$. Therefore for $t\geq Cn(n+\log 2^n/\epsilon)$, the 1-norm distance from stationarity for the $\gamma(p,p)$ terms is at most ϵ.

We now proceed to prove Lemma 2.1. Firstly, we will consider the simple case of $k=1$ to prove this process forms a 1-design as this will help us to understand the more complicated case of $k=2$.

4.1. First moments convergence. Recall that $\rho=2^{-n/2}\sum_p\gamma(p)\sigma_p$ and we wish to evaluate the moments of the coefficients. So for the first moments to converge, we want to know $\mathbb{E}\gamma(p)$.

For $k=1$, the $U(4)$ random circuit uniformly randomises each pair that is chosen. More precisely, a pair of sites i,j are chosen at random and all the coefficients with $p_i\neq 0$ or $p_j\neq 0$ are set to zero. Thus we get an exact 1-design when all sites have been hit. For other gate sets, the terms do not decay to zero but decay by a factor depending on the gap of $\hat{G}$. Call the gap Δ; for $U(4)$ $\Delta=1$ and for others $0<\Delta\leq 1$ and Δ is independent of n. Therefore once each site has been hit m times the terms have decayed by a factor $(1-\Delta)^m$.

For a bound like the mixing time (see Sect. 4.3 for definition), we want to bound the quantity $\sum_{p\neq 0}|\mathbb{E}_W\gamma_W(p)|$, where $\gamma_W(p)$ is the Pauli coefficient after applying the random circuit W. We also want 2-norm bounds, so we bound $\sum_{p\neq 0}(\mathbb{E}_W\gamma_W(p))^2$ too. We will in fact find bounds on $\sum_{p\neq 0}\mathbb{E}_W|\gamma_W(p)|$ and $\sum_{p\neq 0}(\mathbb{E}_W|\gamma_W(p)|)^2$, which are stronger.

A standard problem in the theory of randomised algorithms is the 'coupon collector' problem. If a magazine comes with a free coupon, which is chosen uniformly randomly from n different types, how many magazines should you buy to have a high probability of getting all n coupons? It is not hard to show that $n\ln\frac{n}{\epsilon}$ samples (magazines) have at least a $1-\epsilon$ probability of including all n coupons. Using this, we expect all sites to be hit with probability at least $1-\epsilon$ after $\Theta(n\log\frac{n}{\epsilon})$ steps. This argument can be made precise in this context by bounding the non-identity coefficients. We find, as expected, that the sum is small after $O(n\log n)$ steps:

Lemma 4.1. *After* $O(n \log 1/\epsilon)$ *steps*

$$\sum_{p \neq 0} (\mathbb{E}_W |\gamma_W(p)|)^2 \leq \epsilon,$$

and after $O(n \log \frac{n}{\epsilon})$ *steps,*

$$\sum_{p \neq 0} \mathbb{E}_W |\gamma_W(p)| \leq \epsilon. \tag{4.6}$$

Proof. At each step, a pair of sites is chosen at random and any terms with non-identity coefficients for this pair decay by a factor $(1 - \Delta)$. For example, the term $\sigma_1 \otimes \sigma_0^{\otimes(n-1)}$ decays whenever the first site is chosen. Thus the probability of each term decaying depends on the number of zeroes. We start with the 1-norm bound. □

Suppose the circuit applied after t steps is W_t. Consider $\mathbb{E}_{W_t} |\gamma_{W_t}(p)|$ for any p with d non-zeroes. Since the state ρ is physical, $\operatorname{tr} \rho^2 \leq 1$, so $\sum_p \gamma_0^2(p) \leq 1$. Now, in each step, if any site is chosen where p is non-zero, this term decays by a factor $(1 - \Delta)$. This occurs with probability $1 - \frac{(d-n)(d-n-1)}{n(n-1)} \geq d/n$, the probability of choosing a pair where at least one site is non-zero. Therefore

$$\mathbb{E} |\gamma_{W_t}(p)| \leq ((1 - \Delta)d/n + (1 - d/n)) \, |\gamma_{W_{t-1}}(p)|,$$

where the expectation is over the circuit applied at step t. If we iterate this t times we find

$$\mathbb{E}_W |\gamma_W(p)| \leq \exp(-\Delta t d/n) |\gamma_0(p)|,$$

where the expectation here is over all random circuits for the t steps. We now sum over all p:

$$\sum_{p \neq 0} \mathbb{E}_W |\gamma_W(p)| \leq \sum_{d=1}^{n} \exp(-\Delta t d/n) \sum_{d(p)=d} |\gamma_0(p)|,$$

where $d(p)$ is the number of non-zeroes in p. For the 1-norm bound, we can simply bound $|\gamma_0(p)| \leq 1$ to give $\sum_{d(p)=d} |\gamma_0(p)| \leq \binom{n}{d} 3^d$ so

$$\sum_{p \neq 0} \mathbb{E}_W |\gamma_W(p)| \leq (1 + 3 \exp(-\Delta t/n))^n - 1,$$

where we have used the binomial theorem. Now let $t = \frac{n}{\Delta} \ln \frac{3n}{\epsilon}$. This gives

$$\sum_{p \neq 0} \mathbb{E}_W |\gamma_W(p)| \leq (1 + \epsilon/n)^n - 1 = O(\epsilon).$$

For the 2-norm bound,

$$\begin{aligned}\sum_{p\neq 0}(\mathbb{E}_W|\gamma_W(p)|)^2 &\leq \sum_{p\neq 0}\exp(-2\Delta t d/n)\gamma_0^2(p)\\ &= \sum_{d=1}^{n}\exp(-2\Delta t d/n)\sum_{d(p)=d}\gamma_0^2(p)\\ &\leq \sum_{d=1}^{n}\exp(-2\Delta t d/n)\\ &\leq \frac{\exp(-2\Delta t/n)}{1-\exp(-2\Delta t/n)},\end{aligned}$$

where we have used $\sum_p \gamma_0^2(p) \leq 1$. We find after $\frac{n}{2\Delta}\ln 1/\epsilon$ steps that

$$\sum_{p\neq 0}(\mathbb{E}_W|\gamma_W(p)|)^2 \leq \frac{\epsilon}{1-\epsilon}$$

4.2. Second moments convergence. Firstly, the $\sigma_{p_1}\otimes\sigma_{p_2}$ terms for $p_1 \neq p_2$ decay in a similar way to the non-identity terms in the 1-design analysis. In fact, the proof of Lemma 4.1 carries over almost identically to this case to give

Lemma 4.2. *After* $O(n\log 1/\epsilon)$ *steps*

$$\sum_{p_1\neq p_2}(\mathbb{E}_W|\gamma_W(p_1,p_2)|)^2 \leq \epsilon$$

and after $O(n(n+\log 1/\epsilon))$ *steps*

$$\sum_{p_1\neq p_2}\mathbb{E}_W|\gamma_W(p_1,p_2)| \leq \epsilon.$$

Proof. Instead of the number of zeroes governing the decay rate, we need to count the number of places where p_1 and p_2 differ. This gives

$$\mathbb{E}|\gamma_{W_t}(p_1,p_2)| \leq ((1-\Delta)d/n + (1-d/n))\,|\gamma_{W_{t-1}}(p_1,p_2)|,$$

where now d is the number of differing sites. There are $\binom{n}{d}12^d 4^{n-d}$ states that differ in d places so we find

$$\sum_{p_1\neq p_2}\mathbb{E}_W|\gamma_W(p_1,p_2)| \leq 4^n[(1+3\exp(-\Delta t/n))^n - 1].$$

Set $t = \frac{n}{\Delta}(n\ln 4 + \ln 1/\epsilon)$ to make this $O(\epsilon)$. The 2-norm bound follows in the same way as for Lemma 4.1. □

We now need to prove the $\gamma(p, p)$ terms converge quickly. We have seen above that the sum of the terms $\gamma(p, p)$ is conserved and, for the purposes of proving Lemma 2.1, we assume the sum is 1 and $\gamma(p, p) \geq 0$ for all p.

To illustrate the evolution, consider the simplest case when the gates are chosen from $U(4)$. We have evaluated $\hat{G}$ in Sect. 3.2 for $k = 2$ for this case. Translated into coefficients this yields the following update rule, where we have written it for the case when qubits 1 and 2 are chosen:

$$\gamma_{t+1}(r_1, r_2, r_3, \ldots, r_n, s_1, s_2, s_3, \ldots, s_n) = \begin{cases} 0 & (r_1, r_2) \neq (s_1, s_2) \\ \gamma_t(0, 0, r_3, \ldots, r_n, 0, 0, s_3, \ldots, s_n) & (r_1, r_2) = (s_1, s_2) = (0, 0) \\ \frac{1}{15} \sum_{\substack{r_1', r_2' \\ r_1' r_2' \neq 0}} \gamma_t(r_1', r_2', r_3, \ldots, r_n, r_1', r_2', s_3, \ldots, s_n) & (r_1, r_2) = (s_1, s_2) \neq (0, 0). \end{cases} \tag{4.7}$$

The key idea of Oliveira et al. [26] was to map the evolution of the $\gamma(p, p)$ coefficients to a Markov chain. We can apply this here to get, on state space $\{0, 1, 2, 3\}^n$, the evolution:

1. Choose a pair of sites uniformly at random.
2. If the state is 00 it remains 00.
3. Otherwise, choose the state uniformly at random from $\{0, 1, 2, 3\}^2 \backslash \{00\}$.

This is the correct evolution since, if the initial state is distributed according to $\gamma_t(q, q)$, the final state is distributed according to $\gamma_{t+1}(p, p)$.

The evolution for other gate sets will be similar, but the states will not be chosen uniformly randomly in the third step. However, the state 00 will remain 00 and the stationary distribution on the other 15 states is the same. We will find the convergence times for general gate sets and then consider the $U(4)$ gate set since we can perform a tight analysis for this case.

4.3. Markov chain analysis. Before finding the convergence rate for our problem, we will briefly introduce the basics of Markov chain mixing time analysis. All of these standard results can be found in Ref. [25] and references therein.

A process is Markov if the evolution only depends on the current state rather than the full state history. Therefore the evolution of the state can be thought of as a matrix, the *transition matrix*, acting on a vector which represents the current distribution. We will only be interested in discrete time processes so the state after t steps is given by the t^{th} power of the transition matrix acting on the initial distribution.

We say a Markov chain is *irreducible* if it is possible to get from one state to any other state in some number of steps. Further, a chain is *aperiodic* if it does not return to a state at regular intervals. If a chain is both irreducible and aperiodic then it is said to be *ergodic*. A well known result of Markov chain theory is that all ergodic chains converge to a unique stationary distribution. In matrix language this says that the transition matrix P has eigenvalue 1 with no multiplicity and all other eigenvalues have absolute value strictly less than 1. We will also need the notion of *reversibility*. A Markov chain is reversible if the time reversed chain has the same transition matrix, with respect to some distribution. This condition is also known as *detailed balance*:

$$\pi(x) P(x, y) = \pi(y) P(y, x). \tag{4.8}$$

It can be shown that a reversible ergodic Markov chain is only reversible with respect to the stationary distribution. So above $\pi(x)$ is the stationary distribution of P. An immediate consequence of this is that for a chain with uniform stationary distribution, it is reversible if and only if it is symmetric (i.e. $P(x, y) = P(y, x)$). Note also that reversible chains have real eigenvalues, since they are similar to the symmetric matrix $\sqrt{\frac{\pi(x)}{\pi(y)}}P(x, y)$.

With these definitions and concepts, we can now ask how quickly the Markov chain converges to the stationary distribution. This is normally defined in terms of the 1-norm mixing time. We use (half the) 1-norm distance to measure distances between distributions:

$$||s - t|| = \frac{1}{2}||s - t||_1 = \frac{1}{2}\sum_i |s_i - t_i|. \tag{4.9}$$

We assume all distributions are normalised so then $0 \leq ||s - t|| \leq 1$. We can now define the mixing time:

Definition 4.1. *Let π be the stationary distribution of P. Then if P is ergodic the mixing time τ is*

$$\tau(\epsilon) = \max_s \min_t \{t \geq 0 : \left|\left|P^t s - \pi\right|\right| \leq \epsilon\}. \tag{4.10}$$

We will also use the (weaker) 2-norm mixing time (note this is not the same as τ_2 in Ref. [25]):

Definition 4.2. *Let π be the stationary distribution of P. Then if P is ergodic the 2-norm mixing time τ_2 is*

$$\tau_2(\epsilon) = \max_s \min_t \{t \geq 0 : \left|\left|P^t s - \pi\right|\right|_2 \leq \epsilon\}. \tag{4.11}$$

Unless otherwise stated, when we say mixing time we are referring to the 1-norm mixing time.

There are many techniques for bounding the mixing time, including finding the second largest eigenvalue of P. This gives a good measure of the mixing time because components parallel to the second largest eigenvector decay the slowest. We have (for reversible ergodic chains)

Theorem 4.1 (see Ref. [25], Corollary 1.15).

$$\tau(\epsilon) \leq \frac{1}{\Delta}\ln\frac{1}{\pi_*\epsilon},$$

where $\pi_* = \min \pi(x)$ and $\Delta = \min(1 - \lambda_2, 1 + \lambda_{min})$, where λ_2 is the second largest eigenvalue and λ_{min} is the smallest. Δ is known as the **gap**.

If the chain is irreversible, it may not even have real eigenvalues. However, we can bound the mixing time in terms of the eigenvalues of the reversible matrix PP^*, where $P^*(x, y) = \frac{\pi(y)}{\pi(x)}P(y, x)$. In this case we have ([25], Corollary 1.14)

$$\tau(\epsilon) \leq \frac{2}{\Delta_{PP^*}}\ln\frac{1}{\pi_*\epsilon}, \tag{4.12}$$

where now Δ_{PP^*} is the gap of the chain PP^*. Note that for a reversible chain $P = P^*$ and $\Delta_{PP^*} \approx 2\Delta$, so the bounds are approximately the same.

This can also be converted into a 2-norm mixing time bound:

$$\tau_2(\epsilon) \leq \frac{2}{\Delta_{PP^*}} \ln 1/\epsilon. \tag{4.13}$$

To bound the gap, we will use the comparison theorem in Theorem 4.2 below. In this theorem, we are thinking of the Markov chain as a directed graph where the vertices are the states and there are edges for allowed transitions (i.e. transitions with non-zero probability). For irreducible chains, it is possible to make a path from any vertex to any other; we call the path length the number of transitions in such a path (which will in general depend on the choice of path).

Theorem 4.2 (see Ref. [25], Theorem 2.14). *Let P and $\hat{P}$ be two Markov chains on the same state space Ω with the same stationary distribution π. Then, for every $x \neq y \in \Omega$ with $\hat{P}(x, y) > 0$ define a directed path γ_{xy} from x to y along edges in P and let its length be $|\gamma_{xy}|$. Let Γ be the set of all such paths. Then*

$$\Delta \geq \hat{\Delta}/A$$

for the gaps Δ and $\hat{\Delta}$ where

$$A = A(\Gamma) = \max_{a \neq b, P(a,b) \neq 0} \frac{1}{\pi(a)P(a, b)} \sum_{x \neq y : (a,b) \in \gamma_{xy}} \pi(x)\hat{P}(x, y)|\gamma_{xy}|.$$

For example, when comparing 1-dimensional random walks there is no choice in the paths; they must pass through every point between x and y. Further, the walk can only progress one step at a time so (without loss of generality, for reversible chains) let $b = a + 1$ to give

$$\begin{aligned} A &= \max_a \frac{1}{\pi(a)P(a, a+1)} \sum_{x \leq a} \sum_{y \geq a+1} \pi(x)\hat{P}(x, y)(y - x) \\ &= \max_a \frac{\hat{P}(a, a+1)}{P(a, a+1)}. \end{aligned} \tag{4.14}$$

A generalisation of the comparison theorem involves constructing flows, which are weighted sets of paths between states. This can give a tighter bound since bottlenecks are averaged over. This gives a modified comparison theorem:

Theorem 4.3 ([12], Theorem 2.3). *Let P and $\hat{P}$ be two Markov chains on the same state space Ω with the same stationary distribution π. Then, for every $x \neq y \in \Omega$ with $\hat{P}(x, y) > 0$, construct a set of directed paths $\mathcal{P}_{xy}$ from x to y along edges in P. We define the flow function f which maps each path $\gamma_{xy} \in \mathcal{P}_{xy}$ to a real number in the interval* $[0, 1]$ *such that*

$$\sum_{\gamma_{xy} \in \mathcal{P}_{xy}} f(\gamma_{xy}) = \hat{P}(x, y).$$

Again, let the length of each path be $|\gamma_{xy}|$. Then

$$\Delta \geq \hat{\Delta}/A$$

for the gaps Δ and $\hat{\Delta}$ where

$$A = A(f) = \max_{a \neq b, P(a,b) \neq 0} \frac{1}{\pi(a)P(a,b)} \sum_{x \neq y, \gamma_{xy} \in \mathcal{P}_{xy}: (a,b) \in \gamma_{xy}} \pi(x) f(\gamma_{xy}) |\gamma_{xy}|. \quad (4.15)$$

Note that we recover the comparison theorem when there is just one path between each x and y.

4.3.1. log-Sobolev constant We will need tighter, but more complicated, mixing time results to prove the tight result for the $U(4)$ case. We use the log-Sobolev constant:

Definition 4.3. *The log-Sobolev constant ρ of a chain with transition matrix P and stationary distribution π is*

$$\rho = \min_f \frac{\sum_{x \neq y} (f(x) - f(y))^2 P(x,y) \pi(y)}{\sum_x \pi(x) f(x)^2 \log \frac{f(x)^2}{\sum_y \pi(y) f(y)^2}}.$$

The mixing time result is:

Lemma 4.3 (see Ref. [13], Theorem 3.7'). *The mixing time of a finite, reversible, irreducible Markov chain is*

$$\tau(\epsilon) = O\left(\frac{1}{\rho} \log\log \frac{1}{\pi_*} + \frac{1}{\Delta} \log \frac{d}{\epsilon}\right), \quad (4.16)$$

where ρ is the Sobolev constant, π_* is the smallest value of the stationary distribution, Δ is the gap and d is the size of the state space.

Further, the comparison theorem (Theorem 4.2) works just the same to give

$$\rho \geq \hat{\rho}/A.$$

We will need one more result, due to Diaconis and Saloff-Coste:

Lemma 4.4 ([13], Lemma 3.2). *Let P_i, $i = 1, \ldots, d$, be Markov chains with gaps Δ_i and Sobolev constants ρ_i. Now construct the product chain P. This chain has state space equal to the product of the spaces for the chains P_i and at each step one of the chains is chosen at random and run for one step. Then P has spectral gap given by:*

$$\Delta = \frac{1}{d} \min_i \Delta_i$$

and Sobolev constant:

$$\rho = \frac{1}{d} \min_i \rho_i.$$

4.4. Convergence proof. We now prove the Markov chain convergence results to show that the $\gamma(p, p)$ terms converge quickly. We have already shown that the $\gamma(p_1, p_2)$ terms with $p_1 \neq p_2$ converge quickly and that there is no mixing between these terms and the $\gamma(p, p)$ terms. Therefore, in this section, we remove such terms from $\hat{G}$.

We want to prove the Markov chain with transition matrix (Eq. 4.2)

$$P = \frac{1}{n(n-1)} \sum_{i \neq j} \hat{G}^{(ij)}$$

converges quickly. Firstly, we know from Sect. 3.3 that P has two eigenvectors with eigenvalue 1. The first is the identity state $(\sigma_0 \otimes \sigma_0)$ and the second is the uniform sum of all non-identity terms $(\frac{1}{4^n-1} \sum_{p \neq 0} \sigma_p \otimes \sigma_p)$. From now on, we remove the identity state. This makes the chain irreducible. Since we know it converges, it must be aperiodic also so the chain is ergodic and all other eigenvalues are strictly between 1 and -1.

We show here that the gap of this chain, up to constants, does not depend on the choice of 2-copy gapped gate set. In the second half of the paper we find a tight bound on the gap for the $U(4)$ case which consequently gives a tight bound on the gap for all universal sets.

Since the stationary distribution is uniform, the chain is reversible if and only if P is a symmetric matrix. A sufficient condition for P to be symmetric is for $\hat{G}^{(ij)}$ to be symmetric. We saw in Theorem 3.1 that for the $U(4)$ gate set case $\hat{G}^{(ij)}$ is symmetric. In fact, the proof works identically to show that $\hat{G}^{(ij)}$ is symmetric for any gate set, provided the set is invariant under Hermitian conjugation. However, 2-copy gapped gate sets do not necessarily have this property so the Markov chain is not necessarily reversible. We will find equal bounds (up to constants) for the gaps of both P (if $\hat{G}$ is symmetric) and PP^* (if $\hat{G}$ is not symmetric) below:

Theorem 4.4. *Let μ be any 2-copy gapped distribution of gates. If μ is invariant under Hermitian conjugation then let Δ_P be the eigenvalue gap of the resulting Markov chain matrix P. Then*

$$\Delta_P = \Omega(\Delta_{U(4)}), \tag{4.17}$$

where $\Delta_{U(4)}$ is the eigenvalue gap of the $U(4)$ chain. If μ is not invariant under Hermitian conjugation, then let Δ_{PP^*} be the eigenvalue gap of the resulting Markov chain matrix PP^*. Then

$$\Delta_{PP^*} = \Omega(\Delta_{U(4)}). \tag{4.18}$$

Proof. We will use the comparison method with flows (Theorem 4.3). Firstly consider the case where μ is closed under Hermitian conjugation, i.e. $\hat{G}$ is symmetric.

We will compare P to the $U(4)$ chain, which we call $P_{U(4)}$. Recall that this chain chooses a pair at random and does nothing if the pair is 00 and chooses a random state from $\{0, 1, 2, 3\}^2 \backslash \{00\}$ otherwise.

To apply Theorem 4.3, we need to construct the flows between transitions in $P_{U(4)}$. We will choose paths such that only one pair is modified throughout. For example (with $n = 4$), the transition $1000 \to 2000$ is allowed in $P_{U(4)}$. To construct a path in P, we need to find allowed transitions between these two paths in P. $\hat{G}$ may not include the transition $10 \to 20$ directly, however, $\hat{G}$ is irreducible on this subspace of just two pairs. This means that a path exists and can be of maximum length 14 if it has to cycle through

all intermediate states (in fact, since $\hat{G}$ is symmetric the maximum path length is 8; all that is important here is that it is constant). For example, the transitions $10 \to 11 \to 20$ might be allowed. Then we could choose the full path to be $1000 \to 1100 \to 2000$. In this case we have chosen the path to involve transition pairing sites 1 and 2. However, we could equally well have chosen any pairing; we could pair the first site with any of the others. We can choose 3 paths in this way. For this example, the flow we want to choose will be all 3 of these paths equally weighted. We now use this idea to construct flows between all transitions in $P_{U(4)}$ to prove the result.

Let $x \neq y \in \Omega$ and let $d(x, y)$ be the Hamming distance between the states ($d(x, y)$ gives the number of places at which x and y differ). There are two cases where $P_{U(4)}(x, y) \neq 0$:

1. $d(x, y) = 2$. Here we must choose a unique pairing, specified by the two sites that differ. Make all transitions in P using this pair giving just one path.
2. $d(x, y) = 1$. For this case, choose all possible pairings of the changing site that give allowed transitions in $P_{U(4)}$. For each pairing, construct a path in P modifying only this pair. If the differing site is initially non-zero then there are $n - 1$ such pairings; if the differing site is initially zero then there are $n - z(x)$ pairings where $z(x)$ is the number of zeroes in the state x.

All the above paths are of constant length since we have to (at most) cycle through all states of a pair. We must now choose the weighting $f(\gamma_{xy})$ for each path such that

$$\sum_{\mathcal{P}_{xy}} f(\gamma_{xy}) = P_{U(4)}(x, y), \tag{4.19}$$

where $\mathcal{P}_{xy}$ is the set of all paths from x to y constructed above. We choose the weighting of each path to be uniform. We just need to calculate the number of paths in $\mathcal{P}_{xy}$ to find f:

1. $d(x, y) = 2$. There is just one path so $f(\gamma_{xy}) = P_{U(4)}(x, y) = \Theta(1/n^2)$.
2. $d(x, y) = 1$. If the differing site is initially non-zero then $P_{U(4)}(x, y) = \Theta(1/n)$ and there are $n - 1$ paths so $f(\gamma_{xy}) = \frac{P_{U(4)}(x,y)}{n-1} = \Theta(1/n^2)$. If the differing site is initially zero then $P_{U(4)}(x, y) = \Theta\left(\frac{n-z(x)}{n^2}\right)$ and there are $n - z(x)$ paths so $f(\gamma_{xy}) = \frac{P_{U(4)}(x,y)}{n-z(x)} = \Theta(1/n^2)$.

So for all paths, $f = \Theta(1/n^2)$. We now just need to know how many times each edge (a, b) in P is used to calculate A:

$$A = \max_{a \neq b, P(a,b) \neq 0} A(a, b), \tag{4.20}$$

where

$$A(a, b) = \frac{1}{P(a, b)} \sum_{x \neq y, \gamma_{xy} \in \mathcal{P}_{xy} : (a,b) \in \gamma_{xy}} f(\gamma_{xy}). \tag{4.21}$$

We have cancelled the factors of $\pi(x)$ because the stationary distribution is uniform. We have also ignored the lengths of the paths since they are all constant.

To evaluate $A(a, b)$, we need to know how many paths pass through each edge (a, b). We again consider the two possibilities separately:

1. $d(a,b) = 2$. Suppose a and b differ at sites i and j. Firstly, we need to count how many transitions from x to y in $P_{U(4)}$ could use this edge, and then how many paths for each transition actually use the edge.
   To find which x and y could use the edge, note that x and y must differ at sites i, j or both. Furthermore, the values at the sites other than i and j must be the same as for a (and therefore b). There is a constant number of x, y pairs that satisfy this condition. Now, for each x, y pair satisfying this, paths that use this edge must use the pairing i, j for all transitions. Since in the paths we have chosen above there is a unique path from x to y for each pairing, there is at most one path for each x, y pair that uses edge a, b.
   For $d(a,b) = 2$, $P(a,b) = \Theta(1/n^2)$ so $A(a,b)$ is a constant for this case.
2. $d(a,b) = 1$. Let there be r pairings that give allowed transitions in P between a and b. As above, each pairing gives a constant number of paths. So the numerator is $\Theta(r/n^2)$. Further, $P(a,b) = \Theta(r/n^2)$. So again $A(a,b)$ is constant.

Combining, A is a constant so the result is proven for the case $\hat{G}$ is symmetric.

We now turn to the irreversible case. We now need to bound the gap of $PP^* = PP^T$. This chain selects two (possibly overlapping) pairs at random and applies $\hat{G}$ to one of them and $\hat{G}^T$ to the other. We can use the above exactly by choosing $\hat{G}$ to perform the transitions above and $\hat{G}^T$ to just loop the states back to themselves. By aperiodicity (the greatest common divisor of loop lengths is 1), we can always find constant length paths that do this.

Now we need to know the gap of the $U(4)$ chain. We can, by a simple application of the comparison theorem, show it is $\Omega(1/n^2)$. However, in the second half of this paper we show it is $\Theta(1/n)$. This gives us (using Theorem 4.1):

Corollary 4.1. *The Markov chain P has mixing time $O(n(n + \log 1/\epsilon))$ and 2-norm mixing time $O(n \log 1/\epsilon)$.*

We conjecture that the mixing time (as well as Lemma 4.2) can be tightened to $\Theta(n \log \frac{n}{\epsilon})$, which is asymptotically the same as for the $U(4)$ case:

Conjecture 4.1. The second moments for the case of general 2-copy gapped distributions have 1-norm mixing time $\Theta(n \log \frac{n}{\epsilon})$.

It seems likely that an extension of our techniques in Sect. 5 could be used to prove this.

Combining the convergence results we have proved our general result Lemma 2.1:

Proof (of Lemma 2.1). Combining Corollary 4.1 (for the $\gamma(p, p)$ terms) and Lemma 4.2 (for the $\gamma(p_1, p_2)$, $p_1 \neq p_2$ terms) proves the result.

We have now shown that the first and second moments of random circuits converge quickly. For the remainder of the paper we prove the tight bound for the gap and mixing time of the $U(4)$ case and show how mixing time bounds relate to the closeness of the 2-design to an exact design. Only for the $U(4)$ case is the matrix $\hat{G}$ a projector so in this sense the $U(4)$ random circuit is the most fundamental. While we expect the above mixing time bound is not tight, we can prove a tight mixing time result for the $U(4)$ case. However, using our definition of an approximate k-design, the gap rather than the mixing time governs the degree of approximation.

5. Tight Analysis for the $U(4)$ Case

We have already found tight bounds for the first moments in Lemma 4.1: just set $\Delta = 1$.

5.1. Second moments convergence. We need to prove a result analogous to Lemma 4.2 for the terms $\sigma_{p_1} \otimes \sigma_{p_2}$, where $p_1 \neq p_2$. We already have a tight bound for the 2-norm decay, by setting $\Delta = 1$ into Lemma 4.2. We tighten the 1-norm bound:

Lemma 5.1. *After* $O(n \log \frac{n}{\epsilon})$ *steps*

$$\sum_{p_1 \neq p_2} \mathbb{E}_W |\gamma_W(p_1, p_2)| \leq \epsilon. \tag{5.1}$$

Proof. We will split the random circuits up into classes depending on how many qubits have been hit. Let H be the random variable giving the number of different qubits that have been hit. We can work out the distribution of H and bound the sum of $|\gamma_W(p_1, p_2)|$ for each outcome.

Firstly we have, after t steps,

$$\mathbb{P}(H \leq h) \leq \binom{n}{h} \left(\frac{h(h-1)}{n(n-1)}\right)^t \leq \binom{n}{h} (h/n)^t.$$

Now, for each qubit hit, each coefficient which has p_1 and p_2 differing in this place is set to zero. So after h have been hit, there are only (at most) $16^{(n-h)}$ terms in the sum in Eq. 5.1. As before, the state is a physical state, $\operatorname{tr} \rho^2 \leq 1$ so $\sum_{p_1 p_2} \gamma^2(p_1, p_2) \leq 1$ so $\sum_{p_1 p_2} |\gamma(p_1, p_2)| \leq \sqrt{N}$ if there are at most N non-zero terms in the sum. Therefore we have, after t steps,

$$\begin{aligned}
\sum_{p_1 \neq p_2} \mathbb{E}_W |\gamma_W(p_1, p_2)| &\leq \sum_{h=1}^{n-1} \mathbb{P}(H = h) 16^{(n-h)/2} \\
&\leq \sum_{h=1}^{n-1} \mathbb{P}(H \leq h) 4^{(n-h)} \\
&\leq \sum_{h=1}^{n-1} \binom{n}{h} (h/n)^t 4^{(n-h)} \\
&= \sum_{h=1}^{n-1} \binom{n}{h} (1 - h/n)^t 4^h \qquad h \to n - h \\
&\leq \sum_{h=1}^{n-1} \binom{n}{h} \exp(-ht/n) 4^h.
\end{aligned}$$

Now, let $t = n \ln \frac{n}{\epsilon}$:

$$\sum_{p_1 \neq p_2} \mathbb{E}_W |\gamma_W(p_1, p_2)| \leq \sum_{h=1}^{n-1} \binom{n}{h} \left(\frac{4\epsilon}{n}\right)^h = \left(1 + \frac{4\epsilon}{n}\right)^n - 1 - \left(\frac{4\epsilon}{n}\right)^n = O(\epsilon),$$

where the last line follows from the binomial theorem. □

This, combined with the mixing time result we prove below, completes the proof that the second moments of the random circuit converge in time $O(n \log \frac{n}{\epsilon})$.

5.2. Markov chain of coefficients. The Markov chain acting on the coefficients is reducible because the state $\{0\}^n$ is isolated. However, if we remove it then the chain becomes irreducible. The presence of self loops implies aperiodicity, therefore the chain is ergodic. We have already seen that the chain converges to the Haar uniform distribution (in Sect. 1.1), therefore the stationary state is the uniform state $\pi(x) = 1/(4^n - 1)$. Further, since the chain is symmetric and has uniform stationary distribution, the chain satisfies detailed balance (Eq. 4.8) so is reversible. We now turn to obtaining bounds on the mixing time of this chain.

We want to show that the full chain converges to stationarity in time $\Theta(n \log \frac{n}{\epsilon})$. This implies (see later) that the gap is $\Theta(1/n)$. To prove this, we will construct another chain called the zero chain. This is the chain that counts the number of zeroes in the state. Since it is the zeroes that slow down the mixing, this chain will accurately describe the mixing time of the full chain.

Lemma 5.2. *The zero chain has transition matrix P on state space (we count non-zero positions)* $\Omega = \{1, 2, \ldots, n\}$.

$$P(x, y) = \begin{cases} 1 - \frac{2x(3n-2x-1)}{5n(n-1)} & y = x \\ \frac{2x(x-1)}{5n(n-1)} & y = x - 1 \\ \frac{6x(n-x)}{5n(n-1)} & y = x + 1 \\ 0 & \text{otherwise} \end{cases} \tag{5.2}$$

for $1 \leq x, y \leq n$.

Proof. Suppose there are $n - x$ zeroes (so there are x non-zeroes). Then the only way the number of zeroes can decrease (i.e. for x to increase) is if a non-zero item is paired with a zero item and one of the 9 (out of 15) new states is chosen with no zeroes. The probability of choosing such a pair is $\frac{2x(n-x)}{n(n-1)}$ so the overall probability is $\frac{9}{15}\frac{2x(n-x)}{n(n-1)}$.

The number of zeroes can increase only if a pair of non-zero items is chosen and one of the 6 states is chosen with one zero. The probability of this occurring is $\frac{6}{15}\frac{x(x-1)}{n(n-1)}$.

The probability of the number of zeroes remaining unchanged is simply calculated by requiring the probabilities to sum to 1.

We see that the zero chain is a one-dimensional random walk on the line. It is a lazy random walk because the probability of moving at each step is < 1. However, as the number of zeroes decreases, the probability of moving increases monotonically:

$$1 - P(x, x) = \frac{2x(3n - 2x - 1)}{5n(n - 1)} \geq 2x/5n < 1. \tag{5.3}$$

□

Lemma 5.3. *The stationary distribution of the zero chain is*

$$\pi_0(x) = \frac{3^x \binom{n}{x}}{4^n - 1}. \tag{5.4}$$

Proof. This can be proven by multiplying the transition matrix in Lemma 5.2 by the state Eq. 5.4. Alternatively, it can be proven by counting the number of states with $n - x$ zeroes. There are $\binom{n}{x}$ ways of choosing which sites to make non-zero and each non-zero site can be one of three possibilities: 1, 2 or 3. The total number of states is $4^n - 1$, which gives the result. □

Below we will prove the following theorem:

Theorem 5.1. *The zero chain mixes in time* $\Theta(n \log \frac{n}{\epsilon})$.

The 2-norm mixing time follows easily:

Theorem 5.2. *The zero chain has 2-norm mixing time* $O(n \log 1/\epsilon)$.

Proof. We use a lower bound on the 1-norm mixing time to show that the gap of the zero chain is $\Omega(1/n)$ and then use the 2-norm mixing bound Eq. 4.13. In [25], Theorem 4.9, they prove the lower bound:

$$\tau_1(\epsilon) \geq \frac{1 - \Delta}{\Delta} \ln \frac{1}{2\epsilon}, \tag{5.5}$$

where Δ is the eigenvalue gap. In Theorem 5.1, we showed $\tau_1(\epsilon) \leq Cn \ln \frac{n}{\epsilon}$ for some constant C. Combining,

$$\frac{1 - \Delta}{\Delta} \ln \frac{1}{2\epsilon} \leq Cn \ln \frac{n}{\epsilon} \tag{5.6}$$

for all $\epsilon > 0$. Divide by $\ln 1/\epsilon$ and take the limit $\epsilon \to 0$ to find

$$\frac{1 - \Delta}{\Delta} \leq Cn \tag{5.7}$$

which implies the gap is $\Omega(1/n)$. The 2-norm bound now follows from Eq. 4.13. □

Before proving Theorem 5.1, we will show how the mixing time of the full chain follows from this.

Corollary 5.1. *The full chain mixes in time* $\Theta(n \log \frac{n}{\epsilon})$.

Proof. Once the zero chain has approximately mixed, the distribution of zeroes is almost correct. We need to prove that the distribution of non-zeroes is correct after $O(n \log \frac{n}{\epsilon})$ steps too.

Once each site of the full chain has been hit, meaning it is chosen and paired with another site so not both equal zero, the chain has mixed. This is because, after each site has been hit, the probability distribution over the states is uniform. When the zero chain has approximately mixed, a constant fraction of sites are zero so the probability of hitting a site at each step is $\Theta(1/n)$. By the coupon collector argument, each site will have been hit with probability at least $1 - \epsilon$ in time $O(n \log \frac{n}{\epsilon})$. Once the zero chain has mixed to ϵ', we can run the full chain this extra number of steps to ensure each site has

been hit with high probability. Since the mixing of the zero chain only increases with time, the distance to stationarity of the full chain is now $1-\epsilon-\epsilon'$. We make this formal below.

After $t_0 = O(n \log \frac{n}{\epsilon'})$ steps, the number of zeroes is ϵ'-close to the stationary distribution π_0 by Theorem 5.1 and only gets closer with more steps since the distance to stationarity decreases monotonically. The stationary distribution Eq. 5.4 is approximately a Gaussian peaked at $3n/4$ with $O(n)$ variance. This means that, with high probability, the number of non-zeroes is close to $3n/4$. We will in fact only need that there is at least a constant fraction of non-zeroes; with probability at least $1-\epsilon'-\exp(-\Omega(n))$ there will be at least $n/2$.

To prove the mixing time, we run the chain for time t_0 so the zero chain mixes to ϵ'. Then run for t_1 additional steps. Let $H_{i,t}$ be the event that site i is hit at step t. Let $H_i = \cup_{t=t_0+1}^{t_0+t_1} H_{i,t}$ and $H = \cap_{i=1}^{n} H_i$. We want to show $\mathbb{P}(H)$ is close to 1, or, in other words, that all sites are hit with high probability. Further let X_t be the random variable giving the number of non-zeroes at step t.

If at step $t-1$ site i is non-zero then the event $H_{i,t}$ occurs if the qubit is chosen, which occurs with probability $2/n$. If, however, it was zero then it must be paired with a non-zero thing for $H_{i,t}$ to hold. Conditioned on any history with $X_{t-1} \geq n/2$, this probability is $\geq 1/n$. In particular, we can condition on not having previously hit i and the bound does not change. Combining we have

$$\mathbb{P}\left(H_{i,t}^c \middle| \left[X_{t-1} \geq n/2\right] \bigcap \left(\bigcap_{t'=t_0+1}^{t-1} H_{i,t'}^c\right)\right) \leq 1 - 1/n.$$

Then, after t_1 extra steps,

$$\mathbb{P}\left(H_i^c \middle| \bigcap_{t=t_0}^{t_0+t_1-1} [X_t \geq n/2]\right) \leq (1-1/n)^{t_1},$$

which, using the union bound, gives

$$\mathbb{P}\left(H^c \middle| \bigcap_{t=t_0}^{t_0+t_1-1} [X_t \geq n/2]\right) \leq n(1-1/n)^{t_1}.$$

Now, since the zero chain has mixed to ϵ',

$$\mathbb{P}\left(\bigcap_{t=t_0}^{t_0+t_1-1} [X_t \geq n/2]\right) \leq t_1 \sum_{x=n/2}^{n-1} \pi_0(x) + \epsilon' \leq t_1 \exp(-O(n)) + \epsilon',$$

so

$$\mathbb{P}(H^c) \leq n(1-1/n)^{t_1} + t_1 \exp(-O(n)) + \epsilon'.$$

Now, choose $t_1 = n \ln \frac{2n}{\epsilon}$ so that $\mathbb{P}(H^c) \leq \delta$, where $\delta = \epsilon + t_1 \exp(-O(n))$. Choose $\epsilon = 1/n$ so that δ is $1/\operatorname{poly}(n)$. Now, using the bound on $\mathbb{P}(H^c)$, we can write the state v after $t_1 = O(n \log n)$ steps as

$$v = (1-\delta)\pi + \delta\pi',$$

where π is the stationary distribution and π' is any other distribution. Using this,

$$||v - \pi|| \leq \delta.$$

We now apply Lemma A.14 to show that after $O(n \log \frac{n}{\epsilon})$ steps the distance to stationarity of the full chain is ϵ. □

5.3. Proof of Theorem 5.1. We will now proceed to prove Theorem 5.1. We present an outline of the proof here; the details are in Sect. A.2.

Firstly, note that by the coupon collector argument, the lower bound on the time is $\Omega(n \log n)$. We need to prove an upper bound equal to this. Intuition says that the mixing time should take time $O(n \log n)$ because the walk has to move a distance $\Theta(n)$ and the waiting time at each step is proportional to $n, n/2, n/3, \ldots$ which sums to $O(n \log n)$, provided each site is not hit too often. We will show that this intuition is correct using the Chernoff bound and log-Sobolev (see later) arguments.

We will first work out concentration results of the position after some number of *accelerated* steps. The zero chain has some probability of staying still at each step. The accelerated chain is the zero chain conditioned on moving at each step. We define the accelerated chain by its transition matrix:

Definition 5.1. *The transition matrix for the accelerated chain is*

$$P_a(x, y) = \begin{cases} 0 & y = x \\ \frac{x-1}{3n-2x-1} & y = x - 1 \\ \frac{3(n-x)}{3n-2x-1} & y = x + 1 \\ 0 & \text{otherwise} \end{cases}. \tag{5.8}$$

We use the accelerated chain in the proof to firstly prove the accelerated chain mixes quickly, then to bound the waiting time at each step to obtain a mixing time bound for the zero chain.

To prove the mixing time bound, we will split the walk up into three phases. We will split the state space into three (slightly overlapping) parts and the phase can begin at any point within that space. So each phase has a state space $\Omega_i \subset [1, n]$, an entry space $E_i \subset \Omega_i$ and an exit condition T_i. We say that a phase completes successfully if the exit condition is satisfied in time $O(n \log n)$ for an initial state within the entry space. When the exit condition is satisfied, the walk moves onto the next phase.

The phases are:

1. $\Omega_1 = [1, n^\delta]$ for some constant δ with $0 < \delta < 1/2$. $E_1 = \Omega_1$ (i.e. it can start anywhere) and T_1 is satisfied when the walk reaches n^δ. For this part, the probability of moving backwards (gaining zeroes) is $O(n^{\delta-1})$ so the walk progresses forwards at each step with high probability. This is proven in Lemma A.8. We show that the waiting time is $O(n \log n)$ in Lemma A.9.
2. $\Omega_2 = [n^\delta/2, \theta n]$ for some constant θ with $0 < \theta < 3/4$. $E_2 = [n^\delta, \theta n]$ and T_2 is satisfied when the walk reaches θn. Here the walk can move both ways with constant probability but there is a $\Omega(1)$ forward bias. Here we use a monotonicity argument: the probability of moving forward at each step is

$$\begin{aligned} p(x) &= \frac{3(n-x)}{3n-2x-1} \\ &\geq \frac{3(n-x)}{3n-2x} \\ &\geq \frac{3(1-\theta)}{3-2\theta}. \end{aligned}$$

If we model this random walk as a walk with constant bias equal to $\frac{3(1-\theta)}{3-2\theta}$ we will find an upper bound on the mixing time since mixing time increases monotonically with decreasing bias. Further, the waiting time at $x = a$ stochastically dominates the waiting time at $x = b$ for $b \geq a$. The true bias decreases with position so the walk with constant bias spends more time at the early steps. Thus the position of this simplified walk is stochastically dominated by the position of the real walk while the waiting time stochastically dominates the waiting time of the real walk.

3. $\Omega_3 = [\frac{\theta}{2}n, n]$ and $E_3 = [\theta n, n]$. T_3 is satisfied when this restricted part of the chain has mixed to distance ϵ. Here the bias decreases to zero as the walk approaches $3n/4$ but the moving probability is a constant. We show that this walk mixes quickly by bounding the log-Sobolev constant of the chain.

Showing these three phases complete successfully will give a mixing time bound for the whole chain.

We now prove in the Appendix that the phases complete successfully with probability at least $1 - 1/\operatorname{poly}(n)$:

Lemma 5.4.

$$\mathbb{P}(\text{Phase 1 completes successfully}) \geq 1 - n^{2\delta-1} - 2n^{-\delta}.$$

Lemma 5.5.

$$\begin{aligned} \mathbb{P}(\text{Phase 2 completes successfully}) \geq 1 &- \exp\left(-\frac{2}{3}\mu\theta n\right) - \left(\frac{4}{\theta n}\right)^{\frac{3}{2\mu}} \\ &- \frac{2\exp\left(\frac{-\mu n^{\delta}}{4}\right)}{1-\exp(-\mu/2)} - (q/p)^{n^{\delta}/2}, \end{aligned}$$

where $\mu = \frac{6(1-\theta)}{3-2\theta} - 1$.

Lemma 5.6.

$$\mathbb{P}(\text{Phase 3 completes successfully}) \geq 1 - \left(\frac{\theta}{3(2-\theta)}\right)^{\theta n/2}.$$

We can now finally combine to prove our result:

Proof (of Theorem 5.1). The stationary distribution has exponentially small weight in the tail with lots of zeroes. We show that, provided the number of zeroes is within phase 3, the walk mixes in time $O(n \log \frac{n}{\epsilon})$. We also show that if the number of zeroes is initially within phase 1 or 2, after $O(n \log n)$ steps the walk is in phase 3 with high probability. We can work out the distance to the stationary distribution as follows.

Let p_f be the probability of failure. This is the sum of the error probabilities in Lemmas 5.4, 5.5 and 5.6. The key point is that $p_f = 1/\operatorname{poly}(n)$. Then after $O(n \log \frac{n}{\epsilon})$

steps (the sum of the number of steps in the 3 phases), the state is equal to $(1-p_f)v_3 + p_f v'$, where v_3 is the state in the phase 3 space and v' is any other distribution, which occurs if any one of the phases fails. Since the distance to stationarity in phase 3 is ϵ, $||v_3 - \pi_3|| \leq \epsilon$, where π_3 is the stationary distribution on the state space of phase 3. In Lemma A.12 we show that $\pi_3(x) = \pi(x)/(1-w)$, where $w = \sum_{x=1}^{\theta n/2-1} \pi(x)$. Since $\pi(x)$ is exponentially small in this range, w is exponentially small in n. Now use the triangle inequality to find

$$||v_3 - \pi|| \leq ||v_3 - \pi_3|| + ||\pi_3 - \pi||. \tag{5.9}$$

Since the chain in phase 3 has mixed to ϵ, the first term is $\leq \epsilon$. We can evaluate $||\pi_3 - \pi||$:

$$\begin{aligned} ||\pi_3 - \pi|| &= \frac{1}{2}\sum_{x=1}^{n} ||\pi_3(x) - \pi(x)|| \\ &= \frac{1}{2}\left(\sum_{x=1}^{\theta n/2-1} \pi(x) + \sum_{x=\theta n/2}^{n} (\pi(x)/(1-w) - \pi(x))\right) \\ &= \frac{1}{2}(w + 1 - (1-w)) = w. \end{aligned}$$

So now,

$$\begin{aligned} ||(1-p_f)v_3 + p_f v' - \pi|| &= ||(1-p_f)(v_3 - \pi) + p_f(v' - \pi)|| \\ &\leq (1-p_f)||v_3 - \pi|| + p_f||v' - \pi|| \\ &\leq (1-p_f)(\epsilon + w) + p_f \\ &\leq \delta, \end{aligned}$$

where $\delta = \epsilon + w + p_f$. We are free to choose ϵ: choose it to be $1/n$ so that δ is $1/\operatorname{poly}(n)$. So now the running time to get a distance δ is $t = O(n \log n)$. We then apply Lemma A.14 to obtain the result.

This concludes the proof of Theorem 5.1 so Corollary 5.1 is proved. □

We have now proven Lemma 2.1 and consequently Corollary 2.1. We now show how Theorem 2.2 follows.

6. Main Result

We will now show how the mixing time results imply that we have an approximate 2-design.

Proof (Proof of Theorem 2.2). We will go via the 2-norm since this gives a tight bound when working with the Pauli operators. The supremum can be taken over just physical states ρ [29]. We write ρ in the Pauli basis as usual (as Eq. 2.3).

$$
\begin{aligned}
||\mathcal{G}_W - \mathcal{G}_H||_\diamond^2 &= \sup_\rho ||(\mathcal{G}_W \otimes I)(\rho) - (\mathcal{G}_H \otimes I)(\rho)||_1^2 \\
&\leq 2^{4n} \sup_\rho ||(\mathcal{G}_W \otimes I)(\rho) - (\mathcal{G}_H \otimes I)(\rho)||_2^2 \\
&= \sup_\rho \Bigg|\Bigg| \sum_{\substack{p_1,p_2,p_3,p_4 \\ p_1p_2 \neq 00}} \gamma_0(p_1, p_2, p_3, p_4)(\mathcal{G}_W(\sigma_{p_1} \otimes \sigma_{p_2}) \otimes \sigma_{p_3} \otimes \sigma_{p_4} \\
&\qquad -\mathcal{G}_H(\sigma_{p_1} \otimes \sigma_{p_2}) \otimes \sigma_{p_3} \otimes \sigma_{p_4})\Bigg|\Bigg|_2^2.
\end{aligned}
$$

Now, write (for $p_1p_2 \neq 00$) $\mathcal{G}_W(\frac{1}{2^n}\sigma_{p_1} \otimes \sigma_{p_2}) = \frac{1}{2^n} \sum_{\substack{q_1,q_2 \\ q_1q_2 \neq 00}} g_t(q_1, q_2; p_1, p_2)\sigma_{q_1} \otimes \sigma_{q_2}$. We get

$$
\begin{aligned}
&\sup_\rho \Bigg|\Bigg| \sum_{\substack{p_1,p_2,p_3,p_4,q_1,q_2 \\ p_1p_2 \neq 00, q_1q_2 \neq 00}} \gamma_0(p_1, p_2, p_3, p_4) \left(g_t(q_1, q_2; p_1, p_2) - \frac{\delta_{q_1q_2}\delta_{p_1p_2}}{2^n(2^n+1)}\right) \\
&\qquad \times \sigma_{q_1} \otimes \sigma_{q_2} \otimes \sigma_{p_3} \otimes \sigma_{p_4}\Bigg|\Bigg|_2^2 \\
&= 2^{4n} \sup_\rho \sum_{\substack{p_1,p_2,p_3,p_4,q_1,q_2 \\ p_1p_2 \neq 00, q_1q_2 \neq 00}} \gamma_0^2(p_1, p_2, p_3, p_4) \left(g_t(q_1, q_2; p_1, p_2) - \frac{\delta_{q_1q_2}\delta_{p_1p_2}}{2^n(2^n+1)}\right)^2 \\
&\leq 2^{4n} \sup_\rho \sum_{\substack{p_1,p_2,p_3,p_4 \\ p_1p_2 \neq 00}} \gamma_0^2(p_1, p_2, p_3, p_4)\epsilon^2 \\
&\leq 2^{4n}\epsilon^2,
\end{aligned}
$$

where the first equality comes from the orthogonality of the Pauli operators under the Hilbert-Schmidt inner product and the last inequality comes from the fact that ρ is a physical state so has $\mathrm{tr}\,\rho^2 \leq 1$. This proves the result for the diamond norm, Definition 2.5. For the distance measure defined in Definition 2.6, the argument in [10] can be used together with the 1-norm bound to prove the result. □

It is unfortunate that there is still a dimension factor remaining in the above proof. To get a distance ϵ we have to run the random circuit for $O(n(n + \log 1/\epsilon))$ steps. However, closeness in the diamond-norm may be too stringent a requirement. After $O(n(n + \log 1/\epsilon))$ steps, the random circuit gives a 2-design in the measure used by Dankert et al. (see [10] and Definition 2.6). This is in contrast to the $O(n \log 1/\epsilon)$ steps required by the explicit circuit construction of Dankert et al.

7. Conclusions

We have proved tight convergence results for the first two moments of a random circuit. We have used this to show that random circuits are efficient approximate 1- and 2-unitary designs. Our framework readily generalises to k-designs for any k and the next step in this research is to prove that random circuits give approximate k-designs for all k.

We have shown that, provided the random circuit uses gates from a universal gate set that is also universal on $U(4)$, the circuit is still an efficient 2-design. We also see that the

random circuit with gates chosen uniformly from $U(4)$ is the most natural model. We note that the gates from $U(4)$ can be replaced by gates from any approximate 2-design on two qubits without any change to the asymptotic convergence properties.

One application of this work is to give an efficient method of decoupling two quantum systems by applying a random unitary from a 2-design to one system and then discarding part of it. This technique is used in [2] to construct a variety of encoding circuits for tasks in quantum Shannon theory; thus, we (like [10]) reduce the encoding complexity in [2] (and related works, such as [21]) to $O(n^2)$. Unfortunately, the decoding circuits still remain inefficient.

An algorithmic application of random circuits was given in [19], where they were used to construct a new class of superpolynomial quantum speedups. In that paper, random circuits of length $O(n^3)$ were used in order to guarantee that they were so-called "dispersing" circuits. Our results immediately imply that circuits of length $O(n^2)$ would instead suffice. We believe that this could be further improved with a specialised argument, since [19] assumed that the input to the random circuit was always a computational basis state.

Another potential application of random circuits is to model the evolution of black holes [22]. In Ref. [22], they conjecture that short random local quantum circuits are approximately 2-designs, and thus can be used for decoupling quantum systems (as in [2]). This, in turn, is used to make claims about the rate at which black holes leak information. While our model differs from that of Ref. [22] in that they consider nearest-neighbour interactions and we do not, our techniques and results could be readily extended to cover the case they consider.

Finally, random circuits are interesting physical models in their own right. The original purpose of [26] was to answer the physical question of how quickly entanglement grows in a system with random two party interactions. Lemma 2.1(i) shows that $O(n(n+\log 1/\epsilon))$ steps suffice (in contrast to $O(n^2(n+\log 1/\epsilon))$ which they prove) to give almost maximal entanglement in such a system.

Acknowledgements We are grateful for funding from the Army Research Office under grant W9111NF-05-1-0294, the European Commission under Marie Curie grants ASTQIT (FP6-022194) and QAP (IST-2005-15848), and the U.K. Engineering and Physical Science Research Council through "QIP IRC." We thank Raphaël Clifford, Ashley Montanaro and Dan Shepherd for helpful discussions.

A. Appendix

A.1. Permutation operators. The following theorems about permutation operators will be used repeatedly.

Lemma A.1. *Let C be a cycle of length c in S_c. Then*

$$\mathrm{tr}\ (C\ (A_1 \otimes A_2 \otimes \ldots \otimes A_c)) = \mathrm{tr}\ \left(A_{C(1)} A_{C^{\circ 2}(1)} A_{C^{\circ 3}(1)} \ldots A_1\right).$$

Proof. We have

$$\begin{aligned}\mathrm{tr}\ (C\ (A_1 \otimes A_2 \otimes \ldots \otimes A_c)) &= \sum_{i_1,i_2,\ldots,i_c} \langle i_1 i_2 \ldots i_c | C\ (A_1 \otimes A_2 \otimes \ldots \otimes A_c)\ | i_1 i_2 \ldots i_c \rangle \\ &= \sum_{i_1,i_2,\ldots,i_c} \langle i_1 | A_{C(1)} | i_{C(1)} \rangle \langle i_2 | A_{C(2)} | i_{C(2)} \rangle \\ &\quad \ldots \langle i_c | A_{C(c)} | i_{C(c)} \rangle\end{aligned}$$

$$= \sum_{i_1,i_2,\ldots,i_c} \langle i_1|A_{C(1)}|i_{C(1)}\rangle\langle i_{C(1)}|A_{C^{\circ 2}(1)}|i_{C^{\circ 2}(1)}\rangle \ldots \langle i_{C^{\circ c-1}(1)}|A_1|i_1\rangle$$

since $C^{\circ c}(1) = 1$. Evaluate the sum using the resolution of the identity to get the result. □

With this we can work out the Pauli expansion of the swap operator:

Lemma A.2. *The swap operator $\mathcal{F}$ on two d dimensional systems can be written as*

$$\frac{1}{d}\sum_p \sigma_p \otimes \sigma_p,$$

where $\{\sigma_p\}$ form a Hermitian orthogonal basis with $\operatorname{tr}\sigma_p^2 = d$.

Proof. Expand $\mathcal{F}$ in the basis and use Lemma A.1:

$$\operatorname{tr}\sigma_p \otimes \sigma_q \mathcal{F} = \operatorname{tr}\sigma_p\sigma_q = \begin{cases} d & p = q \\ 0 & \text{otherwise.} \end{cases}$$

The given sum has the correct coefficients in the basis, therefore $\frac{1}{d}\sum_p \sigma_p \otimes \sigma_p = \mathcal{F}$. □

A.2. Zero chain mixing time proofs.

A.2.1. Asymmetric simple random walk We will use some facts about asymmetric simple random walks, i.e. a random walk on a 1D line with probability p of moving right at each step and probability $q = 1 - p$ of moving left.

The position of the walk after k steps is tightly concentrated around $k(p - q)$:

Lemma A.3. *Let X_k be the random variable giving the position of a random walk after k steps starting at the origin with probability p of moving right and probability $q = 1 - p$ of moving left. Let $\mu = p - q$. Then for any $\eta > 0$,*

$$\mathbb{P}(X_k \geq \mu k + \eta) \leq \exp\left(-\frac{\eta^2}{2k}\right)$$

and

$$\mathbb{P}(X_k \leq \mu k - \eta) \leq \exp\left(-\frac{\eta^2}{2k}\right).$$

Proof. The standard Chernoff bound for 0/1 variables $\tilde{Y}_i$ gives, with $\tilde{Y}_i$ equal to 1 with probability p and for $Y_k = \sum_{i=1}^k \tilde{Y}_i$,

$$\mathbb{P}(Y_k \geq kp + \eta) \leq \exp\left(-\frac{2\eta^2}{k}\right),$$

$$\mathbb{P}(Y_k \leq kp - \eta) \leq \exp\left(-\frac{2\eta^2}{k}\right).$$

For our case, set $\tilde{Y}_i = 2\tilde{X}_i - 1$ to give the desired result. □

This result is for a walk with constant bias. We will need a result for a walk with varying (but bounded from below) bias:

Lemma A.4. *Let X_k be the random variable giving the position of a random walk after k steps starting at the origin with probability $p_i \geq p$ of moving right and probability $q_i \leq p$ of moving left at step i. Let $\mu = p - (1-p)$. Then for any $\eta > 0$,*

$$\mathbb{P}(X_k \geq \mu k + \eta) \leq \exp\left(-\frac{\eta^2}{2k}\right)$$

and

$$\mathbb{P}(X_k \leq \mu k - \eta) \leq \exp\left(-\frac{\eta^2}{2k}\right).$$

Proof. Let $\tilde{Y}_i$ be a random variable equal to 1 with probability p and 0 with probability $1-p$. Then let $\tilde{Z}_i$ be a random variable equal to 1 with probability p_i and 0 with probability $1-p_i$. Let $Y_k = \sum_{i=1}^k \tilde{Y}_i$ and $Z_k = \sum_{i=1}^k \tilde{Z}_i$. Then following the standard Chernoff bound derivation (for $\lambda > 0$),

$$\begin{aligned}
\mathbb{P}(Z_k \geq kp + \eta) &= \mathbb{P}\left(e^{\lambda Z_k} \geq e^{\lambda(kp+\eta)}\right) \\
&\leq \frac{e^{\lambda(kp+\eta)}}{\mathbb{E}e^{\lambda Z_k}} \\
&\leq \frac{e^{\lambda(kp+\eta)}}{\mathbb{E}e^{\lambda Y_k}} \\
&\leq \exp\left(-\frac{2\eta^2}{k}\right).
\end{aligned}$$

We can then, as above, set $\tilde{Z}_i = 2\tilde{X}_i - 1$. The calculation is similar for the bound on $\mathbb{P}(X_k \leq \mu k - \eta)$. □

From Lemma A.3 we can prove a result about how often each site is visited. If the walk runs for t steps the walk is at position $t\mu$ with high probability so we might expect from symmetry that each site will have been visited about $1/\mu$ times. Below is a weaker concentration result of this form but is strong enough for our purposes. It says that the amount of time spent $\leq x$ is about x/μ.

Lemma A.5. *For $\gamma > 2$ and integer $x > 0$,*

$$\mathbb{P}\left(\sum_{k=1}^{\infty} \mathbb{I}(X_k \leq x) \geq \gamma x/\mu\right) \leq 2\exp\left(-\frac{\mu x(\gamma-2)}{2}\right),$$

where $\mathbb{I}$ is the indicator function.

Proof. Let $Y_k = \mathbb{I}(X_k \leq x)$. From Lemma A.3,

$$\mathbb{P}(Y_k = 0) \leq \exp\left(-\frac{(k\mu - x)^2}{2k}\right)$$

for $k \leq x/\mu$ and

$$\mathbb{P}(Y_k = 1) \leq \exp\left(-\frac{(k\mu - x)^2}{2k}\right)$$

for $k \geq x/\mu$.

Then the quantity to evaluate is

$$\mathbb{P}\left(\sum_{k=1}^{\infty} Y_k \geq \gamma x/\mu\right).$$

We use a standard trick to split this into two mutually exclusive possibilities and then bound the probabilities separately. Write

$$\begin{aligned}
&\mathbb{P}\left(\sum_{k=1}^{\infty} Y_k \geq \gamma x/\mu\right) \\
&\quad = \mathbb{P}\left(\left(\sum_{k=1}^{\infty} Y_k \geq \gamma x/\mu\right) \bigcap \left(\bigcap_{j=1}^{\gamma x/\mu} [Y_j = 1]\right)\right) \\
&\qquad + \mathbb{P}\left(\left(\sum_{k=1}^{\infty} Y_k \geq \gamma x/\mu\right) \bigcap \left(\bigcup_{j=1}^{\gamma x/\mu} [Y_j = 0]\right)\right). \qquad \text{(A.1)}
\end{aligned}$$

We can bound the first term:

$$\begin{aligned}
\mathbb{P}\left(\left(\sum_{k=1}^{\infty} Y_k \geq \gamma x/\mu\right) \bigcap \left(\bigcap_{j=1}^{\gamma x/\mu} [Y_j = 1]\right)\right) &= \mathbb{P}\left(\bigcap_{k=1}^{\gamma x/\mu} Y_k = 1\right) \\
&\leq \mathbb{P}\left(Y_{\gamma x/\mu} = 1\right) \\
&\leq \exp\left(-\frac{\mu x(\gamma - 1)^2}{2\gamma}\right) \\
&\leq \exp\left(-\frac{\mu x(\gamma - 2)}{2}\right).
\end{aligned}$$

The second term is done similarly:

$$\begin{aligned}
\mathbb{P}\left(\left(\sum_{k=1}^{\infty} Y_k \geq \gamma x/\mu\right) \bigcap \left(\bigcup_{j=1}^{\gamma x/\mu} [Y_j = 0]\right)\right) &\leq \mathbb{P}\left(\bigcup_{k=\frac{\gamma x}{\mu}+1}^{\infty} [Y_k = 1]\right) \\
&\leq \sum_{k=\frac{\gamma x}{\mu}+1}^{\infty} \mathbb{P}\left(Y_k = 1\right) \\
&\leq \sum_{k=\frac{\gamma x}{\mu}+1}^{\infty} \exp\left(-\frac{(k\mu - x)^2}{2k}\right) \\
&\leq \exp\left(-\frac{\mu x(\gamma - 2)}{2}\right).
\end{aligned}$$

The last fact we need about asymmetric simple random walks is a bound on the probability of going backwards. If $p > q$ then we expect the walk to go right in the majority of steps. The probability of going left a distance a is exponentially small in a. This is a well known result, often stated as part of the gambler's ruin problem:

Lemma A.6 (see e.g. [17]). *Consider an asymmetric simple random walk that starts at $a > 0$ and has an absorbing barrier at the origin. The probability that the walk eventually absorbs at the origin is 1 if $p \leq q$ and $(q/p)^a$ otherwise.*

This result is for infinitely many steps. If we only consider finitely many steps, the probability of absorption must be at most this.

A.2.2. Waiting time From above we saw that the probability of moving is at least $2x/5n$ when at position x. The length of time spent waiting at each step is therefore stochastically dominated by a geometric distribution with parameter $2x/5n$. The following concentration result will be used to bound the waiting time (in our case $\beta = 2/5$):

Lemma A.7. *Let the waiting time at each site be $W(x) \sim Geo(\beta x/n)$, the total waiting time $W = \sum_{x=1}^{t} W(x)$ and $t' = \frac{n \ln t}{\beta}$. Then*

$$\mathbb{P}(W \geq Ct') \leq 2t^{(1-C)/2}.$$

Proof. By Markov's inequality for $\lambda > 0$,

$$\mathbb{P}(W \geq Ct') \leq \frac{\mathbb{E}e^{\lambda W}}{e^{\lambda Ct'}}.$$

The $W(x)$ are independent so

$$\mathbb{E}e^{\lambda W} = \prod_{x=1}^{t} \mathbb{E}e^{\lambda W(x)}.$$

Summing the geometric series we find

$$\mathbb{E}e^{\lambda W(x)} = \frac{\frac{\beta x}{n}}{e^{-\lambda} - 1 + \frac{\beta x}{n}},$$

provided $e^{\lambda} < \frac{1}{1-\frac{\beta x}{n}}$ for all $1 \leq x \leq t$. Therefore e^{λ} is of the form $\frac{1}{1-\frac{\alpha\beta}{n}}$, where $0 < \alpha < 1$. With this,

$$\mathbb{E}e^{\lambda W(x)} = \frac{x}{x - \alpha}$$

and

$$\mathbb{E}e^{\lambda W} = \frac{t!\Gamma(1-\alpha)}{\Gamma(t+1-\alpha)}.$$

We are free to choose α within its range to optimise the bound. However, for simplicity, we will choose $\alpha = 1/2$. From Lemma A.13,

$$\mathbb{E}e^{\lambda W} \leq 2\sqrt{t}.$$

The result follows, using the inequality $1 - x \leq e^{-x}$. □

A.2.3. Phase 1 Here we prove that phase 1 completes successfully with high probability. The bias here is large so the walk moves right every time with high probability:

Lemma A.8. *The probability that the accelerated chain moves right at each step, starting from $x = 1$ for t steps, is at least*

$$1 - t^2/n.$$

Proof. The probability of moving right at each step is

$$\begin{aligned}\prod_{x=1}^{t} \frac{3(n-x)}{3n-2x-1} &= \frac{(n-2)(n-3)\dots(n-t)}{(n-5/3)(n-7/3)\dots(n-(2t+1)/3)} \\ &\geq (1-2/n)(1-3/n)\dots(1-t/n) \\ &\geq (1-t/n)^t \geq 1 - t^2/n.\end{aligned}$$

□

Let $t = n^\delta$. Provided $\delta < 1/2$ this probability is close to one. Therefore, with high probability, the walk moves to n^δ in n^δ steps. Using Lemma A.7 the waiting time can be bounded:

Lemma A.9. *Let $W^{(1)}$ be the waiting time during phase 1. Let H be the event that the walk moves right at each step. Then*

$$\mathbb{P}\left(W^{(1)} \geq Ct' | H\right) \leq 2n^{\delta(1-C)/2}, \tag{A.2}$$

where $t' = \frac{5\delta n \ln n}{2}$.

Proof. This follows directly from Lemma A.7, since each site is hit exactly once. □

We now combine these two lemmas to prove that phase 1 completes successfully with high probability:

Proof (Proof of Lemma 5.4). In Lemma A.8, we show that in n^δ accelerated steps, the walk moves right at each step with probability $\geq 1 - n^{2\delta-1}$. Call this event H. Then $\mathbb{P}(H) \geq 1 - n^{2\delta-1}$. Lemma A.9 shows that the waiting time $W^{(1)}$ is bounded with high probability (choosing $C = 3$):

$$\mathbb{P}(W^{(1)} \leq 15n\delta \ln n/2 | H) \geq 1 - 2n^{-\delta}.$$

Then we can bound the probability of phase 1 completing successfully:

$$\begin{aligned}\mathbb{P}(\text{Phase 1 completes successfully}) &\geq \mathbb{P}(H \cap W^{(1)} \leq 15n\delta \ln n/2) \\ &= \mathbb{P}(H)\mathbb{P}(W^{(1)} \leq 15n\delta \ln n/2 | H) \\ &\geq (1 - n^{2\delta-1})(1 - 2n^{-\delta}) \\ &\geq 1 - n^{2\delta-1} - 2n^{-\delta}.\end{aligned}$$

□

A.2.4. Phase 2 Phase 2 starts at $n^\delta/2$ and finishes when the walk has reached θn for some constant $0 < \theta < 3/4$. We show that, with high probability, this also takes time $O(n \log n)$. The probability of moving right during this phase is at least $p = \frac{3(1-\theta)}{3-2\theta}$. We first define some constants that we will derive bounds in terms of. Let γ be a constant > 2. Let $\mu = p - (1-p)$ and $\tilde{\mu} = \mu/\gamma$. Finally let $s = \tilde{\mu}t$ for some t (which will be the number of accelerated steps). Then, with high probability, the walk will have passed s after t steps:

Lemma A.10. *Let X_t be the position of the walk at accelerated step t, where $X_0 = n^\delta$. Then*

$$\mathbb{P}(X_t \leq s) \leq \exp(-\mu^2 t(1-1/\gamma)^2/2).$$

Proof. Let $X'_t = X_t - n^\delta$. Then from Lemma A.4,

$$\mathbb{P}(X'_t \leq \mu t - \eta) \leq \exp\left(-\frac{\eta^2}{2t}\right).$$

Now let $\eta = \mu t - s$ and use

$$\begin{aligned}\mathbb{P}(X_t \leq s) &= \mathbb{P}(X'_t \leq s - n^\delta) \\ &\leq \mathbb{P}(X'_t \leq s)\end{aligned}$$

to complete the proof. □

We now prove a bound on the waiting time:

Lemma A.11. *Let $W^{(2)}$ be the waiting time in phase 2. Then, assuming the walk does not go back beyond $n^\delta/2$,*

$$\mathbb{P}\left(W^{(2)} \geq \frac{15n \ln s}{\mu}\right) \leq (4/s)^{3/2\mu} + \frac{2\exp\left(\frac{-\mu n^\delta}{4}\right)}{1-\exp\left(\frac{-\mu}{2}\right)}. \tag{A.3}$$

Proof. Let $W_k \sim Geo\left(\frac{2X_k}{5n}\right)$, where X_k is the position of the walk at accelerated step k ($X_0 = n^\delta$). We want to bound (w.h.p.) the waiting time $W^{(2)} = \sum_{k=1}^{t} W_k$ of t steps of the accelerated walk.

Define the event H to be

$$H = \left\{ \bigcap_{x \geq n^\delta/2} \left[\sum_{k=1}^{\infty} \mathbb{I}(X_k \leq x) \leq x/\tilde{\mu} \right] \right\}. \tag{A.4}$$

If H occurs, no sites have been hit too often and the walk has not gone back further than $n^\delta/2$. It is important that we also use the restriction that $X_k \geq n^\delta/2$ because the waiting time grows the longer the walk moves back. However, it is very unlikely that the walk will go backwards (even to $n^\delta/2$).

We now define some more notation to bound the waiting time. Let $\mathbf{X} = (X_1, X_2, \ldots, X_t)$ be a tuple of positions and let $N_x(\mathbf{X})$ be the number of times that x appears in $\mathbf{X}$ and let $\mathbf{N}(\mathbf{X}) = (N_1(\mathbf{X}), N_2(\mathbf{X}), \ldots, N_n(\mathbf{X}))$. Then we have $\sum_x N_x(\mathbf{X}) = t$.

As we said above, the waiting time at $x = a$ stochastically dominates the waiting time at $x = b$ for $b \geq a$. In other words,

$$W_k \trianglerighteq W_{k'} \quad \text{if } X_k \leq X_{k'}, \tag{A.5}$$

where $X \trianglerighteq Y$ means that X stochastically dominates Y. Now write the waiting time for all steps:

$$\begin{aligned} W^{(2)}(\mathbf{X}) &= \sum_{k=1}^{t} W_k \\ &= \sum_{x} \sum_{h=1}^{N_x(\mathbf{X})} W_h(x), \end{aligned} \tag{A.6}$$

where $W_h(x) \sim Geo\left(\frac{2x}{5n}\right)$.

If event H occurs, we can put some bounds on N_x. We find that, for all $x \geq n^\delta/2$,

$$\sum_{y=n^\delta/2}^{x} N_y(\mathbf{X}) \leq x/\tilde{\mu} \tag{A.7}$$

and $N_x(\mathbf{X}) = 0$ for $x < n^\delta/2$. Now let $\mathbf{X}_m$ be such that $N_{n^\delta/2}(\mathbf{X}_m) = \frac{n^\delta}{2\tilde{\mu}}$ and $N_x(\mathbf{X}_m) = 1/\tilde{\mu}$ for $x > n^\delta/2$. Then

$$\sum_{y=n^\delta/2}^{x} N_y(\mathbf{X}_m) = x/\tilde{\mu}. \tag{A.8}$$

Now we introduce the relation $\preceq$:

Definition A.1. *Let* $\mathbf{x}$ *and* $\mathbf{y}$ *be n-tuples. Then* $x \preceq y$ *if*

$$\sum_{i=1}^{k} x_i \leq \sum_{i=1}^{k} y_i \tag{A.9}$$

for all $1 \leq k \leq n$ *with equality for* $k = n$.

Note that this is like majorisation, except the elements of the tuples are not sorted. Using this, we find that $\mathbf{N}(\mathbf{X}) \preceq \mathbf{N}(\mathbf{X}_m)$. (Using $\sum_y N_y(\mathbf{X}) = \sum_y N_y(\mathbf{X}') = t$ for all $\mathbf{X}, \mathbf{X}'$.)

If we combine Eqs. A.5 and A.6 we find that $W^{(2)}(\mathbf{X}) \trianglerighteq W^{(2)}(\mathbf{X}')$ if $\mathbf{N}(\mathbf{X}) \succeq \mathbf{N}(\mathbf{X}')$. Roughly speaking, this is simply saying that the waiting time is larger if the earlier sites are hit more often. But since for all $\mathbf{X}$ that satisfy H, $\mathbf{X} \preceq \mathbf{X}_m$, we have $W^{(2)}(\mathbf{X}) \trianglelefteq W^{(2)}(\mathbf{X}_m)$ provided H occurs. We will simplify further by noting that $\mathbf{X_m} \preceq \mathbf{X}_0$, where $N_x(\mathbf{X}_0) = 1/\tilde{\mu}$ for $1 \leq x \leq \tilde{\mu}t = s$ and zero elsewhere. Therefore

$$\mathbb{P}\left(W^{(2)}(\mathbf{X}) \geq \frac{5Cn \ln s}{2\tilde{\mu}} \middle| H\right) \leq \mathbb{P}\left(W^{(2)}(\mathbf{X_0}) \geq \frac{5Cn \ln s}{2\tilde{\mu}}\right).$$

We can bound this by applying Lemma A.7. Let $W_h = \sum_{x=1}^{s} W_h(x)$. From Lemma A.7,

$$\mathbb{P}(W_h \geq Ct') \leq 2s^{\frac{1-C}{2}}, \tag{A.10}$$

where $t' = \frac{5n \ln s}{2}$. However, we want a bound on $\mathbb{P}\left(\sum_{h=1}^{1/\tilde{\mu}} W_h \geq Ct'/\tilde{\mu}\right)$. The same reasoning as in Lemma A.7 bounds this as

$$\mathbb{P}\left(\sum_{h=1}^{1/\tilde{\mu}} W_h \geq Ct'/\tilde{\mu}\right) \leq \left(2s^{\frac{1-C}{2}}\right)^{1/\tilde{\mu}}. \tag{A.11}$$

Therefore

$$\mathbb{P}\left(W^{(2)}(\mathbf{X_0}) \geq \frac{5Cn \ln s}{2\tilde{\mu}}\right) \leq 2^{1/\tilde{\mu}} s^{\frac{(1-C)/2}{\tilde{\mu}}}. \tag{A.12}$$

To complete the proof, we just need to find $\mathbb{P}(H^c)$. We can bound it using the union bound and Lemma A.5:

$$\begin{aligned}
\mathbb{P}(H^c) &= \mathbb{P}\left(\bigcup_{x=n^\delta/2}^{n} \left[\sum_{k=1}^{\infty} \mathbb{I}(X_k \leq x) > x/\tilde{\mu}\right]\right) \\
&\leq \sum_{x=n^\delta/2}^{n} \mathbb{P}\left(\sum_{k=1}^{\infty} \mathbb{I}(X_k \leq x) \geq x/\tilde{\mu}\right) \\
&\leq \sum_{x=n^\delta/2}^{n} 2\exp\left(\frac{-\mu x(\gamma-2)}{2}\right) \\
&\leq \sum_{x=n^\delta/2}^{\infty} 2\exp\left(\frac{-\mu x(\gamma-2)}{2}\right) \\
&= \frac{2\exp\left(\frac{-\mu n^\delta(\gamma-2)}{4}\right)}{1-\exp\left(\frac{-\mu(\gamma-2)}{2}\right)}.
\end{aligned}$$

Now, for any events A and B,

$$\begin{aligned}
\mathbb{P}(A) &= \mathbb{P}(A \cap B) + \mathbb{P}(A \cap B^c) \\
&= \mathbb{P}(A|B)\mathbb{P}(B) + \mathbb{P}(A \cap B^c) \\
&\leq \mathbb{P}(A|B) + \mathbb{P}(B^c),
\end{aligned}$$

and set $C = 2$ and $\gamma = 3$ to obtain the result. □

We now combine these two lemmas to prove that phase 2 completes successfully with high probability:

Proof (Proof of Lemma 5.5). Phase 2 can fail if:

– The walk does not reach θn. The probability of this is bounded by Lemma A.10:

$$\mathbb{P}(X_t \leq \theta n) \leq \exp\left(-\frac{2}{3}\mu\theta n\right).$$

This follows from setting $t = \frac{3\theta n}{\mu}$ and $\gamma = 3$.

– The waiting time is too long. This probability is bounded by Lemma A.11:

$$\mathbb{P}\left(W^{(2)} \geq \frac{15n\ln(\theta n)}{\mu}\right) \leq \left(\frac{4}{\theta n}\right)^{\frac{3}{2\mu}} + \frac{2\exp\left(\frac{-\mu n^{\delta}}{4}\right)}{1-\exp(-\mu/2)} + (q/p)^{n^{\delta}/2}.$$

– The walk gets back to $n^{\delta}/2$. This is bounded by Lemma A.6:

$$\mathbb{P}\left(\text{Walk gets to } n^{\delta}/2\right) \leq (q/p)^{n^{\delta}/2}.$$

So, using the union bound we can bound the overall probability of failure:

$$\mathbb{P}(\text{Phase 2 fails}) \leq \exp\left(-\frac{2}{3}\mu\theta n\right) + \left(\frac{4}{\theta n}\right)^{\frac{3}{2\mu}} + \frac{2\exp\left(\frac{-\mu n^{\delta}}{4}\right)}{1-\exp(-\mu/2)} + (q/p)^{n^{\delta}/2}.$$

A.2.5. Phase 3 This phase starts at θn. We show that this mixes quickly using log-Sobolev arguments.

Lemma A.12. *The zero chain on the restricted state space $x \in [m, n]$, where $m = \theta n/2$ for $0 \leq \theta \leq 3/4$, has mixing time $O\left(n \log \frac{n}{\epsilon}\right)$.*

Proof. We restrict the Markov chain to only run from m by adjusting the holding probability at m, $P(m, m)$. Construct the chain P' with transition matrix

$$P'(x, y) = \begin{cases} 0 & x < m \text{ or } y < m \\ 1 - P(m, m+1) & x = y = m \\ P(x, y) & \text{otherwise} \end{cases}, \tag{A.13}$$

where P is the transition matrix of the full zero chain. This chain then has stationary distribution

$$\pi'(x) = \begin{cases} \pi(x)/(1-w) & m \leq x \leq n \\ 0 & \text{otherwise} \end{cases}, \tag{A.14}$$

where $w = \sum_{x=1}^{m-1} \pi(x)$. To see this, first note that the distribution is normalised. We want to show that

$$\sum_{x=m}^{n} P'(x, y)\pi'(x) = \pi'(y). \tag{A.15}$$

When $y = m$ we are required to prove that $P'(m, m)\pi'(m) + P'(m+1, m)\pi'(m+1) = \pi'(m)$. This follows from the reversibility of the unrestricted zero chain, using $P'(m, m) = 1 - P(m, m+1)$. For $y > m$, Eq. A.15 is satisfied simply because $\pi(x)$ is the stationary distribution of P and related by a constant factor to $\pi'(x)$.

We can now prove this final mixing time result, making use of Lemma 4.4. Let Q_i be the chain that uniformly mixes site i. This converges in one step and has a log-Sobolev constant independent of n; call it ρ_1. Let Q be the chain that chooses a site at random and then uniformly mixes that site. This is the product chain of the Q_i so, by Lemma 4.4, has gap $1/n$ and Sobolev constant $\rho_Q = \rho_1/n$. We can construct the zero chain for this and find its Sobolev constant.

The Sobolev constant is defined (Definition 4.3) in terms of a minimisation over functions on the state space. For the chain Q we can write

$$\rho_Q = \inf_{\phi} f(\phi).$$

If we restrict the infimum to be over functions ϕ with $\phi(x) = \phi(y)$ for x and y containing the same number of zeroes then we obtain the Sobolev constant for the zero-Q chain, ρ_{Q_0}, which is the chain which counts the number of zeroes in the full chain Q. Since taking the infimum over less functions cannot give a smaller value,

$$\rho_{Q_0} \geq \rho_Q \geq \rho_1/n.$$

We can now compare this chain to the zero-P chain. The stationary distributions are the same. The transition matrix for the zero-Q chain is

$$Q_0(x, y) = \begin{cases} \frac{n+2x}{4n} & y = x \\ \frac{x}{4n} & y = x-1 \\ \frac{3(n-x)}{4n} & y = x+1 \\ 0 & \text{otherwise} \end{cases}.$$

Then construct Q_0' by restricting the space to only run from m in exactly the same way as P' is constructed from P. Q_0' has the same stationary distribution as P'. Now we can perform the comparison. From Eq. 4.14:

$$\begin{aligned} A &= \max_{a \geq m} \frac{Q_0'(a, a+1)}{P'(a, a+1)} \\ &= \max_{a \geq m} \frac{5(n-1)}{8a} \leq \frac{5}{8\theta}. \end{aligned}$$

Therefore $\rho_{P'} \geq \frac{8\theta\rho_1}{5n}$. Exactly the same argument applies to show the gap is $\Omega(1/n)$, so the mixing time is (from Eq. 4.16) $O(n \log \frac{n}{\epsilon})$. □

Now we can prove that phase 3 completes successfully with high probability:

Proof (of Lemma 5.6). In Lemma A.12, we show that after $O\left(n \log \frac{n}{\epsilon}\right)$ steps the chain mixes to distance ϵ. We just need to show that the walk goes back to $\theta n/2$ with small probability. This follows from Lemma A.6.

A.3. Moment generating function calculations. The following lemma is needed in the moment generating function calculations.

Lemma A.13. *For Integer $s > 0$,*

$$\frac{\Gamma(s+1)\Gamma(1/2)}{\Gamma(s+1/2)} \leq 2\sqrt{s}. \tag{A.16}$$

Proof. From expanding the Γ functions, Eq. A.16 becomes

$$\frac{s!2^s}{(2s-1)!!} = \frac{2\times 4\times 6\times\cdots\times 2(s-1)\times 2s}{1\times 3\times 5\times\cdots\times(2s-3)\times(2s-1)} = \prod_{x=1}^{s}\frac{2x}{2x-1}.$$

We then proceed by induction. $\prod_{x=1}^{1}\frac{2x}{2x-1}=2$ and by the inductive hypothesis

$$\prod_{x=1}^{s+1}\frac{2x}{2x-1} \le \frac{2(s+1)}{2(s+1)-1}2\sqrt{s}.$$

It is easy to show that $\frac{2(s+1)}{2(s+1)-1} \le \sqrt{\frac{s+1}{s}}$ and the result follows. □

A.4. Mixing times. We find bounds for the mixing time above that are valid with high probability. Below we turn these into full mixing time bounds.

Lemma A.14. *If after $O(n\log n)$ steps the state v of a random walk satisfies*

$$||v-\pi|| \le \delta,$$

where π is the stationary distribution and δ is $1/poly(n)$, then the number of steps required to be at most a distance ϵ from stationarity is

$$O\left(n\log\frac{n}{\epsilon}\right).$$

Proof. Let s be the slowest mixing initial state. Then, after $t = O(n\log n)$ steps we have at worst the state

$$(1-\delta)\pi + \delta s,$$

and if we repeat kt times δ becomes δ^k. So to get a distance ϵ, $k = \left\lceil\frac{\log\epsilon}{\log\delta}\right\rceil$.

Now we evaluate the mixing time:

$$kt = O(n\log n)\left\lceil\frac{\log\epsilon}{\log\delta}\right\rceil = O(n\log n)\left\lceil\frac{\log 1/\epsilon}{\log 1/\delta}\right\rceil = O(n\max(\log n,\log 1/\epsilon)) = O\left(n\log\frac{n}{\epsilon}\right).$$

References

1. Aaronson, S.: *Quantum Copy-Protection.* Talk at QIP, New Delhi, India, December 2007, available at http://www.scottaaronson.com/talks/copy.ppt, 2007

2. Abeyesinghe, A., Devetak, I., Hayden, P., Winter, A.: *The mother of all protocols: Restructuring quantum information's family tree*. http://arxiv.org/abs/:quant-ph/0606225v1, 2006
3. Ambainis, A., Emerson, E.: *Quantum t-designs: t-wise independence in the quantum world*. IEEE Conference on Computational Complexity 2007, http://arxiv.org/abs/:quant-ph/0701126v2, 2007
4. Ambainis, A., Mosca, M., Tapp, A., de Wolf, R.: Private Quantum Channels. FOCS 2000, Washington, DC: IEEE, 2000, pp. 547–553
5. Ambainis, A., Smith, A.: *Small pseudo-random families of matrices: derandomizing approximate quantum encryption*. Lecture Notes in Computer Science **3122**, Berlin-Heidelberg-NewYork: Springer, 2004, pp. 249–260
6. Arnold, V.I., Krylov, A.L.: Uniform distribution of points on a sphere and some ergodic properties of solutions of linear ordinary differential equations in a complex domain. Sov. Math. Dokl. **4**(1), 1962
7. Barenco, A., Berthiaume, A., Deutsch, D., Ekert, A., Jozsa, R., Macchiavello, C.: Stabilization of quantum computations by symmetrization. SIAM J. Comput. **26**(5), 1541–1557 (1997)
8. Barnum, H.: *Information-disturbance tradeoff in quantum measurement on the uniform ensemble and on the mutually unbiased bases*. http://arxiv.org/abs/:quant-ph/0205155v1, 2002
9. Dahlsten, O.C.O., Oliveira, R., Plenio, M.B.: The emergence of typical entanglement in two-party random processes. J. Phys. A Math. Gen. **40**, 8081–8108 (2007)
10. Dankert, C., Cleve, R., Emerson, J., Livine, E.: *Exact and approximate unitary 2-designs: constructions and applications*. http://arxiv.org/abs/:quant-ph/0606161v1, 2006
11. Devetak, I., Junge, M., King, C., Ruskai, M.B.: Multiplicativity of completely bounded p-norms implies a new additivity result. Commun. Math. Phys. **266**, 37–63 (2006)
12. Diaconis, P., Saloff-Coste, L.: Comparison theorems for reversible markov chains. Ann. Appl. Probab. **3**(3), 696–730 (1993)
13. Diaconis, P., Saloff-Coste, L.: Logarithmic Sobolev inequalities for finite Markov chains. Ann. Appl. Probab. **6**(3), 695–750 (1996)
14. DiVincenzo, D., Leung, D., Terhal, B.: Quantum data hiding. Information Theory. IEEE Transactions **48**(3), 580–598 (2002)
15. Emerson, J., Livine, E., Lloyd, S.: Convergence conditions for random quantum circuits. Phys. Rev. A **72**, 060302 (2005)
16. Goodman, R., Wallach, N.: *Representations and Invariants of the Classical Groups.* Cambridge: Cambridge University Press, 1998
17. Grimmett, G., Welsh, D.: *Probability*: An Introduction. Oxford: Oxford University Press, 1986
18. Gross, D., Audenaert, K., Eisert, J.: Evenly distributed unitaries: On the structure of unitary designs. J. Math. Phys. **48**, 052104 (2007)
19. Hallgren, S., Harrow, A.W.: Superpolynomial speedups based on almost any quantum circuit. In: Proc. 35th Intl. Colloq. on Automate Languages an Programming LCUS **5125**, 2, pp. 782–795, 2008
20. Hayashi, A., Hashimoto, T., Horibe, M.: Reexamination of optimal quantum state estimation of pure states. Phys. Rev. A **72**, 032325 (2006)
21. Hayden, P., Horodecki, M., Yard, J., Winter, A.: A decoupling approach to the quantum capacity. Open Syst. Inf. Dyn. **15**, 7–19 (2008)
22. Hayden, P., Preskill, J.: *Black holes as mirrors: quantum information in random subsystems*. JHEP **09**, 120 (2007)
23. Hoory, S., Brodsky, A.: *Simple Permutations Mix Even Better*. http://arxiv.org/abs/math/0411098v2[math.CO] 2004
24. Kitaev, A.Yu., Shen, A.H., Vyalyi, M.N.: *Classical and Quantum Computation.* Providence, RI Amer. Math. Soc. (2002)
25. Montenegro, R., Tetali, P.: Mathematical aspects of mixing times in Markov chains. Found. Trends Theor. Comput. Sci. **1**(3), 237–354 (2006)
26. Oliveira, R., Dahlsten, O.C.O., Plenio, M.B.: Efficient generation of generic entanglement. Phys. Rev. Lett. **98**, 130502, (2007)
27. Paulsen, V.I.: *Completely Bounded Maps and Dilations*. New York: John Wiley & Sons, Inc., 1987
28. Sen, P.: Random measurement bases, quantum state distinction and applications to the hidden subgroup problem. IEEE Conference on Computational Complexity 2006, 2005, pp. 274–287
29. Watrous, J.: Notes on super-operator norms induced by Schatten norms. Quantum Information and Computation **5**(1), 58–68 (2005)
30. Znidaric, M.: Optimal two-qubit gate for generation of random bipartite entanglement. Phys. Rev. A **76**, 012318 (2007)

Communicated by A. Connes

A.2 ICALP Paper

Superpolynomial Speedups Based on Almost Any Quantum Circuit

Sean Hallgren[1] and Aram W. Harrow[2]

[1] Department of Computer Science and Engineering, The Pennsylvania State University
University Park, PA
[2] Department of Mathematics, University of Bristol, Bristol, U.K.
a.harrow@bris.ac.uk

Abstract. The first separation between quantum polynomial time and classical bounded-error polynomial time was due to Bernstein and Vazirani in 1993. They first showed a $O(1)$ vs. $\Omega(n)$ quantum-classical oracle separation based on the quantum Hadamard transform, and then showed how to amplify this into a $n^{O(1)}$ time quantum algorithm and a $n^{\Omega(\log n)}$ classical query lower bound.

We generalize both aspects of this speedup. We show that a wide class of unitary circuits (which we call *dispersing* circuits) can be used in place of Hadamards to obtain a $O(1)$ vs. $\Omega(n)$ separation. The class of dispersing circuits includes all quantum Fourier transforms (including over nonabelian groups) as well as nearly all sufficiently long random circuits. Second, we give a general method for amplifying quantum-classical separations that allows us to achieve a $n^{O(1)}$ vs. $n^{\Omega(\log n)}$ separation from any dispersing circuit.

1 Background

Understanding the power of quantum computation relative to classical computation is a fundamental question. When we look at which problems can be solved in quantum but not classical polynomial time, we get a wide range: quantum simulation, factoring, approximating the Jones polynomial, Pell's equation, estimating Gauss sums, period-finding, group order-finding and even detecting some mildly non-abelian symmetries [Sho97, Hal07, Wat01, FIM$^+$03, vDHI03]. However, when we look at what algorithmic tools exist on a quantum computer, the situation is not nearly as diverse. Apart from the BQP-complete problems [AJL06], the main tool for solving most of these problems is a quantum Fourier transform (QFT) over some group. Moreover, the successes have been for cases where the group is abelian or close to abelian in some way. For sufficiently nonabelian groups, there has been no indication that the transforms are useful even though they can be computed exponentially faster than classically. For example, while an efficient QFT for the symmetric group has been intensively studied for over a decade because of its connection to graph isomorphism, it is still unknown whether it can be used to achieve any kind of speedup over classical computation [Bea97].

The first separation between quantum computation and randomized computation was the Recursive Fourier Sampling problem (RFS) [BV97]. This algorithm had two components, namely using a Fourier transform, and using recursion. Shortly after this,

L. Aceto et al. (Eds.): ICALP 2008, Part I, LNCS 5125, pp. 782–795, 2008.

Simon's algorithm and then Shor's algorithm for factoring were discovered, and the techniques from these algorithms have been the focus of most quantum algorithmic research since [Sim97, Sho97]. These developed into the hidden subgroup framework. The hidden subgroup problem is an oracle problem, but solving certain cases of it would result in solutions for factoring, graph isomorphism, and certain shortest lattice vector problems. Indeed, it was hoped that an algorithm for graph isomorphism could be found, but recent evidence suggests that this approach may not lead to one [HMR$^+$06]. As a way to understand new techniques, this oracle problem has been very important, and it is also one of the very few where super-polynomial speedups have been found [IMS01, BCvD05].

In comparison to factoring, the RFS problem has received much less attention. The problem is defined as a property of a tree with labeled nodes and it was proven to be solvable with a quantum algorithm super-polynomially faster than the best randomized algorithm. This tree was defined in terms of the Fourier coefficients over $\mathbb{Z}_2^n$. The definition was rather technical, and it seemed that the simplicity of the Fourier coefficients for this group was necessary for the construction to work. Even the variants introduced by Aaronson [Aar03] were still based on the same QFT over $\mathbb{Z}_2^n$, which seemed to indicate that this particular abelian QFT was a key part of the quantum advantage for RFS.

The main result of this paper is to show that the RFS structure can be generalized far more broadly. In particular, we show that an RFS-style super-polynomial speedup is achievable using almost any quantum circuit, and more specifically, it is also true for any Fourier transform (even nonabelian), not just over $\mathbb{Z}_2^n$. This illustrates a more general power that quantum computation has over classical computation when using recursion. The condition for a quantum circuit to be useful for an RFS-style speedup is that the circuit be *dispersing*, a concept we introduce to mean that it takes many different inputs to fairly even superpositions over most of the computational basis.

Our algorithm should be contrasted with the original RFS algorithm. One of the main differences between classical and quantum computing is so-called garbage that results from computing. It is important in certain cases, and crucial in recursion-based quantum algorithms because of quantum superpositions, that intermediate computations are uncomputed and that errors do not compound. The original RFS paper [BV97] avoided the error issue by using an oracle problem where every quantum state create from it had the exact property necessary with no errors. Their algorithm could have tolerated polynomially small errors, but in this paper we relax this significantly. We show that even if we can only create states with constant accuracy at each level of recursion, we can still carry through a recursive algorithm which introduces new constant-sized errors a polynomial number of times.

The main technical part of our paper shows that most quantum circuits can be used to construct separations relative to appropriate oracles. To understand the difficulty here, consider two problems that occur when one tries to define an oracle whose output is related to the amplitudes that result from running a circuit. First, it is not clear how to implement such an oracle since different amplitudes have different magnitudes, and only phases can be changed easily. Second, we need an oracle where we can prove that a classical algorithm requires many queries to solve the problem. If the oracle outputs many bits, this can be difficult or impossible to achieve. For example, the matrix

entries of nonabelian groups can quickly reveal which representation is being used. To overcome these two problems we show that there are binary-valued functions that can approximate the complex-valued output of quantum circuits in a certain way.

One by-product of our algorithm is related to the Fourier transform of the symmetric group. Despite some initial promise for solving graph isomorphism, the symmetric group QFT has still not found any application in quantum algorithms. One instance of our result is the first example of a problem (albeit a rather artificial one) where the QFT over the symmetric group is used to achieve a super-polynomial speedup.

2 Statement of Results

Our main contributions are to generalize the RFS algorithm of [BV97] in two stages. First, [BV97] described the problem of Fourier sampling over $\mathbb{Z}_2^n$, which has an $O(1)$ vs. $\Omega(n)$ separation between quantum and randomized complexities. We show that here the QFT over $\mathbb{Z}_2^n$ can be replaced with a QFT over any group, or for that matter with almost any quantum circuit. Next, [BV97] turned Fourier sampling into recursive Fourier sampling with a recursive technique. We will generalize this construction to cope with error and to amplify a larger class of quantum speedups. As a result, we can turn any of the linear speedups we have found into superpolynomial speedups.

Let us now explain each of these steps in more detail. We replace the $O(1)$ vs $\Omega(n)$ separation based on Fourier sampling with a similar separation based on a more general problem called *oracle identification*. In the oracle identification problem, we are given access to an oracle $\mathcal{O}_a : X \to \{0,1\}$ where $a \in A$, for some sets A and X with $\log|A|, \log|X| = \Theta(n)$. Our goal is to determine the identity of a. Further, assume that we have access to a testing oracle $T_a : A \to \{0,1\}$ defined by $T_a(a') = \delta_{a,a'}$, that will let us confirm that we have the right answer.[1]

A quantum algorithm for identifying a can be described as follows: first prepare a state $|\varphi_a\rangle$ using q queries to $\mathcal{O}_a$, then perform a POVM $\{\Pi_{a'}\}_{a' \in A}$ (with $\sum_{a'} \Pi_{a'} \leq I$ to allow for the possibility of a "failure" outcome), using no further queries to $\mathcal{O}_a$. The success probability is $\langle\varphi_a|\Pi_a|\varphi_a\rangle$. For our purposes, it will suffice to place a $\Omega(1)$ lower bound on this probability: say that for each a, $\langle\varphi_a|\Pi_a|\varphi_a\rangle \geq \delta$ for some constant $\delta > 0$. On the other hand, any classical algorithm trivially requires $\geq \log(|A|\delta) = \Omega(n)$ oracle calls to identify a with success probability $\geq \delta$. This is because each query returns only one bit of information. In Theorem 9 we will describe how a large class of quantum circuits can achieve this $O(1)$ vs. $\Omega(n)$ separation, and in Theorems 11 and 12 we will show specifically that QFTs and most random circuits fall within this class.

Now we describe the amplification step. This is a variant of the [BV97] procedure in which making an oracle call in the original problem requires solving a sub-problem from the same family as the original problem. Iterating this ℓ times turns query complexity q into $q^{\Theta(\ell)}$, so choosing $\ell = \Theta(\log n)$ will yield the desired polynomial vs.

[1] This will later allow us to turn two-sided into one-sided error; unfortunately it also means that a non-deterministic Turing machine can find a with a single query to T_a. Thus, while the oracle defined in BV is a candidate for placing BQP outside PH, ours will not be able to place BQP outside of NP. This limitation appears not to be fundamental, but we will leave the problem of circumventing it to future work.

super-polynomial separation. We will generalize this construction by defining an amplified version of oracle identification called *recursive oracle identification*. This is described in the next section, where we will see how it gives rise to superpolynomial speedups from a broad class of circuits.

We conclude that quantum speedups—even superpolynomial speedups—are much more common than the conventional wisdom would suggest. Moreover, as useful as the QFT has been to quantum algorithms, it is far from the only source of quantum algorithmic advantage.

3 Recursive Amplification

In this section we show that once we are given a constant versus linear separation (for quantum versus classical oracle identification), we are able to amplify this to a superpolynomial speedup. We require a much looser definition than in [BV97] because the constant case can have a large error.

Definition 1. *For sets A, X, let $f : A \times X \to \{0,1\}$ be a function. To set the scale of the problem, let $|X| = 2^n$ and $|A| = 2^{\Omega(n)}$. Define the set of oracles $\{\mathcal{O}_a : a \in A\}$ by $\mathcal{O}_a(x) = f(a,x)$, and the states $|\varphi_a\rangle = \frac{1}{\sqrt{|X|}} \sum_{x \in X} (-1)^{f(a,x)} |x\rangle$. The single-level oracle identification problem is defined to be the task of determining a given access to $\mathcal{O}_a$. Let U be a family of quantum circuits, implicitly depending on n. We say that U solves the single-level oracle identification problem if*

$$|\langle a|U|\varphi_a\rangle|^2 \geq \Omega(1)$$

for all sufficiently large n and all $a \in A$. In this case, we define the POVM $\{\Pi_a\}_{a \in A}$ by $\Pi_a = U^\dagger |a\rangle\langle a| U$.

When this occurs, it means that a can be identified from $\mathcal{O}_a$ with $\Omega(1)$ success probability and using a single query. In the next section, we will show how a broad class of unitaries U (the so-called *dispersing* unitaries) allow us to construct f for which U solves the single-level oracle identification problem. There are natural generalizations to oracle identification problems requiring many queries, but we will not explore them here.

Theorem 2. *Suppose we are given a single-level oracle problem with function f and unitary U running in time $\mathrm{poly}(n)$. Then we can construct a modified oracle problem from f which can be solved by a quantum computer in polynomial time (and queries), but requires $n^{\Omega(\log n)}$ queries for any classical algorithm that succeeds with probability $\frac{1}{2} + n^{-o(\log n)}$.*

We start by defining the modified version of the problem (Definition 3 below), and describing a quantum algorithm to solve it. Then in Theorem 4 we will show that the quantum algorithm solves the problem correctly in polynomial time, and in Theorem 6, we will show that randomized classical algorithms require superpolynomial time to have a nonnegligible probability of success.

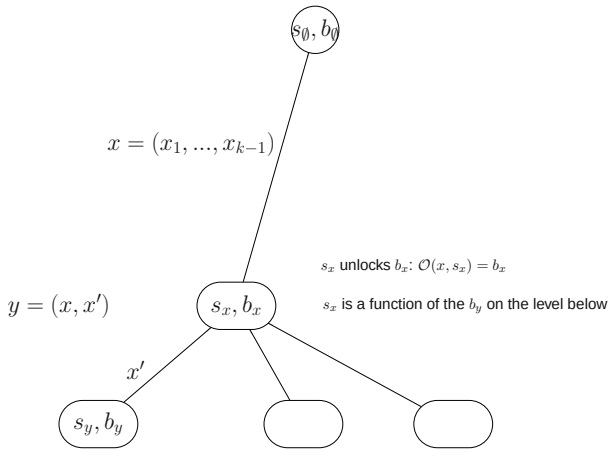

Fig. 1. A depth k node at location $x = (x_1, \ldots, x_k)$ is labeled by its secret s_x and a bit b_x. The secret s_x can be computed from the bits b_y of its children, and once it is known, the bit b_x is computed from the oracle $\mathcal{O}(x, s_x) = b_x$. If x is a leaf then it has no secret and we simply have $b_x = \mathcal{O}(x)$. The goal is to compute the secret bit $b_\emptyset$ at the root.

The recursive version of the problem simply requires that another instance of the problem be solved in order to access a value at a child. Figure 1 illustrates the structure of the problem.

Using the notation from Figure 1, the relation between a secret s_x, and the bits b_y of its children is given by $b_y = f(s_x, x')$, where f is the function from the single-level oracle identification problem. Thus by computing enough of the bits $b_{y_1}, b_{y_2}, \ldots$ corresponding to children $y_1, y_2, \ldots$, we can solve the single-level oracle identification problem to find s_x. Of course computing the b_y will require finding the secret strings s_y, which requires finding the bits of *their* children and so on, until we reach the bottom layer where queries return answer bits without the need to first produce secret strings.

Definition 3. *A level-ℓ recursive oracle identification problem is specified by X, A and f from a single-level oracle identification problem (Definition 1), any function $s : \emptyset \cup X \cup X \times X \cup \ldots \cup X^{\ell-1} \to A$, and any final answer $b_\emptyset \in \{0, 1\}$. Given these ingredients, an oracle $\mathcal{O}$ is defined which takes inputs in*

$$\bigcup_{k=0}^{\ell-1} \left[X^k \times A\right] \cup X^\ell$$

and to return outputs in $\{0, 1, \text{FAIL}\}$. On inputs $x_1, \ldots, x_k \in X, a \in A$ with $1 \leq k < \ell$, $\mathcal{O}$ returns

$$\mathcal{O}(x_1, \ldots, x_k, a) = f(s(x_1, \ldots, x_{k-1}), x_k) \qquad \text{when } a = s(x_1, \ldots, x_k) \qquad (1)$$

$$\mathcal{O}(x_1, \ldots, x_k, a) = \text{FAIL} \qquad \text{when } a \neq s(x_1, \ldots, x_k). \qquad (2)$$

If $k = 0$, then $\mathcal{O}(s(\emptyset)) = b_\emptyset$ and $\mathcal{O}(a) = \text{FAIL}$ if $a \neq s(\emptyset)$. When $k = \ell$,

$$\mathcal{O}(x_1, \ldots, x_\ell) = f(s(x_1, \ldots, x_{\ell-1}), x_\ell).$$

The recursive oracle identification problem is to determine $b_\emptyset$ given access to $\mathcal{O}$.

Note that the function s gives the values s_x in Figure 1. These values are actually defined in the oracle and can be chosen arbitrarily at each node. Note also that the oracle defined here effectively includes a testing oracle, which can determine whether $a = s(x_1, \ldots, x_k)$ for any $a \in A, x_1, \ldots, x_k \in X$ with one query. (When $x = (x_1, \ldots, x_k)$, we use $s(x_1, \ldots, x_k)$ and s_x interchangeably.) A significant difference between our construction and that of [BV97] is that the values of s at different nodes can be set completely independently in our construction, whereas [BV97] had a complicated consistency requirement.

The algorithm. Now we turn to a quantum algorithm for the recursive oracle identification problem. If a quantum computer can identify a with one-sided[2] error $1 - \delta$ using time T and q queries in the non-recursive problem, then we will show that the recursive version can be solved in time $O((q\frac{\log 1/\delta}{\delta})^\ell T)$. For concreteness, suppose that $|\varphi_a\rangle = \frac{1}{\sqrt{|X|}} \sum_{x \in X} (-1)^{f(a,x)} |x\rangle$, so that $q = 1$; the case when $q > 1$ is an easy, but tedious, generalization. Suppose that our identifying quantum circuit is U, so a can be identified by applying the POVM $\{\Pi_{a'}\}_{a' \in A}$ with $\Pi_{a'} = U^\dagger |a'\rangle\langle a'| U$ to the state $|\varphi_a\rangle$.

The intuitive idea behind our algorithm is as follows: At each level, we find $s(x_1, \ldots, x_k)$ by recursively computing $s(x_1, \ldots, x_{k+1})$ for each x_{k+1} (in superposition) and using this information to create many copies of $|\varphi_{s(x_1,\ldots,x_k)}\rangle$, from which we can extract our answer. However, we need to account for the errors carefully so that they do not blow up as we iterate the recursion. In what follows, we will adopt the convention that Latin letters in kets (e.g. $|a\rangle, |x\rangle, \ldots$) denote computational basis states, while Greek letters (e.g. $|\zeta\rangle, |\varphi\rangle, \ldots$) are general states that are possibly superpositions over many computational basis states. Also, we let the subscript $_{(k)}$ indicate a dependence on $(x_1, \ldots, x_k)$. The recursive oracle identification algorithm is as follows:

Algorithm: FIND
Input: $|x_1, \ldots, x_k\rangle|0\rangle$ for $k < \ell$
Output: $a_{(k)} = s(x_1, \ldots, x_k)$ up to error $\varepsilon = (\delta/8)^2$, where δ is the constant from the oracle. This means $|x_1, \ldots, x_k\rangle \left[\sqrt{1-\varepsilon_{(k)}}|0\rangle|a_{(k)}\rangle|\zeta_{(k)}\rangle + \sqrt{\varepsilon_{(k)}}|1\rangle|\zeta'_{(k)}\rangle\right]$, where $\varepsilon_{(k)} \leq \varepsilon$ and $|\zeta_{(k)}\rangle$ and $|\zeta'_{(k)}\rangle$ are arbitrary. (We can assume this form without loss of generality by absorbing phases into $|\zeta_{(k)}\rangle$ and $|\zeta'_{(k)}\rangle$.)

1. Create the superposition $\frac{1}{\sqrt{|X|}} \sum_{x_{k+1} \in X} |x_{k+1}\rangle$.
2. If $k + 1 < \ell$ then let $a_{(k+1)} = \text{FIND}(x_1, \ldots, x_{k+1})$ (with error $\leq \varepsilon$), otherwise $a_{(k+1)} = \emptyset$.
3. Call the oracle $\mathcal{O}(x_1, \ldots, x_{k+1}, a_{(k+1)})$ to apply the phase $(-1)^{f(s(x_1,\ldots,x_k),x_{k+1})}$ using the key $a_{(k+1)}$.
4. If $k + 1 < \ell$ then call FIND† to (approximately) uncompute $a_{(k+1)}$.
5. We are now left with $|\tilde{\varphi}_{(k)}\rangle$, which is close to $|\varphi_{s(x_1,\ldots,x_k)}\rangle$.
   Repeat steps 1–4 $m = \frac{4}{\delta} \ln \frac{8}{\delta}$ times to obtain $|\tilde{\varphi}_{(k)}\rangle^{\otimes m}$
6. Coherently measure $\{\Pi_a\}$ on each copy and test the results (i.e. apply U, test the result, and apply $U^\dagger$).
7. If any tests pass, copy the correct $a_{(k)}$ to an output register, along with $|0\rangle$ to indicate success.
   Otherwise put a $|1\rangle$ in the output to indicate failure.
8. Let everything else comprise the junk register $|\zeta_{(k)}\rangle$.

Theorem 4. *Calling* FIND *on* $|0\rangle$ *solves the recursive oracle problem in quantum polynomial time.*

[2] One-sided error is a reasonable demand given our access to a testing oracle. Most of these results go through with two-sided error as well, but for notational simplicity, we will not explore them here.

Proof. The proof is by backward induction on k; we assume that the algorithm returns with error $\leq \varepsilon$ for $k+1$ and prove it for k. The initial step when $k = \ell$ is trivial since there is no need to compute $a_{\ell+1}$, and thus no source of error. If $k < \ell$, then assume that correctness of the algorithm has already been proved for $k+1$. Therefore Step 2 leaves the state

$$\frac{1}{\sqrt{|X|}} \sum_{x_{k+1} \in X} |x_{k+1}\rangle \left[\sqrt{1-\varepsilon_{(k+1)}}|0\rangle|a_{(k+1)}\rangle|\zeta_{(k+1)}\rangle + \sqrt{\varepsilon_{(k+1)}}|1\rangle|\zeta'_{(k+1)}\rangle \right].$$

In Step 3, we assume for simplicity that the oracle was called conditional on the success of Step 2. This yields

$$|\psi'_{(k)}\rangle := \frac{1}{\sqrt{|X|}} \sum_{x_{k+1} \in X} |x_{k+1}\rangle \left[(-1)^{f(a_{(k)},x_{k+1})} \sqrt{1-\varepsilon_{(k+1)}}|0\rangle|a_{(k+1)}\rangle|\zeta_{(k+1)}\rangle + \sqrt{\varepsilon_{(k+1)}}|1\rangle|\zeta'_{(k+1)}\rangle \right].$$

Now define the state $|\psi_{(k)}\rangle$ by

$$|\psi_{(k)}\rangle := \frac{1}{\sqrt{|X|}} \sum_{x_{k+1} \in X} (-1)^{f(a_{(k)},x_{k+1})} |x_{k+1}\rangle \left[\sqrt{1-\varepsilon_{(k+1)}}|0\rangle|a_{(k+1)}\rangle|\zeta_{(k+1)}\rangle + \sqrt{\varepsilon_{(k+1)}}|1\rangle|\zeta'_{(k+1)}\rangle \right].$$

Note that

$$\langle \psi'_{(k)}|\psi_{(k)}\rangle = \frac{1}{|X|} \sum_{x_{k+1} \in X} \left(1 - \varepsilon_{(k+1)} + (-1)^{f(a_{(k)},x_{k+1})} \varepsilon_{(k+1)} \right).$$

This quantity is real and always $\geq 1 - 2\varepsilon_{(k+1)} \geq \sqrt{1-4\varepsilon}$ by the induction hypothesis. Let

$$|\phi_{(k)}\rangle := \frac{1}{|X|} \sum_{x_{k+1} \in X} (-1)^{f(a_{(k)},x_{k+1})} |x_{k+1}\rangle|0\rangle.$$

Note that $\text{FIND}^\dagger |x_1, \ldots, x_k, \psi_{(k)}\rangle = |x_1, \ldots, x_k, \phi_{(k)}\rangle$. Thus there exists $\varepsilon_{(k)}$ such that applying $\text{FIND}^\dagger$ to $|x_1, \ldots, x_k\rangle|\psi'_{(k)}\rangle$ yields

$$|x_1, \ldots, x_k\rangle \otimes \left[\sqrt{1-4\varepsilon_{(k)}}|\phi_{(k)}\rangle + \sqrt{4\varepsilon_{(k)}}|\phi'_{(k)}\rangle \right],$$

where $\langle \phi_{(k)}|\phi'_{(k)}\rangle = 0$ and $\varepsilon_{(k)} \leq \varepsilon$.

We now want to analyze the effects of measuring $\{\Pi_a\}$ when we are given the state

$$|\varphi_{(k)}\rangle := \sqrt{1-4\varepsilon_{(k)}}|\phi_{(k)}\rangle + \sqrt{4\varepsilon_{(k)}}|\phi'_{(k)}\rangle$$

instead of $|\phi_{(k)}\rangle$. If we define $\|M\|_1 = \text{tr}\sqrt{M^\dagger M}$ for a matrix M, then $\| \, |\varphi_{(k)}\rangle\langle\varphi_{(k)}| - |\phi_{(k)}\rangle\langle\phi_{(k)}| \, \|_1 = 4\sqrt{\varepsilon_{(k)}}$ [FvdG99]. Thus

$$\langle \varphi_{(k)}|\Pi_{a_{(k)}}|\varphi_{(k)}\rangle \geq \langle \phi_{(k)}|\Pi_{a_{(k)}}|\phi_{(k)}\rangle - 4\sqrt{\varepsilon_{(k)}} \geq \delta - 4\sqrt{\varepsilon_{(k)}} \geq \delta/2.$$

In the last step we have chosen $\varepsilon = (\delta/8)^2$.

Finally, we need to guarantee that with probability $\geq 1-\varepsilon$ at least one of the tests in Step 6 passes. After applying U and the test oracle to $|\varphi_{(k)}\rangle$, we have $\geq \sqrt{\delta/2}$ overlap with a successful test and $\leq \sqrt{1-\delta/2}$ overlap with an unsuccessful test. When we repeat this m times, the amplitude in the subspace corresponding to all tests failing is $\leq (1-\delta/2)^{m/2} \leq e^{-m\delta/4}$. If we choose $m = (2/\delta)\ln(1/\varepsilon) = (4/\delta)\ln(8/\delta)$ then the failure amplitude will be $\leq \sqrt{\varepsilon}$, as desired.

To analyze the time complexity, first note that the run-time is $O(T)$ times the number of queries made by the algorithm, and we have assumed that T is polynomial in n. Suppose the algorithm at level k requires $Q(k)$ queries. Then steps 2 and 4 require $mQ(k+1)$ queries each, steps 3 and 6 require m queries each and together $Q(k) = 2mQ(k+1)+2m$. The base case is $k=\ell$, for which $Q(\ell)=0$, since there are no secret strings to calculate for the leaves. The total number of queries required for the algorithm is then $Q(0) \approx (2m)^{2\ell}$. If we choose $\ell = \log n$ the quantum query complexity will thus be $n^{2\log 2m} = n^{O(1)}$ and the quantum complexity will be polynomial in n compared with the $n^{\Omega(\log n)}$ lower bound.

This concludes the demonstration of the polynomial-time quantum algorithm. Now we turn to the classical $n^{\Omega(\log n)}$ lower bound. Our key technical result is the following lemma:

Lemma 5. *Define the recursive oracle identification problem as above, with a function $f : A \times X \to \{0,1\}$ and a secret $s : \emptyset \cup X \cup X \times X \cup \ldots \cup X^{\ell-1} \mapsto A$ encoded in an oracle $\mathcal{O}$. Fix a deterministic classical algorithm that makes $\leq Q$ queries to $\mathcal{O}$. Then if s and* ANS *are chosen uniformly at random, the probability that* ANS *is output by the algorithm is*

$$\leq \frac{1}{2} + \max\left(\frac{Q}{|A|^{1/3}-Q}, Q\left(\frac{\log|A|}{3}\right)^{-\ell}\right).$$

Using Yao's minimax principle and plugging in $|A| = 2^{\alpha n}$, $\ell = \log n$ and $Q = n^{o(\log n)}$ readily yields.

Theorem 6. *If $\log|A| = n^{\Omega(1)}$ and $\ell = \Omega(\log n)$, then any randomized classical algorithm using $Q = n^{o(\log n)}$ queries will have $\frac{1}{2} + n^{-\Omega(\log n)}$ probability of successfully outputting* ANS.

Proof (of Lemma 5). Let $T = \emptyset \cup X \cup \ldots \cup X^{\ell}$ denote the tree on which the oracle is defined. We say that a node $x \in T$ has been *hit* by the algorithm if position x has been queried by the oracle together with the correct secret, i.e. $\mathcal{O}(s(x), x)$ has been queried. The only way to find to obtain information about ANS is for the algorithm to query $\emptyset$ with the appropriate secret; in other words, to hit $\emptyset$.

For $x, y \in T$ we say that x is an *ancestor* of y, and that y is a *descendant* of x, if $y = x \times z$ for some $z \in T$. If $z \in X$ then we say that y is a *child* of x and that x is a *parent* of y. Now define $S \subset T$ to be the set of all $x \in T$ such that x has been hit but none of x's ancestors have been. Also define a function $d(x)$ to be the depth of a node x; i.e. for all $x \in X^k$, $d(x) = k$. We combine these definitions to declare an invariant

$$Z = \sum_{x \in S}\left(\frac{\log|A|}{3}\right)^{-d(x)}$$

The key properties of Z we need are that:

1. Initially $Z = 0$.
2. If the algorithm is successful then it terminates with $Z = 1$.
3. Only oracle queries change the value of Z.
4. Querying a leaf can add at most $(\log |A|/3)^{-\ell}$ to Z.
5. Querying an internal node (i.e. not a leaf) can add at most $2/(|A|^{1/3} - Q)$ to $\mathbf{E}\, Z$, where $\mathbf{E}$ indicates the expectation over random choices of s.

Combining these facts yields the desired bound.

Properties 1–4 follow directly from the definition (with the inequality in property 4 because it is possible to query a node that has already been hit). To establish property 5, suppose that the algorithm queries node $x \in T$ and that it has previously hit k of x's children. This gives us some partial information about $s(x)$. We can model this information as a partition of A into 2^k disjoint sets $A_1, \ldots, A_{2^k}$ (of which some could be empty). From the k bits returned by the oracle on the k children of x we have successfully queried, we know not only that $s(x) \in A$, but that $s(x) \in A_i$ for some $i \in \{1, \ldots, 2^k\}$.

We will now divide the analysis into two cases. Either $k \leq \frac{1}{3} \log |A|$ or $k > \frac{1}{3} \log |A|$. We will argue that in the former case, $|A_i|$ is likely to be large, and so we are unlikely so successfully guess $s(x)$, while in the latter case even a successful guess will not increase Z. The latter case ($k > \frac{1}{3} \log |A|$) is easier, so we consider it first. In this case, Z only changes if x is hit in this step and neither x nor any of its ancestors have been previously hit. Then even though hitting x will contribute $(\log |A|/3)^{-d(x)}$ to Z, it will also remove the k children from S (as well as any other descendants of x), which will decrease Z by at least $k(\log |A|/3)^{-d(x)-1} > (\log |A|/3)^{-d(x)}$, resulting in a net decrease of Z.

Now suppose that $k \leq \frac{1}{3} \log |A|$. Recall that our information about $s(x)$ can be expressed by the fact that $s(x) \in A_i$ for some $i \in \{1, \ldots, 2^k\}$. Since the values of s were chosen uniformly at random, we have $\Pr(A_i) = |A_i|/|A|$. Say that a set A_i is *bad* if $|A_i| \leq |A|^{2/3}/2^k$. Then for a particular bad set A_i, $\Pr(A_i) \leq |A|^{-1/3} 2^{-k}$. From the union bound, we see that the probability that *any* bad set is chosen is $\leq |A|^{-1/3}$.

Assume then that we have chosen a good set A_i, meaning that conditioned on the values of the children there are $|A_i| \geq |A|^{2/3}/2^k \geq |A|^{1/3}$ possible values of $s(x)$. However, previous failed queries at x may also have ruled out specific possible values of x. There have been at most Q queries at x, so there are $\geq |A|^{1/3} - Q$ possible values of $s(x)$ remaining. (Queries to any other nodes in the graph yield no information on $s(x)$.) Thus the probability of hitting x is $\leq 1/(|A|^{1/3} - Q)$ if we have chosen a good set. We also have a $\leq |A|^{-1/3}$ probability of choosing a bad set, so the total probability of hitting x (in the $k \leq \frac{1}{3} \log |A|$ case) is $\leq |A|^{-1/3} + 1/(|A|^{1/3} - Q) \leq 2/(|A|^{1/3} - Q)$. Finally, hitting x will increase Z by at most one, so the largest possible increase of $\mathbf{E}\, Z$ when querying a non-leaf node is $\leq 2/(|A|^{1/3} - Q)$. This completes the proof of property 5 and thus the Lemma.

4 Dispersing Circuits

In this section we define *dispersing* circuits and show how to construct an oracle problem with a constant versus linear separation from any such circuit. In the next sections

we will show how to find dispersing circuits. Our strategy for finding speedups will be to start with a unitary circuit U which acts on n qubits and has size polynomial in n. We will then try to find an oracle for which U efficiently solves the corresponding oracle identification problem. Next we need to define a state $|\varphi_a\rangle$ that can be prepared with $O(1)$ oracle calls and has $\Omega(1)$ overlap with $U^\dagger|a\rangle$. This is accomplished by letting $|\varphi_a\rangle$ be a state of the form $2^{-n/2}\sum_x \pm|x\rangle$. We can prepare $|\varphi_a\rangle$ with only two oracle calls (or one, depending on the model), but to guarantee that $|\langle a|U|\varphi_a\rangle|$ can be made large, we will need an additional condition on U. For any $a \in A$, $U^\dagger|a\rangle$ should have amplitude that is mostly spread out over the entire computational basis. When this is the case, we say that U is *dispersing*. The precise definition is as follows:

Definition 7. *Let U be a quantum circuit on n qubits. For $0 < \alpha, \beta \leq 1$, we say that U is (α, β)-dispersing if there exists a set $A \subseteq \{0,1\}^n$ with $|A| \geq 2^{\alpha n}$ and*

$$\sum_{x\in\{0,1\}^n} |\langle a|U|x\rangle| \geq \beta 2^{\frac{n}{2}}. \tag{3}$$

for all $a \in A$.

Note that the LHS of (3) can also be interpreted as the L_1 norm of $U^\dagger|a\rangle$.

The speedup in [BV97] uses $U = H^{\otimes n}$, which is (1,1)-dispersing since $\sum_x |\langle a|H^{\otimes n}|x\rangle| = 2^{n/2}$ for all a. Similarly the QFT over the cyclic group is (1,1)-dispersing.[3] Nonabelian QFTs do not necessarily have the same strong dispersing properties, but they satisfy a weaker definition that is still sufficient for a quantum speedup. Suppose that the measurement operator is instead defined as $\Pi_a = U(|a\rangle\langle a| \otimes I)U^\dagger$, where a is a string on m bits and I denotes the identity operator on $n-m$ bits. Then U still permits oracle identification, but our requirements that U be dispersing are now relaxed. Here, we give a definition that is loose enough for our purposes, although further weakening would still be possible.

Definition 8. *Let U be a quantum circuit on n qubits. For $0 < \alpha, \beta \leq 1$ and $0 < m \leq n$, we say that U is (α, β)-pseudo-dispersing if there exists a set $A \subseteq \{0,1\}^m$ with $|A| \geq 2^{\alpha n}$ such that for all $a \in A$ there exists a unit vector $|\psi\rangle \in \mathbb{C}^{2^{n-m}}$ such that*

$$\sum_{x\in\{0,1\}^n} |\langle a|\langle\psi|U|x\rangle| \geq \beta 2^{\frac{n}{2}}. \tag{4}$$

This is a weaker property than being dispersing, meaning that any (α, β)-dispersing circuit is also (α, β)-pseudo-dispersing.

We can now state our basic constant vs. linear query separation.

Theorem 9. *If U is (α, β)-pseudo-dispersing, then there exists an oracle problem which can be solved with one query, one use of U and success probability $(2\beta/\pi)^2$. However, any classical randomized algorithm that succeeds with probability $\geq \delta$ must use $\geq \alpha n + \log \delta$ queries.*

[3] Another possible way to generalize [BV97] is to consider other unitaries of the form $U = A^{\otimes n}$, for $A \in \mathcal{U}_2$. However, it is not hard to show that the only way for such a U to be $(\Omega(1), \Omega(1))$-dispersing is for A to be of the form $e^{i\phi_1\sigma_z}He^{i\phi_2\sigma_z}$.

Before we prove this Theorem, we state a Lemma about how well states of the form $2^{-n/2}\sum_x e^{i\phi_x}|x\rangle$ can be approximated by states of the form $2^{-n/2}\sum_x \pm|x\rangle$.

Lemma 10. *For any vector $(x_1,\ldots,x_d)\in\mathbb{C}^d$ there exists $(\theta_1,\ldots,\theta_d)\in\{\pm1\}^d$ such that*

$$\left|\sum_{k=1}^{d} x_k\theta_k\right| \geq \frac{2}{\pi}\sum_{k=1}^{d}|x_k|\,.$$

The proof is in the full version of the paper[HH08].

Proof of Theorem 9: Since U is (α,β)-pseudo-dispersing, there exists a set $A\subset\{0,1\}^m$ with $|A|\geq 2^{\alpha n}$ and satisfying (4) for each $a\in A$. The problem will be to determine a by querying an oracle $\mathcal{O}_a(x)$. No matter how we define the oracle, as long as it returns only one bit per call any classical randomized algorithm making q queries can have success probability no greater than $2^{q-\alpha n}$ (or else guessing could succeed with probability $>2^{-\alpha n}$ without making any queries). This implies the classical lower bound.

Given $a\in A$, to define the oracle $\mathcal{O}_a$, first use the definition to choose a state $|\psi\rangle$ satisfying (4). Then by Lemma 10 (below), choose a vector $\boldsymbol{\theta}$ that (when normalized to $|\theta\rangle$) will approximate the state $U^\dagger|a\rangle|\psi\rangle$. Define $\mathcal{O}_a(x)$ so that $(-1)^{\mathcal{O}_a(x)}=\theta_x=2^{n/2}\langle x|\theta\rangle$. By construction,

$$2^{-n/2}|\langle a|\langle\psi|U|\theta\rangle| \geq \frac{2}{\pi}\beta \tag{5}$$

which implies that creating $|\theta\rangle$, applying U, and measuring the first register has probability $\geq(2\beta/\pi)^2$ of yielding the correct answer a. □

5 Any Quantum Fourier Transform Is Pseudo-dispersing

In this section we start with some special cases of dispersing circuits by showing that any Fourier transform is dispersing. In the next section we show that most circuits are dispersing.

The original RFS paper [BV97] used the fact that $H^{\otimes n}$ is (1,1)-dispersing to obtain their starting $O(1)$ vs $\Omega(n)$ separation. The QFT on the cyclic group (or any abelian group, in fact) is also (1,1)-dispersing. In fact, if we will accept a pseudo-dispersing circuit, then any QFT will work:

Theorem 11. *Let G be a group with irreps $\hat{G}$ and d_λ denoting the dimension of irrep λ. Then the Fourier transform over G is $(\alpha,1/\sqrt{2})$-pseudo-dispersing, where $\alpha=(\log\sum_\lambda d_\lambda)/\log|G|\geq 1/2$.*

Via Theorem 9 and Theorem 2, this implies that any QFT can be used to obtain a superpolynomial quantum speedup. For most nonabelian QFTs, this is the first example of a problem which they can solve more quickly than a classical computer.

Proof (Proof of Theorem 11). Let $A=\{(\lambda,i):\lambda\in\hat{G},i\in\{1,\ldots,d_\lambda\}\}$.

Let V_λ denote the representation space corresponding to an irrep $\lambda \in \hat{G}$. The Fourier transform on G maps vectors in $\mathbb{C}[G]$ to superpositions of vectors of the form $|\lambda\rangle|v_1\rangle|v_2\rangle$ for $|v_1\rangle, |v_2\rangle \in V_\lambda$.

Fix a particular choice of λ and $|i\rangle \in V_\lambda$. If U denotes the QFT on G then let

$$\rho = U^\dagger \left(|\lambda\rangle\langle\lambda| \otimes |i\rangle\langle i| \otimes \frac{I_{V_\lambda}}{d_\lambda} \right) U.$$

Define $V := \operatorname{supp} \rho$, and let $\mathbf{E}_{|\psi\rangle \in V}$ denote an expectation over $|\psi\rangle$ chosen uniformly at random from unit vectors in V[4] Finally, let Π be the projector onto V. Note that $\rho = \Pi / d_\lambda = \mathbf{E} |\psi\rangle\langle\psi|$.

Because of the invariance of ρ under right-multiplication by group elements (i.e. $\langle g_1|\rho|g_2\rangle = \langle g_1 h|\rho|g_2 h\rangle$ for all $g_1, g_2, h \in G$), we have for any g that

$$\langle g|\rho|g\rangle = \frac{1}{|G|} \sum_h \langle gh|\rho|gh\rangle = \frac{1}{|G|} \operatorname{tr}(\rho) = \frac{1}{|G|}. \tag{6}$$

Since $\mathbf{E} |\psi\rangle\langle\psi| = \rho$, (6) implies that

$$\mathop{\mathbf{E}}_{|\psi\rangle \in V} |\langle g|\psi\rangle|^2 = \langle g|\rho|g\rangle = \frac{1}{|G|}.$$

Next, we would like to analyze $\mathbf{E} |\langle g|\psi\rangle|^4$.

$$\mathop{\mathbf{E}}_{|\psi\rangle} |\langle g|\psi\rangle|^4 = \mathop{\mathbf{E}}_{|\psi\rangle} \operatorname{tr} \left(|g\rangle\langle g| \otimes |g\rangle\langle g| \right) \cdot \left(|\psi\rangle\langle\psi| \otimes |\psi\rangle\langle\psi| \right) \tag{7}$$

$$= \operatorname{tr} \left(|g\rangle\langle g| \otimes |g\rangle\langle g| \right) \frac{I + \text{SWAP}}{d_\lambda (d_\lambda + 1)} \left(\Pi \otimes \Pi \right) \tag{8}$$

$$\leq \operatorname{tr} \left(|g\rangle\langle g| \otimes |g\rangle\langle g| \right) \cdot (I + \text{SWAP})(\rho \otimes \rho) \tag{9}$$

$$= 2(\langle g|\rho|g\rangle)^2 = \frac{2}{|G|^2} \tag{10}$$

To prove the equality on the second line, we use a standard representation-theoretic trick (cf. section V.B of [PSW06]). First note that $|\psi\rangle^{\otimes 2}$ belongs to the symmetric subspace of $V \otimes V$, which is a $\frac{d_\lambda (d_\lambda + 1)}{2}$-dimensional irrep of $\mathcal{U}_{d_\lambda}$. Since $\mathbf{E}_{|\psi\rangle} |\psi\rangle\langle\psi|^{\otimes 2}$ is invariant under conjugation by $u \otimes u$ for any $u \in \mathcal{U}_{d_\lambda}$, it follows that $\mathbf{E}_{|\psi\rangle} |\psi\rangle\langle\psi|^{\otimes 2}$ is proportional to a projector onto the symmetric subspace of $V^{\otimes 2}$. Finally, $\text{SWAP}\Pi^{\otimes 2}$ has eigenvalue 1 on the symmetric subspace of $V^{\otimes 2}$ and eigenvalue -1 on its orthogonal complement, the antisymmetric subspace of $V^{\otimes 2}$. Thus, $\frac{I + \text{SWAP}}{2} \Pi^{\otimes 2}$ projects onto the symmetric subspace and we conclude that

$$\mathop{\mathbf{E}}_{|\psi\rangle} |\psi\rangle\langle\psi|^{\otimes 2} = \frac{(I + \text{SWAP})(\Pi \otimes \Pi)}{d_\lambda (d_\lambda + 1)}.$$

[4] We can think of $|\psi\rangle$ either as the result of applying a Haar uniform unitary to a fixed unit vector, or by choosing $|\psi'\rangle$ from any rotationally invariant ensemble (e.g. choosing the real and imaginary part of each component to be an i.i.d. Gaussian with mean zero) and setting $|\psi\rangle = |\psi'\rangle / \sqrt{\langle\psi'|\psi'\rangle}$.

Now we note the inequality

$$\mathbf{E}\,|Y| \geq (\mathbf{E}\,Y^2)^{\frac{3}{2}}/(\mathbf{E}\,Y^4)^{\frac{1}{2}}, \tag{11}$$

which holds for any random variable Y and can be proved using Hölder's inequality [Ber97]. Setting $Y = |\langle g|\psi\rangle|$, we can bound $\mathbf{E}_{|\psi\rangle}\,|\langle g|\psi\rangle| \geq 1/\sqrt{2|G|}$. Summing over G, we find

$$\mathop{\mathbf{E}}_{|\psi\rangle} \sum_{g\in G} |\langle g|\psi\rangle| \geq \frac{1}{\sqrt{2}}\sqrt{|G|}.$$

Finally, because this last inequality holds in expectation, it must also hold for at least some choice of $|\psi\rangle$. Thus there exists $|\psi\rangle \in V$ such that

$$\sum_{g\in G} |\langle g|\psi\rangle| \geq \frac{1}{\sqrt{2}}\sqrt{|G|}.$$

Then U satisfies the pseudo-dispersing condition in (4) for the state $|\psi\rangle$ with $\beta = 1/\sqrt{2}$.

This construction works for each $\lambda \in \hat{G}$ and for $|v_1\rangle$ running over any choice of basis of V_λ. Together, this comprises $\sum_{\lambda\in\hat{G}} d_\lambda$ vectors in the set A.

6 Most Circuits Are Dispersing

Our final, and most general, method of constructing dispersing circuits is simply to choose a polynomial-size random circuit. We define a length-t random circuit to consist of performing the following steps t times.

1. Choose two distinct qubits i, j at random from $[n]$.
2. Choose a Haar-distributed random $U \in \mathcal{U}_4$.
3. Apply U to qubits i and j.

A similar model of random circuits was considered in [DOP07]. Our main result about these random circuits is the following Theorem.

Theorem 12. *For any $\alpha, \beta > 0$, there exists a constant C such that if U is a random circuit on n qubits of length $t = Cn^3$ then U is (α, β)-dispersing with probability*

$$\geq 1 - \frac{2\beta^2}{1 - 2^{-n(1-\alpha)}}.$$

Theorem 12 is proved in the extended version of this paper[HH08]. The idea of the proof is to reduce the evolution of the fourth moments of the random circuit (i.e. quantities of the form $\mathbf{E}_U \operatorname{tr} UM_1U^\dagger M_2UM_3U^\dagger M_4$) to a classical Markov chain, using the approach of [DOP07]. Then we show that this Markov chain has a gap of $\Omega(1/n^2)$, so that circuits of length $O(n^3)$ have fourth moments nearly identical to those of Haar-uniform unitaries from $\mathcal{U}_{2^n}$. Finally, we use (11), just as we did for quantum Fourier transforms, to show that a large fraction of inputs are likely to be mapped to states with large L_1-norm. This will prove Theorem 12 and show that superpolynomial quantum speedups can be built by plugging almost any circuit into the recursive framework we describe in Section 3.

References

[Aar03] Aaronson, S.: Quantum lower bound for recursive Fourier sampling. Quantum Information and Computation 3(2), 165–174 (2003)

[AJL06] Aharonov, D., Jones, V., Landau, Z.: A polynomial quantum algorithm for approximating the jones polynomial. In: STOC 2006: Proceedings of the thirty-eighth annual ACM symposium on Theory of computing, pp. 427–436. ACM Press, New York (2006)

[BCvD05] Bacon, D., Childs, A.M., van Dam, W.: From optimal measurement to efficient quantum algorithms for the hidden subgroup problem over semidirect product groups. In: FOCS 2005: 46th Annual IEEE Symposium on Foundations of Computer Science, pp. 469–478 (2005)

[Bea97] Beals, R.: Quantum computation of Fourier transforms over symmetric groups. In: STOC 1997: Proceedings of the Twenty-Ninth Annual ACM Symposium on Theory of Computing, El Paso, Texas, May 4–6, 1997, pp. 48–53. ACM Press, New York (1997)

[Ber97] Berger, B.: The fourth moment method. Siam J. Comp. 26(4), 1188–1207 (1997)

[BV97] Bernstein, E., Vazirani, U.: Quantum complexity theory. Siam J. Comp. 26(5), 1411–1473 (1997)

[DOP07] Dahlsten, O.C.O., Oliveira, R., Plenio, M.B.: Emergence of typical entanglement in two-party random processes. J. Phys. A 40, 8081–8108 (2007)

[FIM$^+$03] Friedl, K., Ivanyos, G., Magniez, F., Santha, M., Sen, P.: Hidden translation and orbit coset in quantum computing. In: STOC 2003: Proceedings of the Thirty-Fifth Annual ACM Symposium on Theory of Computing, San Diego, CA, pp. 1–9. ACM Press, New York (2003)

[FvdG99] Fuchs, C.A., van de Graaf, J.: Cryptographic distinguishability measures for quantum mechanical states. IEEE Trans. Inf. Th. 45(4), 1216–1227 (1999)

[Hal07] Hallgren, S.: Polynomial-time quantum algorithms for Pell's equation and the principal ideal problem. Journal of the ACM 54(1), 1–19 (2007)

[HH08] Hallgren, S., Harrow, A.W.: Superpolynomial speedups based on almost any quantum circuit (2008)

[HMR$^+$06] Hallgren, S., Moore, C., Rötteler, M., Russell, A., Sen, P.: Limitations of quantum coset states for graph isomorphism. In: STOC 2006: Proceedings of the 38th Annual ACM Symposium on Theory of Computing, pp. 604–617. ACM Press, New York (2006)

[IMS01] Ivanyos, G., Magniez, F., Santha, M.: Efficient quantum algorithms for some instances of the non-abelian hidden subgroup problem. In: Proceedings of the Thirteenth Annual ACM Symposium on Parallel Algorithms and Architectures, Heraklion, Crete Island, Greece, pp. 263–270 (2001)

[PSW06] Popescu, S., Short, A.J., Winter, A.: The foundations of statistical mechanics from entanglement: Individual states vs. averages. Nature 2, 754–758 (2006)

[Sho97] Shor, P.W.: Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. Siam J. Comp. 26(5), 1484–1509 (1997)

[Sim97] Simon, D.R.: On the power of quantum computation. Siam J. Comp. 26(5), 1474–1483 (1997)

[vDHI03] van Dam, W., Hallgren, S., Ip, L.: Quantum algorithms for some hidden shift problems. In: SODA 2003: Proceedings of the Fourteenth Annual ACM-SIAM Symposium on Discrete Algorithms, Baltimore, MD (2003)

[Wat01] Watrous, J.: Quantum algorithms for solvable groups. In: STOC 2001: Proceedings of the Thirty-Third Annual ACM Symposium on Theory of Computing, Crete, Greece, pp. 60–67. ACM Press, New York (2001)

A.3 PRL Paper

Efficient Quantum Circuits for Schur and Clebsch-Gordan Transforms

Dave Bacon
Department of Computer Science and Engineering, University of Washington, Seattle, Washington 98195, USA
Institute for Quantum Information, California Institute of Technology, Pasadena, California 91125, USA
Santa Fe Institute, Santa Fe, New Mexico 87501, USA

Isaac L. Chuang
Center for Bits and Atoms, Research Laboratory for Electronics, Department of Physics, Massachusetts Institute of Technology, Cambridge, Massachusetts 02139, USA

Aram W. Harrow
Department of Physics, Massachusetts Institute of Technology, Cambridge, Massachusetts 02139, USA
Department of Computer Science, University of Bristol, Bristol, BS8 1UB, United Kingdom
(Received 13 July 2004; revised manuscript received 3 March 2006; published 27 October 2006)

The Schur basis on n d-dimensional quantum systems is a generalization of the total angular momentum basis that is useful for exploiting symmetry under permutations or collective unitary rotations. We present efficient {size poly$[n, d, \log(1/\epsilon)]$ for accuracy ϵ} quantum circuits for the *Schur transform*, which is the change of basis between the computational and the Schur bases. Our circuits provide explicit efficient methods for solving such diverse problems as estimating the spectrum of a density operator, quantum hypothesis testing, and communicating without a shared reference frame. We thus render tractable a large series of methods for extracting resources from quantum systems and for numerous quantum information protocols.

DOI: 10.1103/PhysRevLett.97.170502 PACS numbers: 03.67.Lx, 03.67.Mn

A key component of quantum algorithms is their ability to reveal information stored in nonlocal degrees of freedom. In particular, one of the most important building blocks known is the quantum Fourier transform (QFT) [1], an efficient circuit construction for conversion between discrete position and momentum bases. The QFT converts a vector of 2^n amplitudes in $O(n^2)$ steps, in contrast to the $O(n2^n)$ steps required classically.

Another elementary basis change important in quantum physics is between independent local states and those of definite total generalized angular momentum. When two identical spin-1/2 particles interact with a global excitation, due to their permutation symmetry they appear as a singlet or a triplet to the external interaction. When this basis is generalized to n d-dimensional systems (n ''qudits''), we call it the *Schur basis* and call the unitary transformation between local and Schur bases the *Schur transform*.

The Schur transform is central to a plethora of quantum information protocols and to many optimal physical methods for extracting information or resources from a quantum system. These include methods to estimate the spectrum of a density operator [2], perform quantum hypothesis testing [3], perform universal quantum source coding [4], concentrate entanglement noiselessly [5], create states immune to collective decoherence [6], and communicate without a shared reference frame [7]. For all of these tasks (and others), inefficient protocols also exist that work in local bases; however, only the protocols using the Schur basis are optimal. This suggests that the Schur basis is a natural way to treat quantum states based on independent and identically distributed random variables, i.e., to experiments in which many copies of a single quantum state are given. However, unlike the QFT, no efficient algorithm for the Schur transform has been found, rendering protocols which use it nonconstructive. If we wish to implement the Schur transform in the lab to solve any of the problems listed above, an explicit efficient circuit construction for the Schur transform is needed.

Here, we resolve this problem by giving an efficient construction of the Schur transform on n qudits, for arbitrary n and d. This is achieved using a quantum circuit of size poly$[n, d, \log(1/\epsilon)]$ for accuracy ϵ. We believe that this basis change is important not only for quantum information and useful for extracting information about physical systems, but also as a new building block for future quantum algorithms.

The Schur transform.—Consider a system of n qudits, each with a standard local (''computational'') basis $|i\rangle$, $i = 1 \ldots d$. The Schur transform relates transforms on the system performed by local d-dimensional unitary operations to those performed by permutation of the qudits. Recall that the symmetric group $\mathcal{S}_n$ is the group of all permutations of n objects. This group is naturally represented in our system by

$$\mathbf{P}(\pi)|i_1 i_2 \cdots i_n\rangle = |i_{\pi^{-1}(1)} i_{\pi^{-1}(2)} \cdots i_{\pi^{-1}(n)}\rangle, \quad (1)$$

where $\pi \in \mathcal{S}_n$ is a permutation and $|i_1 i_2 \ldots\rangle$ is shorthand for $|i_1\rangle \otimes |i_2\rangle \otimes \ldots$. Let $\mathcal{U}_d$ denote the group of $d \times d$ unitary operators. This group is naturally represented in

our system by

$$\mathbf{Q}(U)|i_1 i_2 \cdots i_n\rangle = U|i_1\rangle \otimes U|i_2\rangle \otimes \cdots \otimes U|i_n\rangle, \quad (2)$$

where $U \in \mathcal{U}_d$.

The Schur transform is based on Schur duality, a well-known [8] and powerful way to relate the representation theory of $\mathbf{P}(\pi)$ and $\mathbf{Q}(U)$. For example, consider the case of two qubits ($n = 2$, $d = 2$). The two-qubit Hilbert space $(\mathbb{C}^2)^{\otimes 2}$ decomposes under $\mathbf{Q}$ into a one-dimensional spin-0 singlet space spanned by $\frac{1}{\sqrt{2}}(|01\rangle - |10\rangle)$ and a three-dimensional spin-1 triplet space spanned by $|00\rangle$, $|11\rangle$, and $\frac{1}{\sqrt{2}}(|01\rangle + |10\rangle)$. Both of these spaces are acted upon in an irreducible manner under the action of $\mathbf{Q}(U)$, $U \in \mathcal{U}_2$, meaning that the action of $\mathbf{Q}(U)$ does not mix these two subspaces and these are the minimal such nonmixing subspaces which exist. Schur duality is related to the fact that these subspaces also happen to be irreducible representations (irreps) of $\mathcal{S}_2$. The singlet state changes sign under permutation of the two spins, and the triplet states are invariant under permutation. These correspond to the sign $\mathcal{P}_{\text{sign}}$ and the trivial $\mathcal{P}_{\text{trivial}}$ irreps of $\mathcal{S}_2$, and thus we can write $(\mathbb{C}^2)^{\otimes 2} \cong (\mathcal{Q}_1 \otimes \mathcal{P}_{\text{trivial}}) \oplus (\mathcal{Q}_0 \otimes \mathcal{P}_{\text{sign}})$, where $\mathcal{Q}_j$ is the spin-j irrep of $\mathcal{U}_2$.

This relation between the two representations exists for an arbitrary number of qudits, and in general both the $\mathcal{U}_d$ and $\mathcal{S}_n$ irreps will be nontrivial. For example, the Hilbert space of three qubits ($n = 3$, $d = 2$) decomposes into $(\mathcal{Q}_{3/2} \otimes \mathcal{P}_{\text{trivial}}) \oplus (\mathcal{Q}_{1/2} \otimes \mathcal{P}_{2,1})$, where $\mathcal{P}_{2,1}$ denotes a particular two-dimensional mixed symmetry irrep of S_3. In terms of the original (local) basis the $\mathcal{Q}_{1/2} \otimes \mathcal{P}_{2,1}$ space contains two spin-1/2 objects, one spanned by $|110\rangle + \omega|011\rangle + \omega^*|101\rangle$ (suppressing normalization) and $|001\rangle + \omega|100\rangle + \omega^*|010\rangle$, and the other obtained by replacing $\omega = e^{2\pi i/3}$ with ω^*. These two spaces correspond to the two dimensions of $\mathcal{P}_{2,1}$.

The general theorem of Schur duality states that for any (integer) d and n,

$$(\mathbb{C}^d)^{\otimes n} \cong \bigoplus_{\lambda \in \text{Part}[n,d]} \mathcal{Q}_\lambda \otimes \mathcal{P}_\lambda, \quad (3)$$

where λ is chosen from the set of possible partitions of n into $\leq d$ parts, and simultaneously labels the $\mathcal{U}_d$-irrep $\mathcal{Q}_\lambda$ and the $\mathcal{S}_n$-irrep $\mathcal{P}_\lambda$. This goes beyond simultaneously diagonalizing the commuting representations $\mathbf{P}$ and $\mathbf{Q}$ because $\mathcal{P}_\lambda$ depends only on n (through λ) and not d. Schur duality means that there exists a basis for $(\mathbb{C}^d)^{\otimes n}$ with states $|\lambda, q_\lambda, p_\lambda\rangle_{\text{Sch}}$, where λ labels the subspaces $\mathcal{Q}_\lambda \otimes \mathcal{P}_\lambda$ and $|q_\lambda\rangle \in \mathcal{Q}_\lambda$ and $|p_\lambda\rangle \in \mathcal{P}_\lambda$ label bases for $\mathcal{Q}_\lambda$ and $\mathcal{P}_\lambda$, respectively.

Just as in the examples above, the Schur basis states $|\lambda, q_\lambda, p_\lambda\rangle_{\text{Sch}}$ are superpositions of the n qudit computational basis states $|i_1 i_2 \ldots i_n\rangle$,

$$|\lambda, q_\lambda, p_\lambda\rangle_{\text{Sch}} = \sum_{i_1,\ldots,i_n} [\mathbf{U}_{\text{Sch}}]^{\lambda,q_\lambda,p_\lambda}_{i_1,i_2,\ldots,i_n} |i_1 i_2 \cdots i_n\rangle. \quad (4)$$

By the isomorphism of Eq. (3), this defines a unitary transformation $\mathbf{U}_{\text{Sch}}$ (with matrix elements as given), the Schur transform we desire. If we think of $\mathbf{U}_{\text{Sch}}$ as a quantum circuit, it maps the state $|\lambda, q_\lambda, p_\lambda\rangle_{\text{Sch}}$ into the computational basis state $|\lambda, q_\lambda, p_\lambda\rangle$, with λ, q_λ, and p_λ expressed as bit strings. Since $\dim(\mathcal{Q}_\lambda)$ and $\dim(\mathcal{P}_\lambda)$ vary with λ we need to pad the $|q\rangle$ and $|p\rangle$ registers; this requires only constant spatial overhead. We know of no efficient classical algorithms to calculate even a single matrix element of $\mathbf{U}_{\text{Sch}}$, the best known results being recursive definitions of these matrix elements which require exponential time to evaluate [9]. The main purpose of this Letter is to show how the entire transformation can be performed on a quantum computer in poly(n, d) steps {implying as a corollary a classical algorithm for Schur transforming a vector of length d^n in time $O[d^n \text{poly}(n, d)]$}.

The defining property of $\mathbf{U}_{\text{Sch}}$ is that it reduces the action of $\mathbf{Q}$ and $\mathbf{P}$ into irreps. For any $\pi \in S_n$ and any $U \in \mathcal{U}_d$, $\mathbf{P}(\pi)$ and $\mathbf{Q}(U)$ commute, so we can express both reductions at once as

$$\mathbf{U}_{\text{Sch}} \mathbf{Q}(U) \mathbf{P}(\pi) \mathbf{U}_{\text{Sch}}^\dagger = \sum_{\lambda \in \text{Part}(d,n)} |\lambda\rangle\langle\lambda| \otimes \mathbf{q}_\lambda(U) \otimes \mathbf{p}_\lambda(\pi), \quad (5)$$

where $\mathbf{q}_\lambda$ and $\mathbf{p}_\lambda$ are irreps of $\mathcal{U}_d$ and $\mathcal{S}_n$, respectively.

Example of the Schur transform.—Consider the case of two qubits ($n = 2$, $d = 2$). Here the Schur transform is the transform between the standard computational basis $|i_1, i_2\rangle$ and a basis describing the singlet and triplet states. Explicitly the matrix of elements for the Schur transform, as in Eq. (4), are given by

$$\begin{array}{l} \\ |\lambda = (1,1), q_\lambda = 0, p_\lambda = 0\rangle_{\text{Sch}} \\ |\lambda = (2,0), q_\lambda = +1, p_\lambda = 0\rangle_{\text{Sch}} \\ |\lambda = (2,0), q_\lambda = 0, p_\lambda = 0\rangle_{\text{Sch}} \\ |\lambda = (2,0), q_\lambda = -1, p_\lambda = 0\rangle_{\text{Sch}} \end{array} \overbrace{\begin{array}{cccc} |00\rangle & |01\rangle & |10\rangle & |11\rangle \\ \hline 0 & \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} & 0 \\ 1 & 0 & 0 & 0 \\ 0 & \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & 0 \\ 0 & 0 & 0 & 1 \end{array}}. \quad (6)$$

Here $\lambda = (1, 1)$ labels the singlet and $\lambda = (2, 0)$ labels the triplet. In this simple case, the permutation irreps are both one dimensional. Further as noted above, when we implement this we must express the label λ, p_λ, q_λ in terms of bit strings from some computational basis. For example, we could label λ by a single qubit and q_λ by two qubits (no qubits are required for p_λ in this example).

Applications of the Schur transform.—The numerous applications of the Schur transform mentioned in the introduction [2–7] solve a variety of problems which are relevant to quantum information theory as well as to experiments designed to acquire information or resources from a quantum system. Applying the Schur transform extracts λ, q, and p values for a given state, allowing the values be manipulated like any other quantum data. Here we briefly review a few of these applications, focusing on the ones most relevant to physics.

One application of the Schur transform is spectrum estimation [2]. In spectrum estimation, we are given access to n copies of a density operator $\boldsymbol{\rho} = \sum_i p_i |i\rangle\langle i|$. Suppose the experimentalist wishes to estimate the values of the eigenvalues p_i but does not know the basis $|i\rangle$. Using the Schur transform is the optimal method for estimating this spectrum for any value of n. In particular, if we are given $\boldsymbol{\rho}^{\otimes n}$, then performing the Schur transform on this state followed by measuring the irrep label λ provides an estimate of the spectrum by taking the partition $\lambda = (\lambda_1, \lambda_2, \ldots, \lambda_d)$ and dividing each λ_i by n: $p_i \approx (\frac{\lambda_1}{n}, \frac{\lambda_2}{n}, \ldots, \frac{\lambda_d}{n})$. In the limit of large n this estimate is optimal [2].

Consider, for example, spectrum estimation for the $n = 2$, $d = 2$ example given above. Let $\boldsymbol{\rho} = p|0\rangle\langle 0| + (1-p)|1\rangle\langle 1|$ be a fixed state and assume the experimentalist does not know the basis $|0\rangle$, $|1\rangle$. In order to estimate this spectrum if we are given two copies of $\boldsymbol{\rho}$, we perform the Schur transform, Eq. (6), on these two qubits and measure the λ register. If we get $\lambda = (2, 0)$ we estimate that the spectrum is that of a pure state $p_1 = 1$, $p_2 = 0$ and if we get $\lambda = (1, 1)$ we estimate that the spectrum is that of the fully mixed state $p_1 = \frac{1}{2}$, $p_2 = \frac{1}{2}$. For a given value of p, the $\lambda = (2, 0)$ case occurs with probability $1 - p(1-p)$, and the $\lambda = (1, 1)$ case occurs with probability $p(1-p)$. Note that only if $p = 0$ or 1 does this estimate exactly reproduce the spectrum. Hence we learn a little about the spectrum with two copies of $\boldsymbol{\rho}$; note that what we have learned is independent of the basis $|0\rangle$, $|1\rangle$. In the limit of a large number of copies, $n \gg 1$, the Schur transform provides the optimal estimate of the spectrum.

Another application of the Schur transform is to encode quantum information into noiseless subsystems which arise due to collective decoherence [6]. Here we run the Schur transform backwards (this can be done for the circuit by applying the inverse of every gate and reversing the order of the gates). If we input into the inverse Schur transform a fixed label $|\lambda\rangle$, some arbitrary information in the $|q_\lambda\rangle$ register, and the information we wish to encode in a noiseless manner into the $|p_\lambda\rangle$ basis, then the n qudit states output from this transform are encoded in a noiseless manner. In particular, the effect of decoherence which couples identically to each of the n qudit states acts trivially on the encoded information. Noiseless subsystems have already been implemented in ion trap quantum computers [10] and our transform makes feasible their use for larger systems.

As an example of the Schur transform in quantum information theory, consider the situation where Alice and Bob share n copies of a partially entangled state $|\psi\rangle_{AB} = \sum_i \sqrt{p_i}\,|i\rangle_A |i\rangle_B$ and they wish to extract the maximal number of maximally entangled states, $\frac{1}{\sqrt{d_E}} \times \sum_{k=1}^{d_E} |i\rangle_A |i\rangle_B$, from these n copies. Alice and Bob's local density matrices are invariant under permutations of their n copies, so if they perform the Schur transform and measure the $|\lambda\rangle$ basis, this leaves their $|p_\lambda\rangle$ registers in a maximally entangled state. If $|\psi\rangle$ is unknown and no classical communication is allowed, then this is an optimal distortion-free entanglement protocol [5]. Note that in order to make this protocol computationally tractable, we need to describe how the $|p_\lambda\rangle$ basis states are labeled in a way that can be efficiently and reversibly mapped to the integers $\{1, \ldots, \dim(\mathcal{P}_\lambda)\}$ [11].

Quantum circuit for the Schur transform.—We construct a quantum circuit [12] for $\mathbf{U}_{\mathrm{Sch}}$ in two stages, first for $d = 2$, then generalizing to $d > 2$. Each of these constructions follows an iterative structure, in which the Schur transform on n qudits is realized using n elementary steps, each of which adds a single qudit to an existing Schur state of the form $|\lambda, q, p\rangle$.

For $d = 2$, this elementary step corresponds to the addition of angular momentum, and the matrix elements of the unitary transform are known as Clebsch-Gordan (CG) coefficients [13]. In this case, λ and q can be conveniently denoted by half integers j and m (with $|m| \le j \le n/2$) which give the total angular momentum and the z-component of angular momentum, respectively. And in terms of j, the CG transform takes as input $|j, m\rangle$ and a single spin $|s = \pm 1/2\rangle$, and outputs a linear combination of the states $|j' = j \pm 1/2, m' = m + s\rangle$. The amplitudes of the linear combination are readily computed using the usual ladder operators for raising and lowering angular momenta [13]. In addition, however, we must distinguish between multiple distinct pathways which add up to give the same total j, as demonstrated by the three qubit example above. In fact, it is the permutation symmetry of these pathways which gives rise to $\mathcal{P}_j$, and thus we track the pathway with another output label $p = j' - j$.

Putting this together, we can define an elementary Clebsch-Gordan transform step $\mathbf{U}_{\mathrm{CG}}$ as a rotation between two specific basis states,

$$\begin{bmatrix} |j'_-, m', \quad p = -\frac{1}{2}\rangle \\ |j'_+, m', \quad p = +\frac{1}{2}\rangle \end{bmatrix} = \begin{bmatrix} \cos\theta_{j,m'} & -\sin\theta_{j,m'} \\ \sin\theta_{j,m'} & \cos\theta_{j,m'} \end{bmatrix} \times \begin{bmatrix} |j, m_+\rangle |s = -\frac{1}{2}\rangle \\ |j, m_-\rangle |s = +\frac{1}{2}\rangle \end{bmatrix}, \quad (7)$$

where $j'_\pm = j \pm 1/2$, $m_\pm = m' \pm 1/2$, and $\cos\theta_{j,m'} = \sqrt{\frac{j+m'+1/2}{2j+1}}$. $\mathbf{U}_{\mathrm{CG}}$ can be realized with three gates in a quantum circuit, as shown in Fig. 1, using as one gate a controlled rotation about $\hat{y}$ by angle $\theta_{j,m'}$. This angle is computed using usual quantum and reversible circuit techniques [12] with error ϵ, using poly$[\log(1/\epsilon)]$ standard circuit elements.

The full Schur transform is implemented by cascading $\mathbf{U}_{\mathrm{CG}}$ as shown in Fig. 2. The complexity of this circuit is thus $\mathcal{O}(n\,\mathrm{poly}\log(1/\epsilon))$. We now claim that $|p_1, \ldots, p_n\rangle := |p\rangle$ labels a basis for $\mathcal{P}_j$. This follows from Eq. (3) and the fact that the $p_k = j_k - j_{k-1}$, $k = 1, \ldots, n$ are invariant under $\mathbf{Q}$, while j and m are invariant under $\mathbf{P}$. In fact, since j_k describes the action of $\mathcal{S}_k$ on the

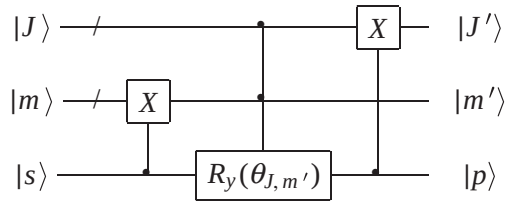

FIG. 1. Quantum circuit implementing $\mathbf{U}_{\mathrm{CG}}$ to convert between the $|j, m\rangle|s\rangle$ and $|j', m', p\rangle$ bases, for the $d = 2$ (qubit) case. Following standard conventions [12], time goes from left to right, the $|j\rangle$ and $|m\rangle$ wires hold multiple qubits, and $|s\rangle$ is one qubit. The controlled X operation $\mathbf{C}_X$ adds the control to the target qubits, i.e., $\mathbf{C}_X|s\rangle|m\rangle = |s\rangle|m + s\rangle$. The doubly controlled $R_y(\theta_{j,m'})$ gate implements the rotation given by Eq. (7) using the j and m' qubits.

first k qubits, $|p_1, \ldots, p_n\rangle$ is a subgroup-adapted basis for the chain $\mathcal{S}_1 \subset \mathcal{S}_2 \subset \ldots \subset \mathcal{S}_n$, also known as Young's orthogonal basis [14]. This basis is also used in the only known fast quantum Fourier transform over $\mathcal{S}_n$ [14,15].

Construction of the Schur transform for $d > 2$ follows the same ideas as for $d = 2$, but is complicated by the challenge of showing that the elementary $\mathbf{U}_{\mathrm{CG}}$ steps for $d > 2$ can be computed in poly(d) steps [a direct construction along the lines of Eq. (7) would require $n^{O(d)}$ steps]. $\mathbf{U}_{\mathrm{Sch}}$ is constructed as a cascade of $O(n)$ $\mathbf{U}_{\mathrm{CG}}$ transforms, just as for $d = 2$. Each $\mathbf{U}_{\mathrm{CG}}$ combines a state $|\lambda, q_\lambda\rangle$ [with $\lambda \in \mathrm{Part}(d, k-1)$ and $|q_\lambda\rangle \in \mathcal{Q}_\lambda$] with a single qudit state $|i_k\rangle$, to obtain a superposition of states $|\lambda', q'_{\lambda'}\rangle$ [with $\lambda' \in \mathrm{Part}(d, k)$ and $|q'_{\lambda'}\rangle \in \mathcal{Q}_{\lambda'}$]. Simultaneously, the permutation labels $|p\rangle$ are constructed; equivalently, we could save the values of λ that we generate in each step, just as $p_1, \ldots, p_n$ are equivalent to $j_1, \ldots, j_n$ for $d = 2$. $\mathbf{U}_{\mathrm{CG}}$ can be computed efficiently because of a recursive relationship between $\mathbf{U}_{\mathrm{CG}}$ for $\mathcal{U}_d \times \mathcal{U}_d$ and that of $\mathcal{U}_{d-1} \times \mathcal{U}_{d-1}$ in terms of reduced Wigner coefficients [16]. Crucially, there is an efficient classical algorithm for the computation of the reduced Wigner coefficients [9] needed for $\mathbf{U}_{\mathrm{CG}}$. Specific details of this calculation are given in detail elsewhere [11]. The complexity of the full Schur transform is thus found to be polynomial in n, d, and $\log(\epsilon^{-1})$.

Conclusion.—We have shown how to efficiently perform the Schur transform. Without efficient implementations of the Schur transform, the various physical and quantum information tasks we have discussed [2–7] are not practical in the lab. As a final note, we comment on the Schur transform as it relates to the search for new quantum algorithms. An important open problem here is to find a black-box problem for which the Schur transform offers a speedup over classical algorithms. In this respect, there are few unitary transforms which have both an efficient quantum circuit and interpretations which might allow these transforms to be useful in an algorithm. We are hopeful that our circuits will be useful for quantum algorithms exactly because they have such clear group representation theory interpretations.

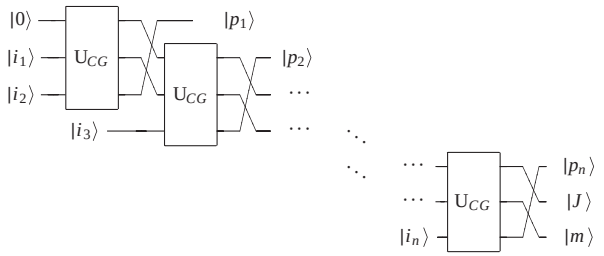

FIG. 2. Quantum circuit for the Schur transformation $\mathbf{U}_{\mathrm{Sch}}$, transforming between $|i_1 i_2 \cdots i_n\rangle$ and $|j, m, p\rangle$. The fact that the $|p_1, p_2, \ldots, p_n\rangle$ is a full basis is intimately related to Schur duality.

This work was partially funded by the NSF Center for Bits and Atoms Contract No. CCR-0122419, the NSF Institute for Quantum Information under Grant No. EIA-0086048, and ARO Contracts No. DAAD19-01-1-06 and No. W911NF-05-R-0009. We thank Nolan Wallach for useful discussions.

[1] P. W. Shor, in *Proceedings of the 37th Symposium on the Foundations of Computer Science* (IEEE, New York, 1996), pp. 56–65.
[2] M. Keyl and R. F. Werner, Phys. Rev. A **64**, 052311 (2001).
[3] M. Hayashi, J. Phys. A **35**, 10 759 (2002).
[4] M. Hayashi and K. Matsumoto, Phys. Rev. A **66**, 022311 (2002); M. Hayashi and K. Matsumoto, Quantum Inf. Comput. **2**, 519 (2002).
[5] M. Hayashi and K. Matsumoto, quant-ph/0209030.
[6] E. Knill, R. Laflamme, and L. Viola, Phys. Rev. Lett. **84**, 2525 (2000); J. Kempe, D. Bacon, D. A. Lidar, and K. B. Whaley, Phys. Rev. A **63**, 042307 (2001).
[7] S. D. Bartlett, T. Rudolph, and R. W. Spekkens, Phys. Rev. Lett. **91**, 027901 (2003).
[8] R. Goodman and N. Wallach, *Representations and Invariants of the Classical Groups* (Cambridge University Press, Cambridge, England, 1998).
[9] L. C. Biedenharn and J. D. Louck, Commun. Math. Phys. **8**, 89 (1968).
[10] D. Kielpinski *et al.*, Science **291**, 1013 (2001).
[11] D. Bacon, I. Chuang, and A. Harrow, quant-ph/0601001.
[12] M. A. Nielsen and I. L. Chuang, *Quantum Computation and Quantum Information* (Cambridge University Press, Cambridge, England, 2000).
[13] J. J. Sakurai, *Modern Quantum Mechanics* (Addison-Wesley, Reading, MA, 1994), Chap. 3, p. 214.
[14] R. Beals, in *Proceedings of the 29th Annual ACM Symposium on Theory of Computing* (ACM Press, New York, 1997), pp. 48–53.
[15] C. Moore, D. Rockmore, and A. Russell, in *Proceedings of the 15th Annual ACM-SIAM Symposium on Discrete Algorithms* (SIAM, Philadelphia, 2004), pp. 778–787.
[16] J. D. Louck, Am. J. Phys. **38**, 3 (1970).

A.4 QIC Paper 1

Quantum Information and Computation, Vol. 8, No. 8&9 (2008) 0715–0721

QUANTUM EXPANDERS FROM ANY CLASSICAL CAYLEY GRAPH EXPANDER

ARAM W. HARROW
Department of Mathematics, University of Bristol, Bristol, U.K.
a.harrow@bris.ac.uk

Received October 10, 2007
Revised February 4, 2008

We give a simple recipe for translating walks on Cayley graphs of a group G into a quantum operation on any irrep of G. Most properties of the classical walk carry over to the quantum operation: degree becomes the number of Kraus operators, the spectral gap lower-bounds the gap of the quantum operation (viewed as a linear map on density matrices), and the quantum operation is efficient whenever the classical walk and the quantum Fourier transform on G are efficient. This means that using classical constant-degree constant-gap families of Cayley expander graphs on groups such as the symmetric group, we can construct efficient families of quantum expanders.

Communicated by: R Cleve & B Terhal

1 Background

Classical expanders can be defined in either combinatorial or spectral terms, while quantum expanders usually have only a spectral definition. Quantum expanders were introduced in [1] for their application to quantum spin chains and in [2] for applications to quantum statistical zero knowledge. Here we (following [1] and [3]) define a (N, D, λ) quantum expander to be a quantum operation $\mathcal{E}$ that

- Has N-dimensional input and output.
- Has $\leq D$ Kraus operators.
- Has second-largest singular value $\leq \lambda$. Equivalently, if $\mathcal{E}(\rho) = \rho$ and $\operatorname{tr} \rho\sigma = 0$ then $\|\mathcal{E}(\sigma)\|_2 \leq \lambda \|\sigma\|_2$, where $\|X\|_2 := \sqrt{\operatorname{tr} X^\dagger X}$.

We say that N is the dimension of the expander, D its degree (by analogy with classical expanders) and $1 - \lambda$ its gap. Note that all quantum operations have at least one fixed state and thus at least one eigenvalue equal to one. The above definition is stricter than the one in [2], which demanded only that an expander increase the von Neumann entropy of a state by at most a constant amount. Finally, we say that an expander is efficient (or "explicit") if it can be implemented on a quantum computer in time $\operatorname{poly}(\log N)$. This paper will describe a new method for constructing quantum expanders, which will in some cases yield efficient $(N, O(1), \Omega(1))$ expanders for all values of $N > 1$.

2 Previous work on efficient quantum expanders

In [4] it was shown that, just as random constant-degree graphs are likely to be expander graphs, quantum operations that apply one of a constant number of random unitaries from $U(N)$ are likely to be quantum expanders, with spectral gap approaching the optimal value as $N \to \infty$. Naturally such expanders cannot be efficiently constructed: generic elements of $U(N)$ require $\Theta(N^2)$ gates to construct[5], and if we want to produce the expander deterministically, the only proposed method[3, Sec. 3.3] does an exhaustive search over $\exp(\Omega(N))$ different unitaries. As there are $\log N$ qubits, this could potentially take time doubly-exponential in the number of qubits.

Prescriptions for potentially efficient constructions are given in [1] and [2]. Both begin with classical expanders and turn them into quantum expanders. The proposal in [1] is to start with a so-called "tensor power expander" and then to add phases. A tensor product expander is a degree D graph (V, E) where: (a) each outgoing edge is labelled $1, \ldots, D$, and (b) if G' is the graph with vertices $V \times V$ and edges given by all pairs $(e_1, e_2) \in E \times E$ such that e_1 and e_2 have the same label, then G' is an expander. Unfortunately, when Cayley graphs (see Section 4 for definition) are labeled in the natural way (with label g corresponding to multiplication by group element g) they are not tensor power expanders. It seems plausible that random constant-degree graphs would be tensor power expanders, but this has not been proven.

The approach of [2] is, like this paper, to turn classical Cayley graph expanders into quantum expanders. Its main idea is to apply a classical expander twice: first in the standard basis, and then conjugated by a sort of generalized Hadamard transform (which they call a "good basis change"), so that it acts in a conjugate basis. Unfortunately, the quantum Fourier transform is not, by itself, always enough to make a good basis change. For some groups, such as $SL(2, q)$, it is, and thus [2] obtain a quantum expander based on the classical LPS expander graph. However, it is unknown how to perform the QFT on $SL(2, q)$ efficiently (see [6] for partial progress), and so we do not know how to efficiently perform the basis change required for their construction. On the other hand, while there are groups such as S_n for which both efficient QFT's and explicit constant-degree expanders are known, none have yet been proved to satisfy the additional property needed for the QFT to be a good basis change.

Very recently, two different constructions of efficient, constant-degree quantum expanders have appeared. The first is described in[3]. Their approach is to generalize the classical zig-zag product[7] to quantum expanders, using a constant number of random unitaries[4] for the base case. Like our paper, [3] also describes a family of constant-degree, constant-gap, efficient expanders. A minor advantage of our construction is that it can be made to work for any dimension $N > 1$, while [3] requires that N be of the form D^{8t} for a positive integer t and that $D > D_0$ for a universal constant D_0.

Another efficient constant-degree expander is given in [8]. Their approach is to turn the classical Margulis expander[9] into an operation on quantum phase space. This results in quantum expanders with the same parameters as the Margulis expander (degree 8, second largest eigenvalue $\lambda \leq 2\sqrt{5}/8$) in any dimension, including even infinite dimensional systems. While their paper only describes an efficient construction for dimensions of the form $N = d^n$ for small d, their approach is easily generalized to run in time $\operatorname{poly} \log N$ for any N.

Finally, if we relax the assumption that expanders have constant degree, then efficient

constructions have been described in [10, 11].

3 Representation theory notation

Let G be a group (either finite or a compact Lie group), and $\hat{G}$ a complete set of inequivalent unitary irreducible representations (irreps). For an irrep $\lambda \in \hat{G}$ and a group element $g \in G$, we denote the representation matrix by $\mathbf{r}_\lambda(g)$, its dimension by d_λ and the space it acts upon by V_λ. Let U_{QFT} be the Fourier transform on G, corresponding to the isomorphism

$$\mathbb{C}[G] \cong \bigoplus_\lambda V_\lambda \otimes V_\lambda^*.$$

It is given by the explicit formula $U_{\mathrm{QFT}} = \sum_{g,\lambda,i,j} \sqrt{d_\lambda/|G|}\mathbf{r}_\lambda(g)_{i,j}|\lambda, i, j\rangle\langle g|$. Let $L_x := \sum_{g\in G} |xg\rangle\langle g|$ denote the left multiplication operator. Then in the Fourier basis, this translates into action on the first tensor factor.

$$U_{\mathrm{QFT}} L_x U_{\mathrm{QFT}}^\dagger = \sum_{\lambda\in\hat{G}} |\lambda\rangle\langle\lambda| \otimes \mathbf{r}_\lambda(x) \otimes I_{d_\lambda}. \tag{1}$$

4 Expander construction

Let G be a group with a generating set $\Gamma \subset G$. Define the Cayley graph $(G;\Gamma)$ to have vertex set G and edges (g, xg) for each $g \in G$ and each $x \in \Gamma$. We will be interested in the case when $(G;\Gamma)$ is an expander graph.

Choose any non-trivial $\lambda \in \hat{G}$. Our quantum expander is defined as follows. Let $\mathcal{E}$ be the quantum operation on V_λ given by

$$\mathcal{E}(\rho) = \frac{1}{|\Gamma|} \sum_{g\in\Gamma} \mathbf{r}_\lambda(g)\rho\,\mathbf{r}_\lambda(g)^\dagger. \tag{2}$$

This operation acts on a d_λ dimensional space by choosing a uniformly random $g \in \Gamma$ and then applying the (unitary) representation matrix $\mathbf{r}_\lambda(g)$. We will see below ways in which $\mathbf{r}_\lambda(g)$ can be implemented on a quantum computer.

I claim that

1. The degree of $\mathcal{E}$ is $\leq |\Gamma|$.
2. If (a) group multiplication in G is efficient, (b) there is a procedure for efficiently sampling from Γ, (c) the QFT on G is efficient and (d) $\log|G| \leq \mathrm{poly}(\log d_\lambda)$, then $\mathcal{E}$ can be implemented efficiently.
3. $$\lambda_2(\mathcal{E}) \leq \lambda_2(W_\Gamma). \tag{3}$$

   Here $\lambda_2(\mathcal{E})$ is the second largest singular value of $\mathcal{E}$, when interpreted as a linear map on density matrices, while $\lambda_2(W_\Gamma)$ is the second-largest singular value of the Cayley graph transition matrix:

   $$W_\Gamma = \frac{1}{|\Gamma|} \sum_{\gamma\in\Gamma} \sum_{g\in G} |\gamma g\rangle\langle g|.$$

Thus, classical Cayley graph expanders give quantum expanders.

Proof of claims (1-3). The first claim is immediate. In the second claim, we use the fact that $\mathbf{r}_\lambda(g)$ can be applied to $|\psi\rangle \in V_\lambda$ by performing the inverse QFT on $|\lambda\rangle|\psi\rangle|0\rangle$, applying the map $|x\rangle \to |gx\rangle$, performing the QFT and keeping only the second register (see [12, Chap. 8] for details). Condition (d) is because we say the QFT on G is efficient if it runs in time $\mathrm{poly}(\log|G|)$, but we would like our expander to run in time $\mathrm{poly}(\log d_\lambda)$. Alternatively (a), (c) and (d) can be replaced by any other efficient procedure for performing $\mathbf{r}_\lambda(g)$ on a quantum computer. (See Section 5 for examples.)

The only non-trivial claim above is (3). Assuming that Γ generates G, the unique stationary state of W_Γ is the uniform distribution

$$|u\rangle := \frac{1}{\sqrt{|G|}} \sum_{g\in G} |g\rangle.$$

We can find the second largest eigenvalue by subtracting off a projector onto the stationary state and taking the operator norm. Thus

$$\lambda_2(W_\Gamma) = \|W_\Gamma - |u\rangle\langle u|\,\|_\infty, \tag{4}$$

where $\|M\|_\infty$ is the largest singular value of M.

Similarly, the maximally mixed state $\tau := I_{d_\lambda}/\sqrt{d_\lambda}$ is a stationary state of $\mathcal{E}$. We choose the normalization so that τ will be a unit vector with respect to the Hilbert-Schmidt inner product $\langle A, B\rangle := \mathrm{tr}\, A^\dagger B$. However, to analyze $\mathcal{E}$ as a linear operator, it is simpler to think of it as acting on vectors. The corresponding linear map is denoted $\hat{\mathcal{E}}$ and is defined to be

$$\hat{\mathcal{E}} := \frac{1}{|\Gamma|} \sum_{\gamma\in\Gamma} \mathbf{r}_\lambda(\gamma) \otimes \mathbf{r}_\lambda(\gamma)^*, \tag{5}$$

where the * denotes the entry-wise complex conjugate with respect to a basis B_λ for V_λ. Then $|\hat{\tau}\rangle := d_\lambda^{-1/2} \sum_{b\in B_\lambda} |b\rangle \otimes |b\rangle$ is a fixed point of $\hat{\mathcal{E}}$. Thus

$$\lambda_2(\mathcal{E}) = \|\hat{\mathcal{E}} - |\hat{\tau}\rangle\langle\hat{\tau}|\,\|_\infty. \tag{6}$$

We now use representation theory to analyze (4) and (6). First, examine (4). Since U_{QFT} is unitary, $\|W_\Gamma - |u\rangle\langle u|\,\|_\infty = \|U_{\mathrm{QFT}} W_\Gamma U_{\mathrm{QFT}}^\dagger - U_{\mathrm{QFT}}\,|u\rangle\langle u|\,U_{\mathrm{QFT}}^\dagger\|_\infty$. Since $U_{\mathrm{QFT}}|u\rangle = |\,\mathrm{trivial}\rangle$, we can use (1) to obtain

$$\lambda_2(W_\Gamma) = \|W_\Gamma - |u\rangle\langle u|\,\|_\infty = \left\| \frac{1}{|\Gamma|} \sum_{\gamma\in\Gamma} \sum_{\lambda\in\hat{G}} |\lambda\rangle\langle\lambda| \otimes \mathbf{r}_\lambda(\gamma) \otimes I_{d_\lambda} - |\mathrm{trivial}\rangle\langle\mathrm{trivial}| \right\|_\infty \tag{7}$$

$$= \left\| \frac{1}{|\Gamma|} \sum_{\gamma\in\Gamma} \sum_{\substack{\lambda\in\hat{G} \\ \lambda\neq\mathrm{trivial}}} |\lambda\rangle\langle\lambda| \otimes \mathbf{r}_\lambda(\gamma) \otimes I_{d_\lambda} \right\|_\infty \tag{8}$$

$$= \max_{\lambda\neq\mathrm{trivial}} \left\| \frac{1}{|\Gamma|} \sum_{\gamma\in\Gamma} \mathbf{r}_\lambda(\gamma) \right\|_\infty \tag{9}$$

A similar argument applies to (6) as well. Here the first step is to decompose $V_\lambda \otimes V_\lambda^*$ into irreps of G. In general,

$$V_\lambda \otimes V_\lambda^* \cong \bigoplus_{\nu \in \hat{G}} V_\nu \otimes \mathbb{C}^{m_\nu},$$

where m_ν is the multiplicity (possibly zero) of V_ν in $V_\lambda \otimes V_\lambda^*$. Let U_{CG} be the unitary transform implementing the above isomorphism. Then by definition,

$$U_{\mathrm{CG}} \left(\mathbf{r}_\lambda(g) \otimes \mathbf{r}_\lambda(g)^*\right) U_{\mathrm{CG}}^\dagger = \sum_{\nu \in \hat{G}} |\nu\rangle\langle\nu| \otimes \mathbf{r}_\nu(g) \otimes I_{m_\nu}. \tag{10}$$

We can use this to analyze the spectrum of $\mathcal{E}$. In particular

$$U_{\mathrm{CG}} \hat{\mathcal{E}} U_{\mathrm{CG}}^\dagger = \sum_{\nu \in \hat{G}} |\nu\rangle\langle\nu| \otimes \left(\frac{1}{|\Gamma|} \sum_{\gamma \in \Gamma} \mathbf{r}_\nu(\gamma)\right) \otimes I_{m_\nu}. \tag{11}$$

From Schur's Lemma, we know that $m_{\mathrm{trivial}} = 1$, corresponding to the stationary state $|\hat{\tau}\rangle$. Thus

$$\begin{aligned}
\lambda_2(\mathcal{E}) &= \|\mathcal{E} - |\hat{\tau}\rangle\langle\hat{\tau}|\,\|_\infty && (12)\\
&= \|U_{\mathrm{CG}}(\mathcal{E} - |\hat{\tau}\rangle\langle\hat{\tau}|)U_{\mathrm{CG}}^\dagger\|_\infty && (13)\\
&= \max_{\substack{m_\nu \neq 0 \\ \nu \neq \mathrm{trivial}}} \left\| \frac{1}{|\Gamma|} \sum_{\gamma \in \Gamma} \mathbf{r}_\nu(\gamma) \right\|_\infty && (14)\\
&\leq \max_{\nu \neq \mathrm{trivial}} \left\| \frac{1}{|\Gamma|} \sum_{\gamma \in \Gamma} \mathbf{r}_\nu(\gamma) \right\|_\infty && (15)\\
&= \lambda_2(W_\Gamma). && (16)
\end{aligned}$$

This completes the proof □.

5 Examples of quantum expanders

If $G = S_n$ then we can use the explicit expander of [13] and the efficient QFT of [14]. The dimension $N = d_\lambda$ can be the size of any irrep of S_n, which asymptotically can be as large as $\sqrt{n!}\exp(-O(\sqrt{n}))$. Run-time is thus poly-logarithmic in the dimension, meaning polynomial in the number of qubits. However if we would like an expander on exactly N dimensions, we are not guaranteed that $n \leq \mathrm{poly}\log(N)$ exists such that $d_\lambda = N$ for some $\lambda \in \hat{S}_n$, nor do we know how to efficiently check, for a given n, whether such a λ exists. (For completeness, we mention here that irreps of S_n are labeled by partitions $(\lambda_1, \ldots, \lambda_n)$ with $\lambda_1 + \ldots + \lambda_n = n$ and $\lambda_1 \geq \ldots \geq \lambda_n \geq 0$. Their dimension is given by $d_\lambda = n! \prod_{i<j}(\lambda_i - \lambda_j - i + j)/\prod_i(\lambda_i + n - i)!$.)

Some other Cayley graph constructions also carry over. For example, the (classical) zig-zag product can be interpreted as a Cayley graph, where the group is an iterated wreath product[15]. Additionally, the irreps of these wreath products are large (although also with possibly inconvenient dimensions) and quantum Fourier transforms on them can be performed efficiently[6]. Thus, classical zig-zag product expanders can also be used to construct efficient,

constant-degree, constant-gap quantum expanders. (We remark in passing that this construction appears not to be related to the quantum zig-zag product of [3].)

If we permit approximate constructions then we can relax the assumption that G is finite. For example, if $G = SU(2)$ then several explicit expanders are known[16, 17], but no efficient circuits are yet known for the QFT. It would suffice even to be able to implement $\mathbf{r}_\lambda(g)$ in time poly-logarithimic in d_λ. This latter result is claimed by [18], but the algorithm there is missing crucial steps.

Finally, to construct expanders for any dimension $N > 1$ we can use the fact that the S_{N+1}-irrep $\lambda = (N, 1)$ has dimension N. To implement $\mathbf{r}_\lambda(\pi)$ for $\pi \in S_{N+1}$ we cannot use the QFT on S_{N+1}, since our run-time needs to be $\operatorname{poly}\log(N)$. However, we can instead embed V_λ into the $N + 1$-dimensional defining representation of S_{N+1}, which is given by $\mathbf{r}_{\mathrm{def}}(\pi)|x\rangle = |\pi(x)\rangle$ for $x = 1, \ldots, N + 1$. This representation is reducible and decomposes into one copy of trivial representation (spanned by $|1\rangle + \ldots + |N + 1\rangle$) and one copy of the N-dimensional irrep $V_{(N,1)}$. To embed V_λ in the defining representation, we can use any $N + 1$-dimensional unitary that maps $|N + 1\rangle$ to $\frac{1}{\sqrt{N+1}} \sum_{x=1}^{N+1} |x\rangle$. Then performing $\mathbf{r}_{\mathrm{def}}(\pi_j)$ (for Cayley graph generator π_j) requires only that $\pi_j(x)$ be computable from j and x in time $\operatorname{poly}(\log N)$. A careful examination of the construction of [13] shows this to be the case. Thus, this technique yields constant-degree, constant-gap explicit expanders for any dimension $N > 1$. (Of course, for low enough values of N the degree will be larger than N^2 and so the resulting expander will be inferior to the trivial "expander" which applies a random generalized Pauli matrix.)

Acknowledgments

I would like to thank Avi Ben-Aroya for useful comments on the first arXiv version of this paper, Matt Hastings for many interesting conversations on this subject, and Cris Moore for pointing out [13] and crucially asking why any classical expander couldn't be turned into a quantum expander. I am also grateful to the Oza family for their kind hospitality while I did most of this work. My funding is from the Army Research Office under grant W9111NF-05-1-0294, the European Commission under Marie Curie grants ASTQIT (FP6-022194) and QAP (IST-2005-15848), and the U.K. Engineering and Physical Science Research Council through "QIP IRC."

References

1. M. B. Hastings, *Entropy and Entanglement in Quantum Ground States*, Phys. Rev. B **76** (2007), 035114, available at arXiv:cond-mat/0701055.
2. A. Ben-Aroya and A. Ta-Shma, *Quantum expanders and the quantum entropy difference problem*, 2007, arXiv:quant-ph/0702129.
3. A. Ben-Aroya, O. Schwartz, and A. Ta-Shma, *An explicit construction of quantum expanders*, 2007, arXiv:0709.0911.
4. M.B. Hastings, *Random unitaries give quantum expanders*, Phys. Rev. A **76** (2007), 032315, available at arXiv:0706.0556.
5. V.V. Shende, S.S. Bullock, and I.L. Markov, *Synthesis of quantum logic circuits*, IEEE Trans. on Computer-Aided Design **25** (2006), no. 6, 1000–1010, available at arXiv:quant-ph/0406176.
6. C. Moore, D. N. Rockmore, and A. Russell, *Generic quantum Fourier transforms*, Proc. 15th SODA, 2004, pp. 778–787, available at arXiv:quant-ph/0304064.

7. O. Reingold, S. Vadhan, and A. Wigderson, *Entropy waves, the zig-zag product and new constant-degree expanders*, Annals of Mathematics **155** (2002), no. 1, 157–187.
8. J. Eisert and D. Gross, *Quantum Margulis Expanders*, Quan. Info. & Comp. **8** (2008), no. 8-9, 0722, available at arXiv:0710.0651.
9. G.A. Margulis, *Explicit constructions of expanders*, Problemy Peredači Informacii **9** (1973), no. 4, 71–80.
10. A. Ambainis and A. Smith, *Small Pseudo-random Families of Matrices: Derandomizing Approximate Quantum Encryption.*, APPROX-RANDOM, 2004, pp. 249–260, available at arXiv: quant-ph/0404075.
11. P. Dickinson and A. Nayak, *Approximate randomization of quantum states with fewer bits of key*, Quantum Computing: Back Action 2006, 2006, pp. 18–36, DOI 10.1063/1.2400876, available at arXiv:quant-ph/0611033.
12. A.W. Harrow, *Applications of coherent classical communication and the Schur transform to quantum information theory*, Ph.D. thesis, Massachusetts Institute of Technology, 2005, arXiv: quant-ph/0512255.
13. M. Kassabov, *Symmetric groups and expanders*, Inventiones Mathematicae **170** (2007), no. 2, available at arXiv:math.GR/0505624.
14. R. Beals, *Quantum computation of Fourier transforms over symmetric groups*, Proc. 29th STOC, 1997, pp. 48–53.
15. E. Rozenman, A. Shalev, and A. Wigderson, *A new family of Cayley expanders*, Proc. 36th STOC, 2004, pp. 445–454.
16. J. Bourgain and A. Gamburd, *New results on expanders*, C. R. Acad. Sci. Paris, Ser. I **342** (2006), 717–721.
17. A. Gamburd, D. Jakobson, and P. Sarnak, *Spectra of elements in the group ring of $SU(2)$*, J. Eur. Math. Soc. **1** (1999), 51–85.
18. C. Zalka, *Implementing high dimensional unitary representations of SU(2) on a Quantum Computer*, 2004, arXiv:quant-ph/0407140.

A.5 QIC Paper 2

Quantum Information and Computation, Vol. 9, No. 3&4 (2009) 0336–0360

CLASSICAL AND QUANTUM TENSOR PRODUCT EXPANDERS

M. B. HASTINGS
Center for Nonlinear Studies and Theoretical Division, Los Alamos National Laboratory
Los Alamos, NM, 87545

A.W. HARROW
Department of Computer Science, University of Bristol
Bristol, U.K.

Received April 8, 2008
Revised December 16, 2008

We introduce the concept of quantum tensor product expanders. These generalize the concept of quantum expanders, which are quantum maps that are efficient randomizers and use only a small number of Kraus operators. Quantum tensor product expanders act on several copies of a given system, where the Kraus operators are tensor products of the Kraus operator on a single system. We begin with the classical case, and show that a classical two-copy expander can be used to produce a quantum expander. We then discuss the quantum case and give applications to the Solovay-Kitaev problem. We give probabilistic constructions in both classical and quantum cases, giving tight bounds on the expectation value of the largest nontrivial eigenvalue in the quantum case.

Keywords: Quantum computing, Unitary transform, Wavelet

Communicated by: R Jozsa & J Watrous

1 Background: classical and quantum expanders

1.1 Definitions

The concept of t-designs[1] provides a way of randomizing quantum states. For example, a 1-design is a set of unitaries $\{U_k\}$, where $k = 1, \ldots, K$, such that the average over the set takes any input state to a maximally mixed state. A 2-design is a set of unitaries such that applying $U_k \otimes U_k$ to a state on a bipartite system generates the twirling operation[2]. Quantum expanders, as studied in Hamiltonian complexity[3], computer science[4], and quantum information theory[5], provide a way of approximately realizing a 1-design by repeatedly applying a completely positive map built out of a small number of unitaries. In this paper, we introduce the concept of "tensor product expanders", which generalize this result and give us a way to approximately realize t-designs. We also discuss the classical case, and show that classical tensor product expanders can be used to generate quantum expanders.

Quantum expanders are a quantum analogue of expander graphs[8]. In the quantum case, we consider a completely positive, trace preserving map

$$\mathcal{E}(M) = \sum_{s=1}^{D} A^\dagger(s) M A(s), \tag{1}$$

where the number of Kraus operators D is relatively small and the map $\mathcal{E}$ has a spectral gap between the largest eigenvalue, equal to unity, and the next largest eigenvalue.[a] We write the spectrum of $\mathcal{E}$ as $\lambda_1, \lambda_2, ...$ with $\lambda_1 = 1$ and $\lambda_2, ...$ all bounded in absolute value by some $\lambda < 1$. We can equivalently consider the operator $\hat{\mathcal{E}} := \sum_{s=1}^{D} A(s) \otimes A(s)^*$.

In this paper we consider the case in which the operators $A^\dagger(s)$ are proportional to unitary operators:

$$A(s) = \frac{1}{\sqrt{D}} U(s). \tag{2}$$

Then the expander map can be implemented by choosing s uniformly at random from $\{1, \ldots, D\}$, and then applying $U(s)$ to the quantum state. The natural generalization of this process, in which we consider k copies of a quantum system, choose a unitary at random, and apply the unitary to all k copies, will be called a k-copy tensor product expander. We will show that these give a way to approximate t-designs for $t = k$.

Random walks on expander graphs can be viewed similarly, as acting on a distribution with a randomly chosen permutation matrix. Consider a directed graph, where each node has D edges leaving it. Label the edges from 1 up to D such that each label appears exactly once among the incoming edges of each vertex and exactly once among the outgoing edges of each vertex. Then, for each edge label s, $1 \leq s \leq D$, define a permutation π_s, where $\pi_s(i) = j$ if a directed edge with label s goes from node i to node j. Then, given a random walk on the graph, the probability distribution $p(i)$ changes in a single step by

$$p(i) \to \frac{1}{D} \sum_{s=1}^{D} \sum_{j=1}^{N} P(s)_{ij} p(j), \tag{3}$$

where $P(s)$ is the permutation matrix corresponding to the permutation π_s; i.e. $P(s)_{ij} = 1$ if $\pi_s(j) = i$ and 0 otherwise.

Hermitian expanders: It is sometimes convenient to guarantee that an expander we construct is Hermitian. To obtain Hermitian $\mathcal{E}$ in the quantum case, we impose

$$U(s + D/2) = U(s)^\dagger. \tag{4}$$

Similarly, in the classical case, we impose

$$\pi_s = \pi_{s+D/2}^{-1} \tag{5}$$

This turns the directed graph into an undirected graph. For notational convenience, we identify $s + D$ with s throughout this paper, so that s is a periodic variable with period D. Note that this constraint (4) requires that D be even. There do exist other ways to construct Hermitian expanders with odd D, if for some s we have $U(s) = U(s)^\dagger$.

1.2 Application to state randomization

For classical expanders, an important implication of the spectral gap is that random walks on an expander graph rapidly approach the stationary distribution. Similarly, quantum expanders can be shown to be rapid mixing. This has application to the problem of *state*

[a] In the non-Hermitian case discussed below, we define the gap instead to be one minus the second-largest singular value of the map $\mathcal{E}$.

randomization, in which classical randomness is used to map a quantum state to an output that is close in trace distance to the maximally mixed state. Ideally the constructions would be [computationally] efficient, meaning they run in time polynomial in the number of qubits, and would use as few random bits as possible.

To make this concrete, suppose that $\mathcal{E}$ is Hermitian and unital with gap $1-\lambda$, and consider a quantum state ρ. We wish to bound the trace norm distance between the maximally mixed state and the state $\mathcal{E}^m(\rho)$ obtained by acting on ρ with some high power, m, of the map $\mathcal{E}$. The calculation exactly follows the classical case. We begin by bounding the ℓ_2 distance. For a matrix A, define $\|A\|_2 = \sqrt{\operatorname{tr} A^\dagger A}$ and $\|A\|_1 = \operatorname{tr}|A| = \operatorname{tr}\sqrt{A^\dagger A}$. Then

$$\left\|\mathcal{E}^m(\rho) - \frac{I}{N}\right\|_2^2 \leq |\lambda|^{2m}, \tag{6}$$

as may be shown by writing ρ as a linear combination of eigenvectors of $\mathcal{E}$, and then by Cauchy-Schwartz,

$$\left\|\mathcal{E}^m(\rho) - \frac{I}{N}\right\|_1 \leq \sqrt{N}|\lambda|^m. \tag{7}$$

Thus, to obtain a given bound on the trace norm distance ϵ, it suffices to take

$$m \geq \log_\lambda(\epsilon/\sqrt{N}). \tag{8}$$

This implies that the set of unitaries, consisting of all unitaries of the form $U(s_1)U(s_2)\cdots U(s_m)$, gives an ϵ-approximate 1-design using

$$K := D^m = \left(\frac{N}{\epsilon^2}\right)^{\frac{1}{2}\log_{1/\lambda}(D)} \tag{9}$$

unitaries.

The exponent $\frac{1}{2}\log(D)/\log(1/\lambda)$ can be thought of as a measure of the efficiency of an expander, meaning the number of bits of randomness it requires to achieve a certain amount of state randomization. Before showing how to evaluate $\frac{1}{2}\log(D)/\log(1/\lambda)$, we review other methods of ℓ_1 state randomization. The simplest is to apply one of N^2 generalized Pauli operators. This can be done efficiently (i.e. in time *poly* $\log(N)$) and perfectly randomizes any state (i.e. $\epsilon = 0$). However, it uses far more randomness than necessary when $\epsilon > 0$. Choosing $K = O(N\epsilon^{-2}\log(1/\epsilon))$ random unitaries was shown to suffice in [10], improving a result of [11] (both of which in fact addressed the more difficult problem of ℓ_∞ state randomization). Similarly an efficient $K = 4N\epsilon^{-2}$ construction was given in [12], which uses less randomness than the efficient constructions of [13] and even than the inefficient constructions based on random unitaries. We note in passing that the constructions in [12, 13] are based on expanders with $\lambda = \epsilon/\sqrt{N}$ and $D = K$.

An expander-based state randomization scheme will be efficient if the underlying expander is efficient and the number of unitaries it uses will be given by (9). Unfortunately $\frac{1}{2}\log(D)/\log(1/\lambda)$ is larger than 2 for all known efficient constant-degree expander constructions[5, 6, 7] (e.g. for the Margulis expander[6], it is ≈ 8.4, and for the zig-zag product[5] it is $2+o(1)$). However, if $U(1),\ldots,U(D/2)$ are chosen at random with $U(s+D/2) = U(s)^\dagger$ then Ref. [19] showed that with high probability $\frac{1}{2}\log(D)/\log(1/\lambda) \approx 1 + \mathcal{O}(\log(N)N^{-1/6}) + 2/\log(D)$, and thus that K is within a small multiplicative factor of N/ϵ^2.

We summarize the above discussion as follows:

Theorem 1 *For any N and any $\epsilon > 0$, consider a set of unitaries $U_1, \ldots, U_K \in \mathcal{U}_N$, which are taken to be strings of unitaries drawn from a set of $D/2$ unitaries $U(1), ..., U(D/2)$ and their conjugates for any $D \geq 4$. Then for most choices of $U(1), ..., U(D/2)$, choose the string length such that*

$$K = \left(\frac{N}{\epsilon^2}\right)^{1+O(N^{-1/6}\log(N))+2/\log(D)} \tag{10}$$

and

$$\left\| \frac{1}{K} \sum_{s=1}^{K} U_s \rho U_s^\dagger - \frac{I}{N} \right\|_1 \leq \epsilon,$$

for all N-dimensional density matrices ρ.

If we take $D \approx 4N/\epsilon^2$ then Theorem 1 can be thought of as tightening the analysis of random unitaries from [10, 11, 12], so that only $(4 + o(1))N/\epsilon^2$ random unitaries are necessary. This shows that Haar-uniform unitaries require almost exactly the same amount of randomness as the construction of [12], although they have the substantial disadvantage of requiring $poly(N)$ time to implement instead of $poly(\log(N))$ time. Since $\lambda \geq (2\sqrt{D-1}/D - O(1/N)) \cdot (1 - O(\log\log(N)/\log(N)))$ for any quantum expander that includes its own inverses [19], one can show that $4N/\epsilon^2$ is the minimum possible values of K for any expander-based randomizing map.

Apart from random unitaries and the large-D constructions of [13, 12], we know of one other class of quantum expanders for which $\frac{1}{2}\log(D)/\log(1/\lambda) \approx 1$. These are obtained by applying the prescription of [7] to the $SU(2)$ expanders described by Lubotsky, Phillips and Sarnak in [14]. Such expanders exist for any N whenever D is odd and $2D-1$ is prime, and satisfy $\lambda = 2\sqrt{D-1}$ exactly. Thus, they provide another $K \approx 4N/\epsilon^2$ method of performing state randomization. However, the only claimed efficient construction of these expanders[15] has an incomplete proof.

In the non-Hermitian case, (6) holds when λ is the second-largest singular value of an expander. If $U(1), \ldots, U(D)$ are chosen uniformly at random, then [19] proved that with high probability the singular values of $\mathcal{E}^m$ for $m = \mathcal{O}(N^{1/6})$ are bounded by $N^2(1/\sqrt{D})^m(1+o(1))$. This implies that the second-largest *eigenvalue* of $\mathcal{E}$ is $\leq \frac{1}{\sqrt{D}}(1 + O(\log(N)N^{-1/6}))$, but does not yield meaningful bounds on the second-largest singular value of $\mathcal{E}$. Indeed, Tobias Osborne has pointed out that when $m = 1$ and $D = 2$, the second largest singular value is equal to unity. If $\mathcal{E}^m$ turned out to have singular values nearly equal to $D^{-m/2}$ then it would imply that $\approx N/\epsilon^2$ random unitaries sufficed to ϵ-randomize a state.

We now turn to tensor product expanders, considering classical tensor product expanders in Section 2 and quantum tensor product expanders in Section 3. The mixing analysis above generalizes in the tensor product case to give approximate t-designs. We will describe randomized constructions of both classical and quantum tensor product expanders. Our basic tool to prove that a random construction gives an expander with high probability is the trace method (see, for example [8, 18]). The basic idea of the trace method is to bound eigenvalues of some linear operator by bounding the trace of high powers of that operator. For example, for a positive definite Hermitian operator whose two largest eigenvalues are equal to unity and to λ, the trace of the m^{th} power is at least equal to $1 + \lambda^m$, so by bounding the trace we bound λ. We focus on high powers of the operator so that the trace will be dominated by the largest eigenvalues. The trace method will be adapted, with slight modifications, to

the various cases, depending on whether classical or classical and quantum, and depending on whether we consider an expander and or a tensor product expander.

2 Classical Tensor Product Expanders

In this section we define classical tensor product expanders, and give a random construction of them. We then show an application of them to constructing quantum expanders.

2.1 Preliminaries, Definitions and Applications

We define an (N, D, λ, k) classical k-copy tensor product expander to be a set of N-by-N permutation matrices $P(s)$, $1 \leq s \leq D$, with the property that the matrix L, defined by

$$L_k = \frac{1}{D} \sum_{s=1}^{D} P(s)^{\otimes k} \tag{11}$$

has some number, f_k^N, eigenvalues equal to unity, with f_k^N defined below, and then all other eigenvalues less than or equal to λ in absolute value. (Again, if L_k is non-Hermitian then we consider its singular values.)

We can obtain Hermitian operators L_k by considering D even, and imposing $P(s+D/2) = P(s)^\dagger$. To obtain Hermitian L_k for D odd, we can instead impose $P(s) = P(s)^\dagger$; that is, the permutation matrices correspond to perfect matchings. Both models corresponds to models of random graphs for $k = 1$ discussed in [9].

These expanders can also be defined by graphs with N^k nodes, labelled $(n_1, n_2, \ldots, n_k)$, where $1 \leq n_i \leq N$. There is an edge from one node $(n_1, \ldots, n_k)$ to another node $(n'_1, \ldots, n'_k)$ if and only if one of the given permutations sends $n_1 \to n'_1, \ldots, n_k \to n'_k$. We refer to this graph as G_k. Alternatively, we can regard $n_1, \ldots, n_k$ as k different random walkers executing a correlated random walk on the original graph.

The function f_k^N is defined to be equal to the number of unit eigenvalues of the operator

$$\frac{1}{N!} \sum_{\pi \in \mathcal{S}_N} P_\pi^{\otimes k} \tag{12}$$

where the sum ranges over *all* permutations π, and P_π is the permutation matrix corresponding to permutation π. Since this operator performs an average over a group action, it is a projector. Applying it to a computational basis state $|n_1, \ldots, n_k\rangle$ maps it to the superposition of all $|n'_1, \ldots, n'_k\rangle$ such that $n'_i = n'_j$ iff $n_i = n_j$. Thus we can represent eigenstates by partitions of $\{1, \ldots, k\}$ into $\leq N$ blocks, such that indices are equal within blocks and unequal across blocks. For example, $f_1^N = 1$, $f_2^N = 2$ (corresponding to the sum of all states with $n_1 = n_2$ and the sum of all states with $n_1 \neq n_2$), $f_3^N = 5$ (corresponding to the possibilities $n_1 = n_2 = n_3$, $n_1 = n_2 \neq n_3$, $n_1 = n_3 \neq n_2$, $n_2 = n_3 \neq n_1$, and $n_1 \neq n_2 \neq n_3 \neq n_1$), and so on. Note that if $N \geq k$ then the constraint that there be $\leq N$ blocks becomes superfluous, and f_k^N becomes simply the k^{th} Bell number B_k, which counts the total number of ways of partitioning a k-element set.

Any matrix L_k of the form (11) is block diagonal with f_k^N different blocks depending on the symmetry of the elements $n_1, \ldots, n_k$ under permutation; we call these subspaces $S_1, S_2, \ldots, S_{f_k^N}$. By the arguments of the above paragraph, we can write the projector in

(12) as

$$\sum_{a=1}^{f_k^n} |u_a\rangle\langle u_a| ,$$

for some unit vectors $|u_a\rangle \in S_a$. These $|u_a\rangle$ are unit eigenvalues not only of (12) but also any L_k.

Rapid mixing: Given the spectral gap, repeatedly applying a classical tensor product expander many times (of order $k \log(N)$) generates an approximately k-wise independent permutation. This means that the results of applying it to k distinct elements are almost indistinguishable from applying a single permutation to each of the k elements. More precisely, given an initial probability distribution, p, in any of the f_k^N different subspaces S_a, we have

$$\|L_k^m p - u_a\|_1 \leq \sqrt{N}^k |\lambda|^m, \tag{13}$$

where u_a is the l_1 normalized eigenvector with eigenvalue unity in this subspace. This approach towards generating k-wise independent permutations has also been considered in [16].

Expanders are not always tensor product expanders. The requirement that a set of permutations form a tensor product expander for $k > 1$ copies is more stringent than the requirement for $k = 1$ copy, as it implies that the correlations between elements are destroyed by the expander. For an example of a classical expander that does not give a tensor product expander, consider any set of D permutation matrices, $P(s)$, on N elements that gives a classical expander. Define a new set of permutation matrices, $P'(s)$, on $2N$ elements, such that $P'(s) = P(s) \oplus P(s)$ for $s = 1, \ldots, D$. Finally, define the permutation $P'(D+1)$ which sends i to $i + N$ if $i \leq N$, and sends i to $i - N$ if $i > N$. Then, these $D + 1$ different permutation matrices define a $k = 1$ expander (they simply correspond to two copies of the original graph, with the possibility of moving between the two copies by using permutation matrix $P'(D+1)$), but does not define a $k = 2$ expander: if two walkers, n_1, n_2 originally are in the same copy as each other, then they remain in the same copy.

Another example comes from Cayley graphs. If G is a group with generators $g_1, \ldots, g_D$ then the Cayley graph on G is defined by taking $N = |G|$ and $P(s)|g\rangle = |g_s g\rangle$ for $s = 1, \ldots, D$. There are many Cayley graph expanders known (c.f. Section 11 of [8]), but applying $P(s) \otimes P(s)$ to any $|g\rangle \otimes |h\rangle$ produces a new state $|\tilde{g}\rangle \otimes |\tilde{h}\rangle$ with $\tilde{g}^{-1}\tilde{h} = g^{-1}h$. Thus, no Cayley graph expander can be a tensor product expander unless it is modified in some way.

The limit of large k: Observe that any k-copy tensor product expander is also a k'-copy tensor product expander for all $k' \leq k$. On the other hand, even if $k > N$ then the k walkers can still occupy only at most N positions. Thus if a map is an N-copy tensor product expander than it is also a k-copy tensor product expander for all k.

An equivalent condition to $\{\pi_1, \ldots, \pi_D\} \subset \mathcal{S}_N$ being an N-tensor product expander is that the Cayley graph generated by $\{\pi_1, \ldots, \pi_D\}$ is an expander. The spectrum of this Cayley graph is identical (up to multiplicity) to that of L_k for all $k \geq N$ (with $P(s)$ defined to be P_{π_s}).

2.2 Random permutations are tensor product expanders

The question then naturally arises whether $k > 1$ tensor product expanders actually exist. Of course there is a trivial $D = N!$ construction where we take $\{\pi_1, \ldots, \pi_N\} = \mathcal{S}_N$ and

achieve $\lambda = 0$ for all k. We would prefer, though, that $D = O(1)$. The construction of [16] nearly achieves this with $D = poly\log(N)$ and $\lambda = 1 - 1/poly(k, \log N)$. For a constant degree construction, we can use Kassabov's expander[17] on $\mathcal{S}_N$. This achieves $D = O(1)$ and λ equal to a constant strictly smaller than 1 for all N and k. Additionally, it can be implemented in time $poly\log(N)$.

In this section, we give a randomized construction of tensor product expanders for any even $D \geq 4$ and with $\lambda \approx \lambda_H^{\frac{1}{k+1}}$, where

$$\lambda_H := \frac{2\sqrt{D-1}}{D}. \tag{14}$$

Theorem 2 *Choose $\pi_1, \ldots, \pi_{D/2} \in \mathcal{S}_N$ at random and then take $\pi_{s+D/2} = \pi_s^{-1}$. Let $P(s) = P_{\pi_s}$. For any k, let λ denote the $f_k^N + 1^{st}$ largest eigenvalue of L_k. Then for any $c > 1$,*

$$\Pr\Big[\lambda \geq c\Big(\lambda_H^{\frac{1}{k+1}} + O(\frac{\log(k) + \log(\log(N))}{\log(N)})\Big)\Big] \leq c^{(-k+1)\log_{1/\lambda_H}(N)}, \tag{15}$$

where Pr[...] denotes probability and λ_H depends on D as given in Eq. (14).

Note that since $\lambda_H^{1/(k+1)}$ converges to unity as k becomes large, the result (15) is only meaningful for $k = \mathcal{O}(\log(N)/\log(\log(N)))$. Constants depending on D are also hidden inside of the $O(...)$ notation. The result is likely far from optimal, since numerical studies indicate that for fixed k and large N, the largest non-trivial eigenvalue λ approaches λ_H. This result for the case $k = 1$ was only recently proven[9]. Our proof, which gives a weaker bound on the expectation value of λ roughly follows the presentation of the trace method in [8, 18], with some modifications.

Proof of Theorem 2: We will apply the trace method separately in each of the subspaces S_a. It suffices to consider only one such subspace S_a, the subspace $S_{f_k^N}$ in which all of the $n_1, n_2, \ldots, n_k$ differ from each other, since every eigenvalue of L_k is an eigenvalue of L_k restricted to $S_{f_k^N}$. For example, consider the case $k = 2$. We have two different subspaces, one with $n_1 = n_2$ and one with $n_1 \neq n_2$. The eigenvectors of the first subspace, of the form $\sum_i p(i)|i\rangle|i\rangle$, correspond to eigenvectors of L_1 of the form $\sum_i p(i)|i\rangle$. Given such an eigenvector, we can construct an eigenvector in the second subspace equal to $\sum_i \sum_{j\neq i} p(i)|i\rangle|j\rangle$ with the same eigenvalue, as claimed.

Let $E[...]$ denote an average over different choices of permutation matrices. Then for any even m,

$$E[|\lambda|] \leq (E[\mathrm{tr}(L_k^m R)] - 1)^{1/m}, \tag{16}$$

where R is the projector onto the given subspace. The expectation value $E[\mathrm{tr}(L_k^m R)]$ equals

$$\Big(\frac{1}{D}\Big)^m \sum_{s_1=1}^{D} \sum_{s_2=1}^{D} ... \sum_{s_m=1}^{D} E[\mathrm{tr}(P(s_1)P(s_2)...P(s_m)R)]. \tag{17}$$

If for some i we have $s_i = s_{i+1} + D/2$, then $P(s_i)P(s_{i+1}) = I$, and we can remove that pair of permutation matrices from the trace above. Similarly, if $s_m = s_1 + D/2$, then we can remove the first and last permutation matrices from the trace, exploiting the cyclic invariance of the trace and the vanishing commutator $[P(s), R] = 0$. We can consider these operations as acting on a word $s_1, s_2, ..., s_m$ on an alphabet $\{1, ..., D\}$. We define a reduced word by

removing pairs of letters of the form $s, s + D/2$. Similarly, if the word ends with a letter s and begins with a letter $s + D/2$, we remove this pair also. We repeat these removals until no further removals are possible. The result is a reduced word of length $m^0 \leq m$; the resulting sequence we write $s'_1, s'_2, ..., s'_{m^0}$. There are at most

$$(D-1)^{m/2} 2^m = D^m \lambda_H^m \tag{18}$$

choices of $s_1, ..., s_m$ which give $m^0 = 0$; the number of these choices is equal to D^m times the return probability of a random walk of length m on a Cayley tree of degree D. For these choices, we have $E[\mathrm{tr}(P(s_1)P(s_2)...P(s_m)R)] = \mathrm{tr}(R) \leq N^k$.

We now consider the other choices of $s_1, ..., s_m$, where $m^0 > 0$. In general,

$$E[\mathrm{tr}(P(s'_1)P(s'_2)...P(s'_{m^0})R)] \leq N^k E[\mathrm{tr}(P(s'_1)P(s'_2)...P(s'_{m^0})R_{1,2,...,k})], \tag{19}$$

where $R_{1,2,...,k}$ projects onto the state with $n_1 = 1, n_2 = 2, ..., n_k = k$. To compute this expectation value, we define $v_0^a = a$, for $1 \leq a \leq k$. Then, define v_i^a, for $i \geq 1$ and $1 \leq a \leq k$, to be $\pi_{s'_i}(v_{i-1}^a)$. Then, the probability that $v_{m^0}^a = a$ for all a is equal to the desired result. We compute this probability as follows. Consider this as happening sequentially, where first we define v_1^a for all a, then we define v_2^a, and so on. We say that a choice of v_i^a is "free" if at no previous step $j < i$ did we compute $\pi_{s'_j}(v_{j-1}^b)$ with $s'_j = s'_i$ and $v_{j-1}^b = v_{i-1}^a$. If a choice of v_i^a is free, and if t values of $\pi_{s'_i}$ have been previously revealed, than we can simply pick v_i^a at random from the $N - t$ possibilities, thus revealing some of the information about the permutation $\pi_{s_i^a}$, and increasing t by one for that permutation. If a choice is not free, then it is "forced", in which case we have no choice about the value of $\pi_{s'_i}(v_{i-1}^a)$.

We say that a coincidence occurs at step i for walker a if this is a free step and the randomly selected vertex coincides with a previously selected vertex (previously selected by *any* of the walkers). Note that for $v_{m^0}^a$ to equal a for all a, we must have at least k coincidences. There are two cases: either there are at least $k + 1$ coincidences, or else there are exactly k coincidences.

The probability of there being at least $k+1$ coincidences can be computed as follows. Let $i_1, i_2, ..., i_{k+1}$ be the steps of the first $k+1$ coincidences and $a_1, a_2, ..., a_{k+1}$ be the corresponding walkers. The probability of having these coincidences for given $i_1, ...$ and $a_1, ...$ is bounded by $(mk/(N-mk))^{k+1}$. Summing over all possible steps and walkers, we find that the probability of having at least $k + 1$ coincidences is bounded by

$$m^{k+1} k^{k+1} (mk/(N - mk))^{k+1}. \tag{20}$$

If there are exactly k coincidences, then each walker has exactly one coincidence given that $v_{m_0}^a = a$ for all a. There are two possibilities: either all of the coincidences occur on the last step, or at least one coincidence does not occur on the last step. The probability of the first case is at most $(1/(N-mk))^k$. If at least one coincidence does not occur on the last step, then let walker b be the first walker to have a coincidence, occurring on step j. Note that each of the vertices $1, ..., a$ must be the randomly selected vertex on exactly one coincidence, again given that $v_{m_0}^a = a$ for all a. Because there are no further coincidences for walker b, we have $s'_i = s'_{i+j}$ for all i. The fraction of reduced words of length m_0 that obey this constraint for given $j \leq m_0/2$ is at most $(D-1)^{-m_0/2}$. The fraction of words that have a reduced

word of length m_0 is at most $(D-1)^{m_0/2}\lambda_H^m$. Therefore, the fraction of words that have a reduced word obeying this constraint, after summing over j, is at most $m\lambda_H^m$. The probability of having these coincidences is bounded by $(m/(N-mk))^k$, where the factor of m arises from the choice of step on which the coincidence occurs (this is in fact a large overestimate). The product of these probabilities is $m\lambda_H^m(m/(N-mk))^k$. The total of these two possibilities is

$$(1/(N-mk))^k + (m/(N-mk))^k m\lambda_H^m). \tag{21}$$

Adding the sum of the expectation value over words with $m^0 = 0$ (which is bounded by $N^k\lambda_H^m$ by Eq. (18) to N^k times the sum of (20,21), we find that

$$\begin{aligned} E[\mathrm{tr}(P(s_1')P(s_2')...P(s_{m^0}')R)] \quad &\leq \quad N^k\lambda_H^m + N^k m^{k+1}k^{k+1}(mk/(N-mk))^{k+1} \\ &\quad +(N/(N-mk))^k + (Nm/(N-mk))^k m\lambda_H^m. \end{aligned} \tag{22}$$

and therefore

$$\begin{aligned} & E[\mathrm{tr}(P(s_1')P(s_2')...P(s_{m^0}')R)] - 1 \\ \leq \quad & N^k\lambda_H^m + N^k m^{k+1}k^{k+1}(mk/(N-mk))^{k+1} \\ & +[(N/(N-mk))^k - 1] + (Nm/(N-mk))^k m\lambda_H^m \\ = \quad & N^k\lambda_H^m + N^k m^{k+1}k^{k+1}(mk/(N-mk))^{k+1} \\ & +\mathcal{O}(mk^2/N) + (Nm/(N-mk))^k m\lambda_H^m. \end{aligned} \tag{23}$$

We pick

$$m = (k+1)\log_{1/\lambda_H}(N) \tag{24}$$

to minimize this expectation value, finding

$$(E[\mathrm{tr}(P(s_1')P(s_2')...P(s_{m^0}')R)] - 1)^{1/m} \leq \lambda_H^{1/(k+1)}(\mathcal{O}(mk))^{(k+1)/m}. \tag{25}$$

Applying Markov's inequality then yields the proof of the Theorem. ∎

2.3 Quantum expanders from classical tensor product expanders

One application of $k = 2$ classical tensor product expanders is to constructing quantum expanders. We give two constructions.

The first approach was introduced, but not formally analyzed, in [3]. Let $P(s)$ be a set of random permutation matrices defining a $k = 2$ tensor product expander, as in the random construction of a $k = 2$ tensor product expander above. Then, define $\sigma(s)$, for $s = 1...D$, to be a diagonal matrix. For $s = 1, ..., D/2$ we choose $\sigma(s)$ to have diagonal entries ± 1 chosen independently at random and we choose $\sigma(s + D/2) = P(s)\sigma(s)P(s)^\dagger$. Then, in [3] it was shown numerically that the A matrices,

$$A(s) = \frac{1}{\sqrt{D}}P(s)\sigma(s), \tag{26}$$

define a quantum expander with high probability. Note that the choice of $\sigma(s + D/2)$ is such that $A(s + D/2) = A(s)^\dagger = (1/\sqrt{D})\sigma(s)P(s)^\dagger$ so that this is a Hermitian expander because $P(s) = P(s+D/2)^\dagger$. Numerically, λ was observed to approach λ_H for large N. We now prove that we do indeed get a quantum expander with high probability, but with a weaker bound on λ_H.

Theorem 3 *Choose $\pi_1, \ldots, \pi_{D/2} \in \mathcal{S}_N$ at random and then take $\pi_{s+D/2} = \pi_s^{-1}$. Let $P(s) = P_{\pi_s}$. Choose $\sigma(s)$ as described above. Let λ denote the second largest eigenvalue of the map with Kraus operators given by the matrices $A(s)$ in Eq. (26). Then, for any $c > 1$,*

$$\Pr\Big[\lambda \geq c\Big(\lambda_H^{\frac{1}{3}} + O(\frac{\log(\log(N))}{\log(N)})\Big)\Big] \leq c^{-3\log_{1/\lambda_H}(N)}. \tag{27}$$

The Hermitian, completely positive map $\mathcal{E}$ defined by the A matrices in (26) sends a diagonal matrix to a diagonal matrix and an off-diagonal matrix to an off-diagonal matrix. So, we consider the spectrum of $\mathcal{E}$ in the diagonal and off-diagonal sectors separately. In the diagonal sector, the spectrum of $\mathcal{E}$ is the same as that of the $k = 1$ expander defined by the given permutation matrices, and hence has a gap between the largest eigenvalue, equal to unity, and the next largest eigenvalue.

The off-diagonal sector requires a little more work. We again use the trace method. Let λ be the largest eigenvalue in absolute value in the off-diagonal sector. Let $M(i,j)$ be an N-by-N dimensional matrix with a one in the i^{th} row and j^{th} column, and zeroes everywhere else, so that these form a basis for the space of N-by-N matrices. The $M(i,j)$ with $i \neq j$ form a basis for the space of off-diagonal matrices. Define (M,N) to be an inner product on the space of N^k-by-N^k dimensional matrices by $(M,N) = \mathrm{tr}(M^\dagger N)$. Then for any even m,

$$E[|\lambda|] \leq \Big(E[\sum_{i\neq j}\Big(M(i,j), \mathcal{E}^m(M(i,j))\Big)]\Big)^{1/m}. \tag{28}$$

Note that compared to Eq. (16), a factor of unity is not subtracted from the expectation value on the right-hand side of Eq. (28).

The evaluation of the right-hand side of Eq. (28) proceeds analogously to that of Eq. (16). The computation in the case $m^0 = 0$ is identical. In the case $m^0 > 0$, we again define coincidences and paths. The only difference is that now rather than just computing the probability that $v^a_{m^0} = a$ for all $a = 1, 2$, the paths come in with signs which may be plus or minus one. This can only reduce the contribution of the terms with $m^0 > 0$. We bound the case with $k+1$ coincidences as before. We also bound the case with k coincidences not all occurring on the last step as before. The only difference is the case in which all coincidences happen on the last step $i = m^0$. The probability of this happening is $(1/N)^2$. The sign, however, is completely random; it is equally likely to be plus or minus one. Thus, the paths with exactly k coincidences, all occurring on step $i = m^0$, contribute zero to the expectation value (28). Thus,

$$E[|\lambda|^m] \leq (N^k + m)\lambda_H^m + N^k m^{k+1} k^{k+1} (mk/(N - mk))^{k+1}. \tag{29}$$

Picking m as before, we find that $E[|\lambda|] \leq \lambda_H^{1/3}(1 + \mathcal{O}(\log(\log(N))/\log(N))$. Applying Markov's inequality yields the theorem. ∎

We now describe our second construction of a quantum expander from a classical tensor product expander.

Theorem 4 *Suppose $\{P(1), \ldots, P(D)\}$ form a $(N, D, 1-\epsilon, 2)$ classical tensor product expander (i.e. $k = 2$). Assume that $N \geq 2$. Let*

$$Z = \sum_{j=1}^{N} |j\rangle\langle j|\, e^{\frac{2\pi i j}{N}}$$

and $p = 1/(1+\epsilon)$. *Define a quantum operation* $\mathcal{E}(M)$ *with* $D+1$ *Kraus operators* $\sqrt{\frac{p}{D}}P(1),\ldots,\sqrt{\frac{p}{D}}P(D),\sqrt{1-p}Z$. *Then* $\mathcal{E}$ *is a* $(N, D+1, 1-\frac{\epsilon}{48})$ *quantum expander.*

Thus, any constant-gap classical 2-TPE can be used to construct a constant-gap quantum expander. No attempt has been made to optimize the constant 48, which we believe can be made arbitrarily close to one when N is large and ϵ is close to 1.

Note that $\sqrt{\frac{p}{D}}P(1),\ldots,\sqrt{\frac{p}{D}}P(D),\sqrt{1-p}Z$ is not in general Hermitian, but if $\{P(1),\ldots,P(D)\}$ is Hermitian then $\{\sqrt{\frac{p}{D}}P(1),\ldots,\sqrt{\frac{p}{D}}P(D),\sqrt{\frac{1-p}{2}}Z,\sqrt{\frac{1-p}{2}}Z^\dagger\}$ is a Hermitian $(N, D+2, 1-\epsilon/48)$ expander; this is proved by using the triangle inequality to relate its gap to the gap of the expander in Theorem 4.

Proof of Theorem 4: The idea is that the classical TPE randomizes the diagonal elements of the density matrix simply because it is an expander, and it randomizes the off-diagonal elements because it is a $k=2$ TPE. Next the phase operation Z adds a phase to the off-diagonal elements so that they are no longer fixed by the classical TPE. Thus the only fixed state will be the identity matrix.

More formally, let $|\varphi_1\rangle = \frac{1}{\sqrt{N}}\sum_{i=1}^N |i\rangle|i\rangle$ and $|\varphi_2\rangle = \frac{1}{\sqrt{N(N-1)}}\sum_{i\neq j}|i\rangle|j\rangle$. These two states form an orthonormal basis for the invariant subspace of $\frac{1}{D}\sum_{s=1}^D P(s)\otimes P(s)$. Thus the fact that $P(1),\ldots,P(D)$ form a 2-TPE implies the bound

$$\left\| \frac{1}{D}\sum_{s=1}^{D} P(s)\otimes P(s) - \varphi_1 - \varphi_2 \right\| \leq \lambda.$$

Next, a short calculation shows that $\langle\varphi_2|Z|\varphi_2\rangle = -1/(N-1)$. Now apply the following Lemma to the subspace orthogonal to $|\varphi_1\rangle$.

Lemma 1 *Let* Π *be a projector and let* X *and* Y *be operators such that* $\|X\|\leq 1$, $\|Y\|\leq 1$, $\Pi X = X\Pi = \Pi$, $\|(I-\Pi)X(I-\Pi)\|\leq 1-\epsilon_X$ *and* $\|\Pi Y\Pi\|\leq 1-\epsilon_Y$. *Assume* $0<\epsilon_X,\epsilon_Y<1$. *Then for any* $0<p<1$, $\|pX+(1-p)Y\|<1$. *Specifically,*

$$\|pX+(1-p)Y\| \leq 1 - \frac{\epsilon_Y}{12}\min(p\epsilon_X, 1-p). \tag{30}$$

Setting $p = 1/(1+\epsilon_X)$, *we obtain*

$$\|pX+(1-p)Y\| \leq 1 - \frac{\epsilon_X\epsilon_Y}{12(1+\epsilon_X)} \leq 1 - \frac{\epsilon_X\epsilon_Y}{24}. \tag{31}$$

The Lemma is proved in Appendix 1. We apply the Lemma by taking $X = \frac{1}{D}\sum_{s=1}^D P(s)\otimes P(s) - \varphi_1$, $Y = Z\otimes Z^* - \varphi_1$ and $\Pi = \varphi_2$. Then plugging $\epsilon_X = \epsilon$ and $\epsilon_Y = 1-1/(N-1)\geq 1/2$ into (31) completes the proof of Theorem 4. ∎

3 Quantum Tensor Product Expanders

In this section we define quantum tensor product expanders and show that random unitaries provide a way of constructing tensor product expanders. We begin with some preliminaries and definitions, present applications to the Solovay-Kitaev problem of approximating unitaries by a string of elementary operations, and finally prove that random unitaries give tensor product expanders. The proof of this last statement begins in subsection 3.3; it closely follows [19] and should be read in conjunction with that paper.

3.1 Preliminaries, Definitions, and Applications

Suppose we have a collection of unitaries $\{U(1), \ldots, U(D)\} \in \mathcal{U}_N$. Define a quantum operation $\mathcal{E}_k$ that applies $U(s)^{\otimes k}$ for $s \in \{1, \ldots, D\}$ chosen uniformly at random. In other words

$$\mathcal{E}_k(M) = \frac{1}{D} \sum_{s=1}^{D} U(s)^{\otimes k} M (U(s)^\dagger)^{\otimes k}, \tag{32}$$

where M is an $N^k \times N^k$ matrix. Since an $N^k \times N^k$ matrix can also be viewed as an N^{2k}-dimensional vector, we can also interpret $\mathcal{E}_k$ as a linear operator on an N^{2k}-dimensional vector space. Define this operator to be

$$\hat{\mathcal{E}}_k := \frac{1}{D} \sum_{s=1}^{D} U(s)^{\otimes k} \otimes (U(s)^*)^{\otimes k}. \tag{33}$$

Note that $\mathcal{E}_k$ and $\hat{\mathcal{E}}_k$ are isospectral.

In previous work[19, 4, 5] $\mathcal{E}_1$ was said to be a (N, D, λ) quantum expander if the second-largest eigenvalue of $\hat{\mathcal{E}}_2$ was $\leq \lambda$. In fact, the definition of quantum expanders included even quantum operations that were not mixtures of unitaries, as long as they could be expressed using $\leq D$ Kraus operators. Here we will change notation from [19, 4, 5] slightly. We say that a set of unitaries $\{U(1), \ldots, U(D)\}$ is a (N, D, λ, k) tensor product expander if the operator $\mathcal{E}_k$ has F_k^N (defined below) eigenvalues equal to one, and all of its other eigenvalues have absolute value $\leq \lambda$. This differs from the notation of [19, 4, 5] in that the set of unitaries, rather than the quantum operation, constitutes the quantum expander[b] When N and D are understood, we sometimes simply say that $\{U(1), \ldots, U(D)\}$ are a k-tensor product expander with gap $1 - \lambda$.

We define F_k^N to be the rank of the projector

$$\hat{\mathcal{T}}_k := \int_{U \in \mathcal{U}_N} U^{\otimes k} \otimes (U^*)^{\otimes k} \mathrm{d}U$$

or equivalently of the operation $\mathcal{T}_k$, which is defined by

$$\mathcal{T}_k(M) = \int_{U \in \mathcal{U}_N} U^{\otimes k} M (U^\dagger)^{\otimes k}. \tag{34}$$

(Throughout the paper the integration measure $\mathrm{d}V$ will be the Haar measure.) This map is the "twirling" operation[2]. Since $\mathcal{T}_k$ is a Hermitian map and $\mathcal{T}_k(\mathcal{T}_k(M)) = \mathcal{T}_k(M)$, the map $\mathcal{T}_k(M)$ has all eigenvalues equal to zero or unity.

For $\pi \in \mathcal{S}_k$, we define the $N^k \times N^k$ matrix $\mathbf{P}_N(\pi)$ is defined to be

$$\mathbf{P}_N(\pi) = \sum_{i_1=1}^{N} \cdots \sum_{i_k=1}^{N} |i_1, \ldots, i_N\rangle\langle i_{\pi(1)}, \ldots, i_{\pi(N)}|.$$

Since $\mathbf{P}_N(\pi)$ commutes with any matrix of the form $U^{\otimes k}$, it follows that $\mathcal{T}_k(\mathbf{P}_N(\pi)) = \mathcal{E}_k(\mathbf{P}_N(\pi)) = \mathbf{P}_N(\pi)$ for any π. We claim that the $\mathbf{P}_N(\pi)$ (and their linear combinations)

[b] One can slightly generalize this by defining a set of unitaries and a set of associated probabilities to be a tensor product expander; however in this paper we consider applying each unitary with equal probability summing to unity.

constitute all of the unit eigenvalues of $\mathcal{E}_k$. This fact follows from Schur-Weyl duality, and specifically Thm 3.3.8 of [24] which states that $\mathcal{T}_k(M) = M$ if and only if M is a linear combination of $\mathbf{P}_N(\pi)$ operators. Thus $F_k^N = \dim Span\{\mathbf{P}_N(\pi) : \pi \in \mathcal{S}_k\}$.

An important special case is when $N \geq k$. In this case, the set $\{\mathbf{P}_N(\pi)|1, 2, \ldots, k\rangle : \pi \in \mathcal{S}_k\}$ is linearly independent, which implies that $\{\mathbf{P}_N(\pi) : \pi \in \mathcal{S}_k\}$ is linearly independent and thus that $F_k^N = k!$.

In the quantum case, tensor product expanders give us a way to approximate the twirling operator $\mathcal{T}_k$ of [2]. This is because

$$\|\mathcal{E}_k^m - \mathcal{T}_k\|_\infty \leq \lambda^m, \tag{35}$$

so whenever $\lambda < 1$, $\mathcal{E}_k^\infty = \mathcal{T}_k$. Let us consider various other possibilities for implementing twirling as a sum of different unitary transformations: one approach to exactly implementing the twirling operation is to use t-designs[1], but the number of unitaries that must be implemented in this case grows with N. Another approach was discussed in [20], which avoids having the number of unitaries grow in N, but requires the ability to implement a number of unitaries growing linearly in the logarithm of the error of the approximation. In contrast, tensor product expander require only the ability to implement a constant number of unitaries to get arbitrarily good approximations. This is a definite advantage; however, in practice, our construction of tensor product expanders here, which relies on the ability to construct random unitary operations, probably cannot be efficiently implemented using gates; instead, we would like to efficiently implement a deterministically constructed tensor product expander. This raises the interesting question of whether the constructions of [5] can lead to tensor product expanders also.

The limit of large k: The situation when k is large has some similarities to the classical case. It still holds that any (N, D, λ, k) quantum tensor product expander is also a (N, D, λ, k') quantum tensor product expander for all $k' \leq k$. In particular, if a set of unitaries forms a (N, D, λ, ∞) quantum tensor product expander than it is also a (N, D, λ, k) quantum tensor product expander for any finite k. This is equivalent to generating a Cayley graph expander on $\mathcal{U}_N$. One difference between the quantum and classical cases is that there is no upper bound to the size of irreps of $\mathcal{U}_N$, like there is for $\mathcal{S}_N$.

Note that constant degree Cayley graph expanders are known for $\mathcal{U}_2$; indeed, choosing the matrices at random will yield an expander with probability one[26]. However, no proof of this fact is known for $N > 2$.

3.2 Solovay-Kitaev gate approximation

One application of tensor product expanders is to the problem of approximating an arbitrary $V \in \mathcal{U}_N$ with a string of gates from a fixed universal set $\{U(1), \ldots, U(D)\}$. The fact that $\{U(1), \ldots, U(D)\}$ is universal means that $\langle U(1), \ldots, U(D)\rangle$ is dense in $\mathcal{U}_N$ (optionally neglecting an overall phase). This means that for any $V \in \mathcal{U}_N$ and any $\epsilon > 0$, there exists a string $s_1, \ldots, s_m$ such that $U(s_1)U(s_2)\cdots U(s_m)$ is within a distance ϵ of V. Often we also want to know (a) how quickly m grows with $1/\epsilon$ and (b) how long it takes to find $s_1, \ldots, s_m$. When $\{U(1), \ldots, U(D)\}$ contain their own inverses, the Solovay-Kitaev theorem[21] gives a $poly\log(1/\epsilon)$ time (for fixed N) algorithm to find an ϵ-approximation with $m = O(\log^{3+o(1)}(1/\epsilon))$. Very little is known in the case without access to inverses, except

that $U(s)^\dagger$ can be simulated to error ϵ using $O(1/\epsilon^{N^2})$ applications of $U(s)$, meaning that the Solovay-Kitaev construction can be used with this amount of overhead.

Turning to lower bounds, observe a ball of radius ϵ in $\mathcal{U}_N$ has volume $\Theta(\epsilon^{N^2})$. This implies that to approximate all strings to within error ϵ requires $\Omega((1/\epsilon)^{N^2})$ different unitaries, or equivalently a $\Omega(N^2 \log 1/\epsilon)$ string length. A long-standing open question is whether the Solovay-Kitaev approximation can in general be improved to use the optimal $O(\log 1/\epsilon)$ number of gates. Such optimally short approximations are known to exist whenever a particular random walk on $\mathcal{U}_N$ has a gap[22]: specifically, the walk consisting of multiplying by $U(s)$ for s randomly chosen from $1, \ldots, D$. For $\mathcal{U}_2$, it was recently proven that generic $U(1), \ldots, U(D)$ are gapped[23] and thus yield short approximating strings. However, the situation for $\mathcal{U}_N$ for $N > 2$ remains open.

In this section we will prove that when k is sufficiently large, unitaries forming k-tensor product expanders yield optimal $O(N^2 \log 1/\epsilon)$-length ϵ-approximations for any gate in $\mathcal{U}_N$.

Theorem 5 . *Suppose $\{U(1), \ldots, U(D)\}$ form a k-tensor product expander with gap $1-\lambda$ for $k \gg \frac{N^3 \log^2(1/\epsilon)}{\epsilon}$. Then for any $V \in \mathcal{U}_N$ there exists a string $s_1, \ldots, s_m \in \{1, \ldots, D\}$ with $m = O(N^2 \log_{1/\lambda}(1/\epsilon))$ and $d(V, U(s_1)U(s_2)\cdots U(s_m)) \leq \epsilon$.*

Here we define the distance between two unitaries $d(U, V)$ by

$$d(U, V) = \min_{\phi \in [0, 2\pi]} \|U - e^{i\phi} V\|_2 = 2N - 2|\operatorname{tr} U^\dagger V|,$$

so that it ignores overall phase.

The main result from [22] can be thought of a weaker version of Theorem 5: it requires $k = \infty$ to achieve the same conclusion. Unfortunately, Theorem 6 only shows that generic sets of unitaries are k-tensor product expanders for $k \sim N^{1/6}/\log(N)$. Thus, at present the existence of expanders satisfying the assumptions of Theorem 5 is a nontrivial conjecture. It is possible that there exists some strengthening of the results of Theorem 6 which will allow us to show that generic unitaries fulfill the assumptions of Theorem 5.

Proof of Theorem 5: Let $|\Phi\rangle = \frac{1}{\sqrt{N}} \sum_{i=1}^N |i\rangle|i\rangle$ be the maximally entangled state on $C^N \otimes C^N$. Define $\rho(U) = \left[(U \otimes I)\Phi(U^\dagger \otimes I)\right]^{\otimes k}$. Observe that

$$\operatorname{tr} \rho(U)\rho(V) = |\operatorname{tr} U^\dagger V|^{2k}/N^{2k} = \left(1 - \frac{d(U, V)}{2N}\right)^{2k} \tag{36}$$

Let $B_{\epsilon/3}$ be the ball of radius $\epsilon/3$ around the identity: $B_{\epsilon/3} = \{U | d(U, I) \leq \epsilon/3\}$. Let $Vol(\epsilon/3)$ denote the volume of $B_{\epsilon/3} = \mathcal{O}((\epsilon/3)^{N^2})$. Define

$$\rho_\epsilon(U) = \frac{1}{Vol(\epsilon/3)} \int_{V \in B_{\epsilon/3}} \rho(VU) \mathrm{d}V, \tag{37}$$

Similarly we define

$$\rho_H = \int_{V \in \mathcal{U}_N} \rho(V) \mathrm{d}V. \tag{38}$$

These states are normalized so that $\operatorname{tr} \rho_\epsilon(U) = \operatorname{tr} \rho_H = 1$. Since $\rho(V) \geq 0$ for all V, we have the operator inequality $\rho_\epsilon(U) \leq \rho_H / Vol(\epsilon/3)$ for any U. Also observe that $\rho_H = (\mathcal{T}_k \otimes id_N^{\otimes k})(\rho(U))$ for any U, where id_N denotes the identity operation on $N \times N$ density matrices.

We will find it convenient to think of density matrices as vectors with the Hilbert-Schmidt inner product $\langle A, B\rangle = \operatorname{tr} A^\dagger B$. In this picture $\mathcal{T}_k$ is a projector, and so

$$\operatorname{tr} \rho_\epsilon(U)\rho_H = \operatorname{tr} \rho_\epsilon(U)(\mathcal{T}_k \otimes id_N^{\otimes k})(\rho_\epsilon(U)) = \operatorname{tr} \rho_H^2.$$

To bound $\operatorname{tr} \rho_H^2$, observe that the support of ρ_H lies within $Span\{|\psi\rangle^{\otimes k} : |\psi\rangle \in C^{N^2}\}$, which (according to [25, 24]) has dimension $N^2 + k - 1N^2 = k(k+1)\cdots(k+N^2-1)/N^2! \leq k^{N^2}$. Thus $\operatorname{tr} \rho_H^2 \geq k^{-N^2}$.

Now we use the fact that $\|\mathcal{E}_k^m - \mathcal{T}_k\|_\infty \leq \lambda^m$ together with Cauchy-Schwartz to bound

$$\operatorname{tr} \rho_\epsilon(I)\mathcal{E}^m(\rho_\epsilon(U)) \geq \operatorname{tr} \rho_\epsilon(I)\rho_H - \lambda^m \operatorname{tr} \rho_\epsilon(I)^2 \geq \operatorname{tr} \rho_H^2 \left(1 - \frac{\lambda^m}{Vol(\epsilon/3)^2}\right) \geq \frac{1}{2} \operatorname{tr} \rho_H^2 \geq \frac{1}{2k^{N^2}}, \tag{39}$$

where in the second-to-last step we have assumed $m \geq \log(2/Vol(\epsilon/3)^2)/\log(1/\lambda) = \mathcal{O}(N^2 \log_{1/\lambda}(1/\epsilon))$.

On the other hand, if there is no string $s_1, ..., s_m$ such that $d(U(s_1)U(s_2)...U(s_m), U) \leq \epsilon$, then

$$\operatorname{tr} \rho_\epsilon(I)\mathcal{E}^m(\rho_\epsilon(U)) \leq \left(1 - \frac{\epsilon}{6N}\right)^{2k} \leq e^{-\frac{k\epsilon}{3N}}. \tag{40}$$

If $k/\log k \gg N^3/\epsilon$ then (39) and (40) cannot simultaneously hold. Therefore there must exist at least one string $s_1, \ldots, s_m$ for which $d(U(s_1)U(s_2)...U(s_m), U) \leq \epsilon$. ∎

3.3 Trace Method and Schwinger-Dyson Equations

The next three sections are devoted to the expansion properties of randomly chosen unitaries. Recall that we would like to construct a quantum tensor product expander by randomly choosing $U(1), \ldots, U(D) \in \mathcal{U}_N$. There are two cases. In the non-Hermitian case, the unitary matrices $U(s)$ are chosen independently with the Haar measure. In the Hermitian case, D is even and the unitary matrices $U(s)$ for $s = 1, \ldots, D/2$ are chosen independently with the Haar measure and $U(s + D/2) = U(s)^\dagger$, so that $\mathcal{E}_k$ is a Hermitian operator. We focus on the Hermitian case, and the techniques can be readily extended to cover the non-Hermitian case. Our main result is that for random $U(s)$, with high probability we do indeed get a tensor product expander:

Theorem 6 . *Let $\{U(1), \ldots, U(D/2)\}$ be chosen randomly with the Haar measure from the unitary group $\mathcal{U}_N$, and let $U(s + D/2) = U(s)^\dagger$. Let $k \leq \mathcal{O}(N^{1/6}/\log(N))$ and let λ denote the $F_k^N + 1^{st}$ eigenvalue of $\mathcal{E}_k$ as defined in (32). Then, for any $c > 1$,*

$$\Pr\left[\lambda \geq c(1 + \mathcal{O}(k \log(N) N^{-1/6})\lambda_H\right] \leq c^{-(1/4k)N^{1/6}}, \tag{41}$$

where λ_H depends on D and is given in Eq. (14).

We use a trace method to bound the eigenvalues of $\mathcal{E}_k(M)$. We have

$$\begin{aligned} &\sum_{i_1,i_2,\ldots,i_k} \sum_{j_1,j_2,\ldots,j_k} \Big(M(i_1,j_1) \otimes M(i_2,j_2) \otimes ... \otimes M(i_k,j_k), \mathcal{E}_k^m(M(i_1,j_1) \otimes \\ &\qquad \otimes M(i_2,j_2) \otimes ... \otimes M(i_k,j_k))\Big) \\ = \quad &\sum_{a=1}^{N^{2k}} |\lambda_a|^m \geq k! + |\lambda|^m, \end{aligned} \tag{42}$$

where we pick m to be an even integer. We will derive bounds on the expectation value of the trace to bound the expectation of $|\lambda|^m$. Eq. (42) can be re-written as

$$k! + |\lambda|^m \leq \left(\frac{1}{D}\right)^m \sum_{s_1=1}^{D} \sum_{s_2=1}^{D} ... \sum_{s_m=1}^{D} \mathrm{tr}(U(s_m + D/2)...U(s_2 + D/2)U(s_1 + D/2))^k \mathrm{tr}(U(s_1)U(s_2)...U(s_m))^k]. \quad (43)$$

Let $E[...]$ denote the average over the unitary group. Averaging Eq. (43) we find

$$E_{1,k} \equiv \left(\frac{1}{D}\right)^m \sum_{s_1=1}^{D} \sum_{s_2=1}^{D} ... \sum_{s_m=1}^{D} E_{0,k}(s_1, ..., s_m) \geq k! + E[|\lambda|^m], \quad (44)$$

$$\begin{aligned} E_{0,k}(s_1, ..., s_m) &\equiv E[\mathrm{tr}(U^\dagger(s_m)...U^\dagger(s_2)U^\dagger(s_1))^k \mathrm{tr}(U(s_1)U(s_2)...U(s_m))^k] \\ &= E[\mathrm{tr}(U(s_m + D/2)...U(s_2 + D/2)U(s_1 + D/2))^k \\ &\quad \mathrm{tr}(U(s_1)U(s_2)...U(s_m))^k]. \end{aligned} \quad (45)$$

As in [19], we write the average in Eq. (45) as an average of the form

$$E[L_1 L_2 ... L_c], \quad (46)$$

where

$$L_1 = \mathrm{tr}(U(s_{1,1})U(s_{1,2})...U(s_{1,m_1})), \quad L_2 = \mathrm{tr}(U(s_{2,1})U(s_{2,2})...U(s_{2,m_2})), \quad ... \quad (47)$$

Here we have an average of c traces, each of which is a product of some number of unitary matrices. In particular, Eq. (45) has $c = 2k$, with $L_1 = L_2 = ... = L_k = L_{k+1}^\dagger = ... = L_{2k}^\dagger$.

The Schwinger-Dyson equations for a product of this form are[19]:

$$\begin{aligned} &E[\mathrm{tr}(U(s_{1,1})U(s_{1,2})...U(s_{1,m_1}))L_2...L_c] \quad (48) \\ &= -\frac{1}{N} \sum_{j=2}^{m_1} \delta_{s_{1,1},s_{1,j}} E[\mathrm{tr}(U(s_{1,1})...U(s_{1,j-1}))\mathrm{tr}(U(s_{1,j})...U(s_{1,m_1}))L_2...L_c] \\ &+ \frac{1}{N} \sum_{j=2}^{m_1} \delta_{s_{1,1},s_{1,j+D/2}} E[\mathrm{tr}(U(s_{1,2})...U(s_{1,j-1}))\mathrm{tr}(U(s_{j+1,1})...U(s_{1,m_1}))L_2...L_c] \\ &- \frac{1}{N} \sum_{l=2}^{c} \sum_{j=1}^{m_l} \delta_{s_{1,1},s_{l,j}} E[\mathrm{tr}(U(s_{1,1})...U(s_{1,m_1})U(s_{l,j})U(s_{l,j+1})...U(s_{l,j-1})) \\ &\qquad L_2...L_{l-1}L_{l+1}...L_c] \\ &+ \frac{1}{N} \sum_{l=2}^{c} \sum_{j=1}^{m_l} \delta_{s_{1,1},s_{l,j}+D/2} E[\mathrm{tr}(U(s_{1,2})...U(s_{1,m_1})U(s_{l,j+1})U(s_{l,j+2})...U(s_{l,j-1})) \\ &\qquad L_2...L_{l-1}L_{l+1}...L_c]. \end{aligned}$$

Note that in the above equation an expression like $U(s_{l,j+1})U(s_{l,j+2})...U(s_{l,j-1})$ means $U(s_{l,j+1})U(s_{l,j+2})...U(s_{l,m_l})U(s_{l,1})U(s_{l,2})...U(s_{l,j-1})$.

Our general algorithm for reducing traces starts by canceling all pairs of matrices $U(s)U(s+D/2)$ appearing successively in the same trace, and replacing $\mathrm{tr}(I)$ by N. We then apply Eq. (48), repeating the cancellation of successive $U(s)U(s+D/2)$ and replacement of $\mathrm{tr}(I)$ by N on each iteration. A term terminates at a given level n if there are no matrices left after n iterations.

Let m_1^0 be the length of the trace after canceling successive $U(s)U(s+D/2)$ before any iterations; on every successive iteration, the length of the first trace, m_1, is bounded by m_1^0. As in [19], the number of different choices of $s_1, ..., s_m$ which give rise to a given m_1^0 is bounded by

$$(D-1)^{m_1^0/2}(D-1)^{m/2}2^m. \tag{49}$$

This number is equal to D^m times the probability that a random walker on a Cayley tree arrives at a distance m_1^0 from the starting point after a walk of m steps. This number is independent of the particular values of $s_{1,1}, ..., s_{1,m_1^0}$. There are $[D/(D-1)](D-1)^{m_1^0}$ different possible values of $s_{1,1}, ..., s_{1,m_1^0}$ and therefore the total number of choices of $s_1, ..., s_m$ which give rise to a given choice of $s_{1,1}, ..., s_{1,m_1^0}$ is bounded by

$$\frac{D-1}{D}\Big(\frac{1}{\sqrt{D-1}}\Big)^{m_1^0}(D-1)^{m/2}2^m. \tag{50}$$

The number of terms terminating at the n^{th} level is bounded by

$$(2km-1)^n. \tag{51}$$

To see this, note that at each iteration of the Schwinger-Dyson equation, the number of terms on the right-hand side is bounded by the number of matrices on the left-hand side minus one. Initially, there are $2km$ matrices, and this number does not increase under Eq. (48).

We can estimate the value of a term which terminates at a given level $n > 1$ as follows. First, there is a sign equal to plus or minus 1. Next, there is a factor of $(1/N)^n$. Finally, there is a factor of N for each trace of the form $\mathrm{tr}(I)$ that appeared in this process. Suppose there are p such traces, giving a factor of N^p. How big can p be? Initially we have $c = 2k$ different traces. The given term at level n arose from a specific choice of terms on the right-hand side of Eq. (48) on the first iteration. This specific choice has k_1 different traces in it, with k_1 equal to either $k-1$ or $k+1$. After the second iteration there are k_2 traces, then k_3, and so on. The number of traces $k_2, k_3, ...$ can be determined as follows: an application of Eq. (48) may increase the number of traces by one if the term arises from the first or second line on the right-hand side, or may decrease the number of traces by one if the term arises from the third or fourth line on the right-hand side of Eq. (48). Next, some of the traces may be trivial, being equal to $\mathrm{tr}(I)$. In the event that the term arose from the first, second, or third line of Eq. (48) it is not possible for any of the traces to be trivial, under the assumption that any repetitions of the form $U(s)U(s+D/2)$ have been previously replaced by I in the trace on the left-hand side of the equation. However, in the event that the term arose from the fourth line, then it is possible for one of the traces to be trivial, increasing p by one. Thus, for each $b \leq n$, $k_b - k_{b-1}$ is equal to either $+1, -1$, or -2. Let q be equal to the number of times the first or second line was used from Eq. (48) and $n-q$ equal the number of times the third or fourth line was used. Then, in order for all traces to be trivial in this particular term resulting

from n iterations of Eq. (48),

$$2k + q - (n - q) - p = 0. \tag{52}$$

Also, since p can only increase when a term from the fourth line is used,

$$p \leq n - q. \tag{53}$$

Thus,

$$p \leq \lfloor (2k + n)/3 \rfloor. \tag{54}$$

Therefore, the value of a term terminating at the n^{th} level, $n > 0$, is bounded in absolute value by

$$N^{\lfloor (2k+n)/3 \rfloor - n}. \tag{55}$$

Note that if $m_1^0 > 0$ then there are no terms terminating at level n with $n < k$, so for $m_1^0 = 0$, the trace is equal to N^{2k}, while for $m_1^0 > 0$, the terms are bound in absolute value by N^0 (this bound is only reached if $k = n$).

Eq. (48) generates an infinite series, whose n^{th} term is the sum of all terms terminating at level n. As in [19], this series is absolutely convergent for $2km < N$. In fact, the following stronger claim holds: Eq. (48) generates an absolutely convergent series for $2km - 1 < N$ which converges to the expectation value of the trace. To see this, note that the value p above, the number of traces of I, is always bounded by $2km$. Thus, the value of a term terminating at the n^{th} level is bounded by

$$N^{2km} N^{-n}. \tag{56}$$

Depending on n, sometimes (55) gives a better bound and sometimes (56) gives a better bound, but to estimate convergence we will use (56). Eq. (51) shows that the number of terms terminating at level n is bounded by $(2km - 1)^n$. Thus, the absolute value of the sum of terms terminating at level n is bounded by $N^{2km}((2km-1)/N)^n$, and so for $2km-1 < N$, the series is absolutely convergent. Further, a term which has not terminated at the n^{th} level contains at most $2km$ traces in it, and hence is bounded in absolute value by $N^{2km}(1/N)^n$. Therefore, the sum of all terms which have not terminated at the n^{th} level is also bounded by $N^{2km}((2km - 1)/N)^n)$, and hence for $2km - 1 < N$ the series converges to the average of the trace.

3.4 Example

We now work out a simple example to give some idea of the use of the Schwinger-Dyson equations. This example will also be used later in the idea of "complete rung cancellation" and gives intuition behind the claim that for $N \geq k$ we have $k!$ eigenvalues equal to unity. Let the matrix X be chosen from the unitary group with the Haar measure and evaluate the expectation value for $N \geq k$

$$E[\Big(\mathrm{tr}(X)\mathrm{tr}(X^\dagger)\Big)^k]. \tag{57}$$

For $k = 1$, a single application of Eq. (48) shows that this is equal to unity. For $k = 2$, we find

$$E\left[\Big(\mathrm{tr}(X)\mathrm{tr}(X^\dagger)\Big)^2\right] = 2E\left[\Big(\mathrm{tr}(X)\mathrm{tr}(X^\dagger)\Big)\right] - (1/N)E\left[\mathrm{tr}(XX)\Big(\mathrm{tr}(X^\dagger)^2\Big)\right] \tag{58}$$

$$\begin{aligned} &= 2 + (1/N)^2 E\left[\left(\mathrm{tr}(X)\mathrm{tr}(X^\dagger)\right)^2\right] - 2(1/N)^2 E\left[\mathrm{tr}(X)\mathrm{tr}(X^\dagger)\right] \\ &= 2 + (1/N)^2 E\left[\left(\mathrm{tr}(X)\mathrm{tr}(X^\dagger)\right)^2\right] - 2(1/N)^2. \end{aligned}$$

For $N \geq 2$, this shows that $E[\left(\mathrm{tr}(X)\mathrm{tr}(X^\dagger)\right)^2] = 2$.

It is interesting to see what happens to the expectation value in Eq. (58) for $N = 1, k = 2$. Then, the last line Eq. (58) gives simply $E[\left(\mathrm{tr}(X)\mathrm{tr}(X^\dagger)\right)^2] = E[\left(\mathrm{tr}(X)\mathrm{tr}(X^\dagger)\right)^2]$, giving no information about the trace. For general N, the sum of terms terminating at level 1 is equal to zero, while the sum of terms terminating at levels $2, 3, 4, 5, 6...$ is equal to $2, -2/N, 2/N, -2/N^2, 2/N^2, ...$ respectively. Thus, we do not have a convergent series for $N = 1, k = 2$.

Up to now we have considered the series whose n^{th} term is the sum of terms terminating at a given level n. We now consider instead the expectation value of Eq. (57) as a series in $1/N$. For $N \geq k$, this series is again absolutely convergent to the desired expectation value. It is easy to see that for arbitrary k, and for $N \gg k$, the expectation value (57) is equal to $k! + \mathcal{O}(1/N)$, as there are $k!$ terms which terminate at level k. We now show that for $N \geq k$, the expectation value (57) is equal to $k!$ exactly. Note that the expectation value in Eq. (57) is equal to the trace of the map $\mathcal{T}_k$ (defined in (34))

Thus, the trace of the map $\mathcal{T}_k(M)$ is equal to the number of unit eigenvalues of $\mathcal{T}_k(M)$. For $N \geq k$ the trace of this map can then be written as the sum of an infinite series in $1/N$, and using the fact that the number of unit eigenvalues is equal to an integer for all integer N, we find that all terms in the series in $1/N$, beyond the term of order N^0, must vanish exactly (the calculation above represents an explicit check of this for $k = 2$ and it may be readily verified for any k). Thus, for all $N \geq k$, the expectation value of Eq. (57) is equal to $k!$. This gives an alternate proof that $F_k^N = k!$ when $N \geq k$.

3.5 Counting and Main Result

In this section we prove a bound on the expectation value of the sum in Eq. (44), which will give us a bound on the expectation value of the m^{th} power of λ, proving the theorem. The next three paragraph are devoted to outlining the basic idea of the proof, before beginning the technical details.

The basic idea of the proof is to prove the bound on the sum by proving a bound on the number of different choices of $s_1, ..., s_m$ such that, when the resulting trace is evaluated using the Schwinger-Dyson equations, there is a term which terminates at level n, for any given n. We give this bound on the number of choices of $s_1, ..., s_m$ in Eq. (61). We then combine this bound with a bound on the contribution to the trace of terms which terminate at level n. The idea is that there are a only small number of choices of $s_1, ..., s_m$ which produce terms which terminates at a small level n, and while there are a large number of choices of $s_1, ..., s_m$ which produce terms which terminate at high levels, such terms are small.

One technical caveat in this work is that for *any* choice of $s_1, ..., s_m$ there will be certain terms which terminate at a low level n. These are terms in which we use the Schwinger-Dyson equations to contract $U(s_i)$ in one trace with $U(s_i)^\dagger$ in a different trace. If for some i, we contract all unitaries $U(s_i)$ in this way, we have what is called a "complete rung cancellation"

below. We consider such terms separately, and they are responsible for producing the leading order expectation value of the trace in $1/N$: these terms sum to give a contribution $k!$ to the expectation value of the trace, precisely corresponding to the expectation we expect from the unit eigenvalues.

Ignoring those terms with complete rung cancellations, we see that a term in the Schwinger-Dyson equations must involve contracting $U(s_i)$ with $U(s_j)$ or $U(s_j)^\dagger$ for some $i \neq j$. Such terms involve constraints: such a term would require that either $s_i = s_j + D/2$ or $s_i = s_j$. In order for such a term to terminate at a low level, there must be many such constraints, and this is why there are only a few choices of $s_1, ..., s_m$ which produce terms which terminate at low levels. To show precisely that there are only a few such choices of $s_1, ..., s_m$, we follow a different strategy. To explain this strategy, suppose you knew a choice of $s_1, ..., s_m$ which gave rise to a term which terminated at some level n and you were given the task of explaining to someone which choice of $s_1, ..., s_m$ you used. One way to do this would be to simply list the m different values of s. This would require communicating $\log_2(D^m)$ bits. We instead show how to uniquely specify the choices of $s_1, ..., s_m$ in a different way, by specifying most of the choices of $s_1, .., .s_m$ by describing which cancellations were used. For small n, this will allow one to communicate the specific choice of $s_1, .., s_m$ in much shorter way, thus implying that that there are only a few choices of $s_1, ..., s_m$ which produce the desired term terminating at level n. We now put this idea into practice.

On a given iteration of the Schwinger-Dyson equations, we go from a product of c traces to a product of $c+1$, $c-1$, or $c-2$ traces. As in [19], we keep track of how the matrices move under this iteration process using a function $f_n((l,i))$ from pairs of integers to pairs of integers. We say that the matrix $U(s_{l,i})$ in the given product of traces, $L_1 L_2 ... L_c$, is in position (l,i). Let us consider the case of a term on the first line, where c increases by one. Then, for any given j in the sum on the first line, we say that the matrix in position $(1,i)$, for $i < j$ on the $n+1^{st}$ iteration corresponds to the matrix in position $(1,i)$ on the n^{th} iteration, and so $f_n((1,i)) = (1,i)$, while the matrix in position $(2,i)$ on the $n+1^{st}$ iteration corresponds to the matrix in position $(1, i+j-1)$ on the n^{th} iteration, so $f_n((1,i+j-1)) = (2,i)$. The matrix in position (l,i), for $2 < l \leq k+1$ on the $n+1^{st}$ iteration corresponds to the matrix $(l-1,i)$ on the n^{th} iteration, so $f_n(l-1,i) = (l,i)$. We follow a similar procedure for the other lines of Eq. (48) and if there are cancellations, we keep track of how the matrix moves under the cancellations.

We then keep track of which matrix after n iterations corresponds to a given matrix before any iterations, by defining $F_n((l,i)) = f_n(f_{n-1}(...f_1((l,i))$ for $l = 1, 2, ..., 2k$. Let us say that the matrix at position (l,i) is "trivially moved" under the n^{th} iteration of the Schwinger-Dyson equations if if it is not in either position $(1,1)$ or position $(1,j)$ using a term on the first or second line, or in either position $(1,1)$ or position (l,j) using a term from the third or fourth line. If a matrix is not trivially moved, and the matrix is not in position $(1,1)$, then the Schwinger-Dyson equations imply a relation between $s_{l,i}$ and $s_{1,1}$.

A given term in Eq. (48) arises from a given choice of (l,j): for a term on the first or second line let us say $l = 1$. Let $(1,1) = F_n(l_0, j_0)$ and let $(l,j) = F_n(l_0', j_0')$. If a matrix is *not* trivially moved under on the n^{th} iteration then there are two cases: (1) either $l_0 \leq k$ and $l_0' \leq k$ or $l_0 > k$ and $l_0' > k$. That is, either both matrices appeared in one of the first k traces, which are traces of products of conjugates of unitaries, or both matrices appeared in one of

the last k traces, which are traces of unitaries. Or, case (2): $l_0 \leq k$ and $l_0' > k$ or $l_0 > k$ and $l_0' \leq k$. That is, one matrix was in one of the first k traces and the other was in one of the last k traces. We then break the first case into two sub-cases: (a), $j_0 = j_0'$ or (b), $j_0 \neq j_0'$. We also break the second case into two sub-cases: (a), $j_0 = m_1 + 1 - j_0'$ or (b), $j_0 \neq m_1 + 1 - j_0'$. In case $1a$ both matrices are unitary matrices $U(s_{1,j_0})$ or both are $U(s_{1,j_0})^\dagger$ and in case $2a$, one matrix is $U(s_{1,j_0})$ and the other is $U(s_{1,j_0})^\dagger$. In case $1b$, we know that $s_{1,j_0} = s_{1,j_0'}$ for $j_0 \neq j_0'$ while in case $2b$ we know that $s_{1,j_0} = s_{1,j_0'} + D/2$ for $j_0 \neq j_0'$. Thus, in case $1b$ or $2b$ the term in the Schwinger-Dyson equation implies some constraint about the choice of $s_{1,j}$. To illustrate these different cases, consider the example (58): the first term on the right-hand side of the top line is an example of case $2a$, while the second term on the same line is an example of case $1a$.

Consider a given j; if on some iteration and for some l the matrix which was originally in position (l, j) is not trivially moved and we have case $1b$ or $2b$, then we can identify some k such that either $s_{1,j} = s_{1,k}$ or $s_{1,j} = s_{1,k} + D/2$. Let us write $k = \tau(j)$ in both cases, for some function $\tau(j)$. We define a term to have a "complete rung cancellation of matrix j" if it is not possible to identify such a k for the given j. We claim that the sum of all terms with a complete rung cancellation of matrix i is equal to $k!$ so long as $k \leq N$. To show this, consider the product of traces

$$\begin{aligned} &\mathrm{tr}(U(s_m + D/2)...U(s_{i+1} + D/2)X^\dagger U(s_{i-1} + D/2)...U(s_1 + D/2))^k \times \\ &\times \mathrm{tr}(U(s_1)...U(s_{i-i})XU(s_{i+1})...U(s_m)^k, \end{aligned} \tag{59}$$

where X is some arbitrary unitary matrix. Averaging this trace over all unitary matrices $U(s)$ and over all unitary matrices X with the Haar measure, we find that the trace is equal to $k!$: this can be established by applying Eq. (48) to this trace, and always cyclically permuting the trace so that X is in the first position. This calculation is very similar to the example calculation (57) above. However, applying the Schwinger-Dyson equations to the trace (59) without first applying the cyclic permutation generates precisely the sum of terms mentioned above, those in which there is a complete rung cancellation of matrix i. Thus, this sum of terms equals $k!$. We further claim that for any given $i_1, i_2, ..., i_d$, the sum of all terms with complete rung cancellations of matrices $i_1, i_2, ...i_d$ is equal to $k!$, as may be shown by considering a trace in which matrices $U(s_{i_1}), U(s_{i_2}), ...$ are replaced by $X_1, X_2, ...$, and the trace is averaged over the different $X_1, X_2, ...$ Then, using the inclusion-exclusion principle, the sum of terms in which for no i is there a complete rung cancellation of matrix i is equal to the sum of all terms minus $k!$. So, we now focus on the sum of terms with no complete rung cancellations, which we define to be $E'_{0,k}(s_1, ..., s_m)$; if a given choice of $s_1, ..., s_m$ gives rise to a term which terminates at level n with no complete rung cancellations, then it is possible to identify a $\tau(i)$ for each i.

We now follow the same approach as in [19] to bound the number of choices of $s_1, ..., s_{m_1^0}$ which can produce a term which terminates at a level n with no complete rung cancellations. Given the sequence of choices of terms on the right-hand side of the Schwinger-Dyson equation (48), as well as knowledge of which cancellations occurred at each iteration, we know the function $\tau(i)$, and given this function $\tau(i)$ there are now only at most $[D/(D-1)](D-1)^{m_1^0/2}$ possible values of $s_{1,1}, ..., s_{1,m_1^0}$. Thus, the total number of choices of $s_1, ..., s_{m_1^0}$ which can produce a term which terminates at level n is bounded by the number of possible choices

of terms and cancellations in the Schwinger-Dyson equation (48) at each of the n iterations multiplied by $[D/(D-1)](D-1)^{m_1^0/2}$. At each iteration of the Schwinger-Dyson equations, we make a particular choice of l, j at each level, which requires specifying one particular matrix out of all the matrices on the right-hand side; there are at most $2m_1k-1$ matrices on the right-hand side, so there are at most $2m_1k-1$ choices (in [19], the slightly worse bound $(2m_1k-1)^2$ was found; we tighten the bound here). At each such iteration of the Schwinger-Dyson equations, there may be cancellations in two different traces if the term came from the second line of Eq. (48), with at most m_1 cancellations in each trace, or cancellations in two different places of a single trace, if the term came from the fourth line of Eq. (48), with at most m_1 cancellations in each place. Let us call the number of cancellations c_1, c_2 with $0 \leq c_1 \leq m_1$ and $0 \leq c_2 \leq m_1$. Then, by specifying l, j, c_1, c_2 for each iteration, we succeed in fully specifying how the matrices move under the n iterations of the Schwinger-Dyson equation; this requires specifying n numbers ranging from $1...2km_1 - 1$, and $2n$ numbers ranging from $0...m_1$.

Thus, there are at most

$$[D/(D-1)](D-1)^{m_1^0/2}(2km_1^0-1)^n(m_1^0+1)^{2n} \leq [D/(D-1)](D-1)^{m_1^0/2}(2km_1^0)^{3n} \quad (60)$$

choices of $s_1, ..., s_{m_1^0}$ which can produce a term which terminates at level n. Using Eq. (50), the number of choices of $s_1, ..., s_m$ which can produce a term which terminates at level n is at most

$$\sum_{m_1^0=0}^{m} (D-1)^{m/2}2^m(2km_1^0)^{3n} \leq (D-1)^{m/2}2^m\frac{(2km+1)^{3n+1}}{3n+1}. \quad (61)$$

For any $s_1, ..., s_m$, we define $n_{min}(s_1, ..., s_m)$ to be the smallest level at which a term terminates with no complete rung cancellations. The sum of terms with $m_1^0 = 0$, which is the same as the sum of terms with $n_{min} = 0$, is bounded by

$$N^{2k}D^{-m}(D-1)^{m/2}2^m = N^2\lambda_H^m. \quad (62)$$

Thus, we re-write the sum in Eq. (44) as

$$E_{1,k} \leq k! + N^{2k}N^2\lambda_H^m\Big(\frac{1}{D}\Big)^m \sum_{n=k}^{\infty}\sum_{s_1=1}^{D}\sum_{s_2=1}^{D}...\sum_{s_m=1}^{D}\delta_{n_{min}(s_1,...,s_m),n}E'_{0,k}(s_1, ..., s_m). \quad (63)$$

Therefore, for any $s_1, ..., s_m$ with $n_{min} > 0$,

$$\begin{aligned} E'_{0,k}(s_1, ..., s_m) &\leq \sum_{n \geq n_{min}(s_1,...,s_m)} N^{2(k-n)/3}(2km-1)^n \quad (64)\\ &= N^{2k/3}\frac{[N^{-2/3}(2km-1)]^{n_{min}}}{1-N^{-2/3}(2km-1)}. \end{aligned}$$

From Eqs. (61,63,64),

$$\begin{aligned} E_{1,k} &\leq k! + \lambda_H^m\Big\{N^2 + N^{2k/3}\sum_{n=k}^{\infty}\frac{(2km+1)^{3n+1}}{3n+1}\frac{[N^{-2/3}(2km-1)]^n}{1-N^{-2/3}(2m-1)}\Big\} \quad (65)\\ &\leq 1 + \lambda_H^m\Big\{N^2 + N^{2k/3}\sum_{n=k}^{\infty}\frac{2km+1}{(3n+1)[1-N^{-2/3}(2km-1)]}[N^{-2/3}(2km+1)^4]^n\Big\}. \end{aligned}$$

We then pick $m = (1/4k)N^{1/6}$, so that $N^{-2/3}(2km+1)^4 \leq 1/2$ and

$$\begin{aligned} |\lambda| &\leq (E_{1,k}-1)^{1/m} \leq N^{2/m}\lambda_H(1+\mathcal{O}(1))^{1/m} \\ &= \lambda_H(1+\mathcal{O}(\log(N)kN^{-1/6}). \end{aligned} \tag{66}$$

Using Markov's inequality, the probability that $|\lambda|$ is greater than $c(1+\mathcal{O}(k\log(N)N^{-1/6})\lambda_H(D)$, for any $c \geq 1$, is bounded by $c^{-(1/4k)N^{1/6}}$. ∎

4 Discussion

We have introduced quantum and classical tensor product expanders. These provide a way to approximate t-designs by acting many times with a small number of unitaries. An important open question is whether efficient implementations of these tensor product expanders exist.

Acknowledgments

AWH thanks Richard Low for catching an error in the proof of Theorem 4, as well as useful discussions about Lemma 1. MBH thanks the KITP for hospitality while some of this research was completed. MBH was supported in part by the National Science Foundation under Grant No. PHY05-51164 and supported by U. S. DOE Contract No. DE-AC52-06NA25396. AWH was supported by the European Commission under a Marie Curie Fellowship (ASTQIT, FP-022194), the integrated EC project "QAP" (contract no. IST-2005-15848), the U.K. EPSRC, project "QIP IRC" and the Army Research Office under grant W9111NF-05-1-0294.

References

1. C. Dankert, R. Cleve, J. Emerson, and E. Livine. Exact and Approximate Unitary 2-Designs: Constructions and Applications. arXiv:quant-ph/0606161, 2006; D. Gross, K. Audenaert, and J. Eisert. Evenly distributed unitaries: on the structure of unitary designs *J. Math. Phys.* 48, 052104, 2007. arXiv:quant-ph/0611002.
2. R. F. Werner. Quantum states with Einstein-Podolsky-Rosen correlations admitting a hidden-variable model *Phys. Rev. A* **40**, 4277, 1989; C. H. Bennett et. al. Purification of Noisy Entanglement and Faithful Teleportation via Noisy Channels. *Phys. Rev. Lett.* **76**, 722, 1996. arXiv:quant-ph/9511027; C. H. Bennett, D. P. Divincenzo, J. A. Smolin, W.K. Wootters. Mixed State Entanglement and Quantum Error Correction. *Phys. Rev. A* **54**, 3824, 1996. arXiv:quant-ph/9604024.
3. M. B. Hastings. Entropy and Entanglement in Quantum Ground States. *Phys. Rev. B* **76**, 035114, 2007. arXiv:cond-mat/0701055.
4. A. Ben-Aroya and A. Ta-Shma. Quantum expanders and the quantum entropy difference problem. arXiv:quant-ph/0702129, 2007.
5. A. Ben-Aroya, O. Schwartz and A. Ta-Shma. An explicit construction of quantum expanders. arXiv:quant-ph/0709.0911; A. Ben-Aroya, O. Schwartz and A. Ta-Shma. Quantum Expanders: Motivation and Constructions. *Proc. of 2008 IEEE CCC*, pp. 292–303, 2008.
6. J. Eisert and D. Gross. Quantum Margulis expanders. *Q. Inf. Comp.*, vol. 8, pp. 722–733, 2008. arXiv:0710.0651.
7. A. W. Harrow. Quantum expanders from any classical Cayley graph expander. *Q. Inf. Comp.*, vol. 8, no. 8/9, pp. 715-721, 2008. arXiv:quant-ph/0709.1142.
8. S. Hoory, N. Linial, and A. Wigderson. Expander graphs and their applications. *Bull. Amer. Math. Soc.* **43**, 439, 2006.
9. J. Friedman. A proof of Alon's second eigenvalue conjecture and related problems. *Proc. of 35th annual STOC*, pp. 720–724, 2003. arXiv:cs/0405020.

10. G. Aubrun. A remark on the paper 'Randomizing quantum states: constructions and applications'. arXiv:0802.4193, 2008.
11. P. Hayden, D. W. Leung, P. W. Shor and A. J. Winter. Randomizing quantum states: constructions and applications. *Commun. Math. Phys.* vol. 250, no. 2, pp. 371–391, 2004. arXiv:quant-ph/0307104.
12. P. Dickinson and A. Nayak. Approximate Randomization of Quantum States With Fewer Bits of Key. *AIP Conference Proceedings*, **864**, 18, 2006. arXiv:quant-ph/0611033
13. A. Ambainis and A. Smith. Small Pseudo-Random Families of Matrices: Derandomizing Approximate Quantum Encryption. *Proc. RANDOM 2004*, pp. 249–260, 2004. arXiv:quant-ph/0404075.
14. A. Lubotsky, R. Phillips and P. Sarnak. Hecke operators and distributing points on the sphere I. *Comm. Pure Appl. Math.* **39**, Supplement I, S149-S186, 1986; A. Lubotsky, R. Phillips and P. Sarnak. Hecke operators and distributing points on the sphere II. *Comm. Pure Appl. Math.* **40**, 401-420, 1987.
15. C. Zalka. Implementing high dimensional unitary representations of $SU(2)$ on a Quantum Computer. arXiv:quant-ph/0407140, 2004.
16. A. Brodsky and S. Hoory. Simple Permutations Mix Even Better. arXiv:math/0411098; S. Hoory, A. Magen, S. Myers, and C. Rackoff. Simple Permutations Mix Well. *Proc. 31st ICALP*, 2004.
17. M. Kassabov. Symmetric groups and expanders. *Inventiones Mathematicae*, vol. 170, n. 2, 2007. arXiv:math.GR/0505624
18. A. Brodir and E. Shamir. On the second eigenvalue of random regular graphs. *Proc. 28th FOCS*, pp. 286–294, 1987.
19. M. B. Hastings. Random unitaries give quantum expanders. *Phys. Rev. A* **76**, 032315, 2007. arXiv:0706.0556
20. G. Tóth and J. J. García-Ripoll. Efficient algorithm for multi-qudit twirling for ensemble quantum computation. *Phys. Rev. A* **75**, 042311, 2007. arXiv:quant-ph/0609052.
21. A. Y. Kitaev, A.H. Shen and M.N. Vyalyi. *Classical and Quantum Computation*. American Mathematical Society, Providence, RI. 2002.
22. A. W. Harrow, B. Recht and I. L. Chuang. Efficient Discrete Approximations of Quantum Gates. *J. Math. Phys.* **43**, 4445 (2002). arXiv:quant-ph/0111031
23. J. Bourgain and A. Gamburd. New results on expanders. *C. R. Acad. Sci. Paris*, Ser. I. vol. 342, pp. 717–721, 2006.
24. R. Goodman and N. R. Wallach. *Representations and invariants of the classical groups*, 1998.
25. A. Barenco, A. Berthiaume, D. Deutsch, A. Ekert, R. Jozsa and C. Macchiavello. Stabilization of quantum computation by symmetrization. *SIAM J. Comput.* vol. 26, no. 5, pp. 1541–1557, 1997. arXiv:quant-ph/9604028
26. A. Gamburd, D. Jakobson and P. Sarnak. Spectra of elements in the group ring of $SU(2)$. J. Eur. Math. Soc., vol. 1, pp. 51–85, 1999.

Appendix A Proof of Lemma 1

First, we reduce to the case when the matrices are 2×2 with $\Pi = |1\rangle\langle 1|$ and X is diagonal. Express $\|pX + (1-p)Y\|$ as the maximum of $\langle\psi|pX + (1-p)Y|\psi\rangle$ over all unit vectors $|\psi\rangle$. Write $|\psi\rangle$ as $|\psi\rangle = \cos(\theta)|\psi_1\rangle + \sin(\theta)|\psi_2\rangle$, where $0 \leq \theta \leq \pi/2$ and $|\psi_1\rangle, |\psi_2\rangle$ are normalized vectors such that $\Pi|\psi_1\rangle = |\psi_1\rangle$ and $(I-\Pi)|\psi_2\rangle = |\psi_2\rangle$. Our conditions on X imply that $\langle\psi|X|\psi\rangle = \cos^2(\theta) + \langle\psi_2|X|\psi_2\rangle \sin^2(\theta)$ and that $|\langle\psi_2|X|\psi_2\rangle| \leq 1 - \epsilon_X$. Next, for $i,j = 1,2$ define $Y_{i,j} = \langle\psi_i|Y|\psi_j\rangle$. Since $\|Y\| \leq 1$, we also have that $\|\sum_{i,j=1}^2 Y_{i,j}|i\rangle\langle j|\| \leq 1$. We can now replace Y with $\sum_{i,j=1}^2 Y_{i,j}|i\rangle\langle j|$ and X with $|1\rangle\langle 1| + \langle\psi_2|X|\psi_2\rangle\ |2\rangle\langle 2|$.

Now suppose that $|\langle\psi|X|\psi\rangle| \geq 1 - \epsilon_X\epsilon_Y/12$. Using our bound on $|\langle\psi_2|X|\psi_2\rangle|$, we obtain

$$1 - \frac{\epsilon_X\epsilon_Y}{12} \leq \cos^2(\theta) + \sin^2(\theta)(1-\epsilon_X) = 1 - \sin^2(\theta)\epsilon_X,$$

implying that $\sin^2(\theta) \leq \epsilon_Y/12$. We will show that this yields an upper bound on $\langle\psi|Y|\psi\rangle$.

Since $\|Y\| \leq 1$, we have

$$|Y_{1,2}|, |Y_{2,1}| \leq \sqrt{1 - |Y_{1,1}^2|}.$$

Thus

$$\begin{aligned} |\langle\psi|Y|\psi\rangle| &\leq \cos^2(\theta)|Y_{1,1}| + \sin(\theta)\cos(\theta)(|Y_{1,2}| + |Y_{2,1}|) + \sin^2(\theta)|Y_{2,2}| \\ &\leq \cos(\theta)|Y_{1,1}| + \sin(\theta)2\sqrt{1 - |Y_{1,1}|^2} + \frac{\epsilon_Y}{12}. \end{aligned} \tag{A.1}$$

If θ were not constrained then the first two terms of (A.1) would be maximized by taking θ to be $\hat{\theta} = \arctan(2\sqrt{1 - |Y_{1,1}|^2}/|Y_{1,1}|) \geq \arctan(2\sqrt{2\epsilon_Y - \epsilon_Y^2}/(1-\epsilon_Y)) \geq \arctan(2\sqrt{2\epsilon_Y})$. Using $\sin^2(\arctan(z)) = z^2/(1+z^2)$, we have $\sin^2(\hat{\theta}) \geq 8\epsilon_Y/(1+8\epsilon_Y) \geq \epsilon_Y/2$. Since θ is constrained to lie in $[0, \arcsin(\sqrt{\epsilon_Y/12})]$, it cannot equal $\hat{\theta}$. Thus maximizing (A.1) will require setting θ to one of the endpoints of the allowed region. In particular, the maximum value of (A.1) occurs when $\sin^2(\theta) = \epsilon_Y/12$. A similar argument proves that setting $|Y_{1,1}| = 1 - \epsilon_Y$ maximizes (A.1) as well. Now we calculate

$$|\langle\psi|Y|\psi\rangle| \leq (1 - \epsilon_Y) + 2\sqrt{\frac{\epsilon_Y}{12}}\sqrt{2\epsilon_Y - \epsilon_Y^2} + \frac{\epsilon_Y}{12} \leq 1 - \left(1 - \sqrt{\frac{2}{3}} - \frac{1}{12}\right)\epsilon_Y \leq 1 - \frac{\epsilon_Y}{10} \tag{A.2}$$

We have shown that for any ψ, either $\langle\psi|X|\psi\rangle \leq 1 - \epsilon_X\epsilon_Y/12$ or $\langle\psi|Y|\psi\rangle \leq 1 - \epsilon_Y/10$. We now use the triangle inequality to bound

$$\begin{aligned} \langle\psi|pX + (1-p)Y|\psi\rangle &\leq \max\left(p\left(1 - \frac{\epsilon_X\epsilon_Y}{12}\right) + (1-p), p + (1-p)\left(1 - \frac{\epsilon_Y}{10}\right)\right) \\ &\leq 1 - \frac{\epsilon_Y}{12}\min(p\epsilon_X, 1-p). \end{aligned} \tag{A.3}$$

Since this bound applies for all normalized $|\psi\rangle$, it must also upper-bound $\|pX + (1-p)Y\|$. Thus we obtain (30). The remaining steps of the Lemma are direct calculations. ∎

A.6 SODA Paper

The Quantum Schur and Clebsch-Gordan Transforms: I. Efficient Qudit Circuits

Dave Bacon[*] Isaac L. Chuang[†] and Aram W. Harrow[‡]

Abstract

We present an efficient family of quantum circuits for a fundamental primitive in quantum information theory, the Schur transform. The Schur transform on n d-dimensional quantum systems is a transform between a standard computational basis to a labelling related to the representation theory of the symmetric and unitary groups. If we desire to implement the Schur transform to an accuracy of ϵ, then our circuit construction uses a number of gates which is polynomial in n, d and $\log(\epsilon^{-1})$. The key tool in our construction is a $\mathrm{poly}(d, \log n, \log(\epsilon^{-1}))$ algorithm for the $\mathcal{U}_d$ Clebsch-Gordan transform. Our efficient circuit construction renders numerous protocols in quantum information theory computationally tractable and yields a new possible approach to quantum algorithms which is distinct from the standard paradigm of the quantum Fourier transform.

1 Introduction

The last decade has seen the development and expansion of a robust theory of quantum information[1] However despite much progress in understanding optimal rates for manipulating and transmitting quantum information, many results may not be of practical value, even if large-scale quantum computers and quantum communication networks could be built. This is because many of the optimal protocols assume unbounded (or at least exponential) quantum computational resources for each local party. An analogous situation arises classically, for example, in the theory of classical error correcting codes, where it can be difficult to reconcile the goals of efficient communication rates and computationally-efficient encoding and decoding.

While the goal of performing classical coding tasks in polynomial or even linear time has long been studied, quantum information theory results have typically ignored questions of efficiency. For example, random quantum coding results (such as [2, 3, 4, 5]) require an exponential number of bits to describe, and like classical random coding techniques, do not yield efficient algorithms. There are a few important exceptions. Some quantum coding tasks, such as Schumacher compression[6, 7], are essentially equivalent to classical circuits, and as such can be performed efficiently on a quantum computer by carefully modifying an efficient classical algorithm to run reversibly and to deal properly with ancilla systems[8]. Another example, which illustrates some of the challenges involved, is Ref. [9]'s efficient implementation of entanglement concentration[10]. Quantum key distribution[11] not only runs efficiently, but can be implemented with entirely, or almost entirely, single-qubit operations and classical computation. Finally, some randomized quantum code constructions have been given efficient constructions using classical derandomization techniques in [12].

In this paper we present an efficient family of quantum circuits for a transform used ubiquitously[13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23] in quantum information protocols: the Schur transform. Our efficient construction of the Schur transform adds to the above list a powerful new tool for finding algorithms that implement quantum communication tasks.

The Schur transform is a unitary transform on n d-dimensional quantum systems (n qudits). The basis change corresponding to the Schur transform goes from a standard computational basis on the n qudits to a labelling related to the representation theory of the symmetric and unitary groups; much like the Fourier transform, it thus transforms from a local to a more global, collective basis, which captures symmetries of the system. In this article we show how to efficiently implement the Schur transform as a quantum circuit. The size of the circuit we construct is polynomial in the number of qudits, n, the dimension of the individual quantum systems,

[*]Dept. of Computer Science and Engineering, Univ. of Washington, Seattle, WA, USA

[†]Dept. of Electrical Engineering and Computer Science and Dept. of Physics, Massachusetts Institute of Technology, Cambridge, MA, USA

[‡]Dept. of Computer Science, Univ. of Bristol, Bristol, U.K.

d, and the log of accuracy to which we implement the transform, $\log(\epsilon^{-1})$. Our efficient quantum circuit for the Schur transform makes possible efficient quantum circuits for numerous quantum information tasks: optimal spectrum estimation[13, 24], universal entanglement concentration[14], universal compression with optimal overflow exponent[15, 16], encoding into decoherence-free subsystems[18, 19, 20, 21], optimal hypothesis testing[17], and quantum and classical communication without shared reference frames[22]. The central role of the Schur transform in all of these protocols (as well as others like the quantum reverse Shannon theorem[25] where other aspects of the protocol remain inefficient) is due to the fact that the symmetries of independent and identically distributed quantum states are naturally treated by the representation theory of the symmetric and unitary groups.

The Schur transform is only defined up to a choice of the Schur basis, and a key technical component of our algorithm will be the selection of certain subgroup-adapted bases for the Schur basis. In particular we use the Gel'fand-Zetlin basis[26] and the Young-Yamanouchi basis (sometimes called Young's orthogonal basis)[27]. The usefulness of subgroup-adapted bases to quantum algorithms was recognized by Beals[28] and Moore and Russell[29] in their algorithms for efficient Fourier transforms on nonabelian finite groups. We will similarly exploit the recursive structure of subgroup-adapted bases to build efficient recursive algorithms for the Clebsch-Gordan and Schur transforms. However, we emphasize that the Schur transform is not equivalent to the Fourier transform over $\mathcal{S}_n$, $\mathcal{U}_d$ or any other group, while connections between such transforms and the Schur transform exist, and will be discussed in part II of this paper (see also Chapter 8 of [30]).

By choosing the Gel'fand-Zetlin basis and the Young-Yamanouchi basis, we are able to show that the Schur transform can be constructed from a cascade of Clebsch-Gordan transforms (in rough analogy to the iterative constructions of [29]). To implement the Clebsch-Gordan transform, we use the Wigner-Eckart theorem and the Gel'fand-Zetlin basis to derive a recursive expression for the d dimensional Clebsch-Gordan transform in terms of the $d-1$ dimensional Clebsch-Gordan transform and small, efficiently implementable, unitary transforms. (The efficiency of this reduction is reminiscent of the use of adapted diameter in [29], but not directly related since $\mathcal{U}_d/\mathcal{U}_{d-1}$ is not finite.) The resulting recursive circuit for the Clebsch-Gordan transform can achieve accuracy ϵ using $\mathrm{poly}(d, \log n, \log 1/\epsilon)$ gates in contrast with the $n^{O(d^2)}$ gates that would be required by a naive construction. The total size of our circuit for the Schur transform is thus $n \cdot \mathrm{poly}(d, \log n, \log 1/\epsilon)$.

The remainder of the paper is as follows. In Section 2 we define the Schur transform along with the necessary basic concepts from representation theory. In Section 3 we introduce the basis labelling scheme used in the Schur transformation using the concept of a subgroup-adapted basis. Once we have a concrete Schur basis defined, we describe the Clebsch-Gordan transform and explain how to use it to give an efficient circuit for the Schur transform in Sec. 4. Details on efficiently implementing the Clebsch-Gordan transform are in an appendix.

2 Representation theory and the Schur transform

Schur duality relates to the representation theory of the symmetric group on n elements, $\mathcal{S}_n$, and the group of $d \times d$ unitary matrices, $\mathcal{U}_d$. In this section we will state facts about these representations without proof; for more details the reader should consult [31] or the longer version of this paper ([30] and future work).

2.1 Representation theory: A representation $(\mathbf{r}, V)$ of a group G is a complex vector space V together with a homomorphism from G to $\mathrm{End}(V)$, i.e. a function $\mathbf{r} : G \to \mathrm{End}(V)$ such that $\mathbf{r}(g_1)\mathbf{r}(g_2) = \mathbf{r}(g_1 g_2)$. We say a representation $(\mathbf{r}, V)$ is irreducible (an *irrep*) if the only $\mathbf{r}$-invariant subspaces of V are the empty subspace $\{0\}$ and the entire space V. In order to apply our results to quantum computing, we consider only the case when V is complex and finite dimensional and $\mathbf{r}(g)$ is unitary for all $g \in G$. This way a unit vector in V can represent the state of a quantum system and group elements $g \in G$ correspond to unitary rotations $\mathbf{r}(g)$, which could in principle be performed by a quantum computer.

We now turn to the representations relevant to the Schur transform. Consider a system of n d-dimensional quantum systems: n qudits. Fix a standard basis $|i\rangle$, $i = 1 \ldots d$ for the state space of each qudit: $\mathbb{C}^d$. A basis for $(\mathbb{C}^d)^{\otimes n}$ (which we call the *computational basis*) is then $|i_1\rangle \otimes |i_2\rangle \otimes \cdots \otimes |i_n\rangle = |i_1, i_2, \ldots, i_n\rangle$ where $i_k = 1 \ldots d$. In terms of this basis, we can define the action of $\mathcal{S}_n$ as

$$\mathbf{P}(s)|i_1, i_2, \ldots, i_n\rangle = |i_{s^{-1}(1)}, i_{s^{-1}(2)}, \ldots, i_{s^{-1}(n)}\rangle$$

for $s \in \mathcal{S}_n$. The unitary group $\mathcal{U}_d$, on the other hand, acts on $(\mathbb{C}^d)^{\otimes n}$ according to the n-fold product action as

$$\mathbf{Q}(U)|i_1, i_2, \cdots, i_n\rangle = U|i_1\rangle \otimes U|i_2\rangle \otimes \cdots \otimes U|i_n\rangle$$

for any $U \in \mathcal{U}_d$.

Note that $\mathbf{P}$ and $\mathbf{Q}$ commute, which means they can be simultaneously decomposed into irreps. Schur duality (or Schur-Weyl duality)[31, 32] goes farther and describes the exact nature of this decomposition, but in order to state it, we will first need to specify the irreps of $\mathcal{S}_n$ and $\mathcal{U}_d$.

Let $\mathcal{I}_{d,n} = \{\lambda = (\lambda_1, \lambda_2, \ldots, \lambda_d) | \lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_d \geq 0 \text{ and } \sum_{i=1}^d \lambda_i = n\}$ denote partitions of n into $\leq d$ parts. We consider two partitions $(\lambda_1, \ldots, \lambda_d)$ and $(\lambda_1, \ldots, \lambda_d, 0, \ldots, 0)$ equivalent if they differ only by trailing zeroes; according to this principle, $\mathcal{I}_n := \mathcal{I}_{n,n}$ contains all the partitions of n. Partitions label irreps of $\mathcal{S}_n$ and $\mathcal{U}_d$ as follows: if we let d vary, then $\mathcal{I}_{d,n}$ labels irreps of $\mathcal{S}_n$ (or sometimes we use $\mathcal{I}_n := \mathcal{I}_{n,n}$), and if we let n vary, then $\mathcal{I}_{d,n}$ labels polynomial irreps of $\mathcal{U}_d$ (or sometimes we use $\mathbb{Z}_{++}^d := \cup_n \mathcal{I}_{d,n}$). Call these irreps $(\mathbf{p}_\lambda, \mathcal{P}_\lambda)$ and $(\mathbf{q}_\lambda^d, \mathcal{Q}_\lambda^d)$ respectively, for $\lambda \in \mathcal{I}_{d,n}$. We need the superscript d because the same partition λ can label different irreps for different $\mathcal{U}_d$; on the other hand the $\mathcal{S}_n$-irrep $\mathcal{P}_\lambda$ is uniquely labeled by λ since $n = \sum_i \lambda_i$.

For the case of n qudits, Schur duality states that there exists a basis (which we label $|\lambda\rangle|q_\lambda\rangle|p_\lambda\rangle_{\text{Sch}}$ and call the *Schur basis*) which simultaneously decomposes the action of $\mathbf{P}(s)$ and $\mathbf{Q}(U)$ into irreps:

$$\begin{aligned} \mathbf{Q}(U)|\lambda\rangle|q_\lambda\rangle|p_\lambda\rangle_{\text{Sch}} &= |\lambda\rangle(\mathbf{q}_\lambda^d(U)|q_\lambda\rangle)|p_\lambda\rangle_{\text{Sch}} \\ \mathbf{P}(s)|\lambda\rangle|q_\lambda\rangle|p_\lambda\rangle_{\text{Sch}} &= |\lambda\rangle|q_\lambda\rangle(\mathbf{p}_\lambda(s)|p_\lambda\rangle)_{\text{Sch}} \end{aligned}$$

and that the common representation space $(\mathbb{C}^d)^{\otimes n}$ decomposes as

$$(\mathbb{C}^d)^{\otimes n} \overset{\mathcal{U}_d \times \mathcal{S}_n}{\cong} \bigoplus_{\lambda \in \mathcal{I}_{d,n}} \mathcal{Q}_\lambda^d \otimes \mathcal{P}_\lambda. \tag{2.1}$$

The Schur basis can be expressed as superpositions over the standard computational basis states $|i_1, i_2, \ldots, i_n\rangle$ as

$$|\lambda, q_\lambda, p_\lambda\rangle_{\text{Sch}} = \sum_{i_1, i_2, \ldots, i_n} \left[\mathbf{U}_{\text{Sch}}^\dagger\right]_{i_1, i_2, \ldots, i_n}^{\lambda, q_\lambda, p_\lambda} |i_1 i_2 \ldots i_n\rangle, \tag{2.2}$$

where $\mathbf{U}_{\text{Sch}}$ is the unitary transformation implementing the isomorphism in (2.1). Thus, for any $U \in \mathcal{U}_d$ and any $s \in \mathcal{S}_n$,

$$\mathbf{U}_{\text{Sch}}^\dagger \mathbf{Q}(U)\mathbf{P}(s)\mathbf{U}_{\text{Sch}} = \sum_{\lambda \in \mathcal{I}_{d,n}} |\lambda\rangle\langle\lambda| \otimes \mathbf{q}_\lambda^d(U) \otimes \mathbf{p}_\lambda(s). \tag{2.3}$$

If we now think of $\mathbf{U}_{\text{Sch}}$ as a quantum circuit, it will map the Schur basis state $|\lambda, q_\lambda, p_\lambda\rangle_{\text{Sch}}$ to the computational basis state $|\lambda, q_\lambda, p_\lambda\rangle$ with λ, q_λ, and p_λ expressed as bit strings. The dimensions of the irreps $\mathbf{p}_\lambda$ and $\mathbf{q}_\lambda^d$ vary with λ, so we will need to pad the $|q_\lambda, p_\lambda\rangle$ registers when they are expressed as bit strings. We will label the padded basis as $|\lambda\rangle|q\rangle|p\rangle$, explicitly dropping the λ dependence. Later in the paper we will show how to do this padding efficiently with only a logarithmic additive spatial overhead. We will refer to the transform from the computational basis $|i_1, i_2, \ldots, i_n\rangle$ to the basis of three strings $|\lambda\rangle|q\rangle|p\rangle$ as the Schur transform. The Schur transform is shown schematically in Fig. 1. Notice that just as the standard computational basis $|i\rangle$ is arbitrary up to a unitary transform, the bases for $\mathcal{Q}_\lambda^d$ and $\mathcal{P}_\lambda$ are also both arbitrary up to a unitary transform, though we will later choose particular bases for $\mathcal{Q}_\lambda^d$ and $\mathcal{P}_\lambda$.

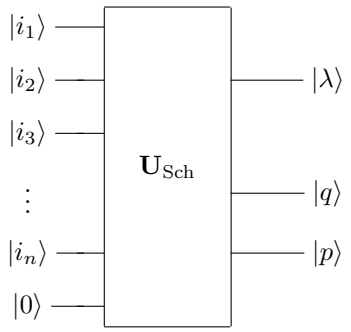

Figure 1: The Schur transform. Notice how the direct sum over λ in (2.1) becomes a tensor product between the $|\lambda\rangle$ register and the $|q\rangle$ and $|p\rangle$ registers. Since the number of qubits needed for $|q\rangle$ and $|p\rangle$ vary with λ, we need slightly more spatial resources, which are here denoted by the ancilla input $|0\rangle$.

2.2 Applications of the Schur Transform The Schur transform is useful in a surprisingly large number of quantum information protocols. Here we, review these applications, with particular attention to the use of the Schur transform circuit in each protocol. We emphasize again that our construction of the Schur transform simultaneously makes all of these tasks computationally efficient.

2.2.1 Spectrum and state estimation Suppose we are given many copies of an unknown mixed quantum state, $\boldsymbol{\rho}^{\otimes n}$ and wish to estimate the spectrum of $\boldsymbol{\rho}$. An asymptotically optimal estimate (in the sense of the error exponent of large deviations) for the spectrum of $\boldsymbol{\rho}$ can be obtained by applying the Schur transform, measuring λ and taking the spectrum estimate to be $(\lambda_1/n, \ldots, \lambda_d/n)$[13, 24]. Thus an efficient implementation of the Schur transform will efficiently implement the spectrum estimating pro-

tocol (note that it is efficient in d, not in $\log(d)$). Estimating $\boldsymbol{\rho}$ reduces to measuring $|\lambda\rangle$ and $|q\rangle$, although an optimal estimator is known only for the case of $d = 2$[33]. Further, optimal quantum hypothesis testing can be obtained by a similar, but more complicated, protocol[17, 34]. The only one of these known to have an implementation not based on the Schur transform is spectrum estimation, which can be performed using the Fourier transform on $\mathcal{S}_n$ with a technique known as "generalized phase estimation" or as "the phase kickback trick"[35].

2.2.2 Universal distortion-free entanglement concentration Let $|\psi\rangle_{AB}$ be a bipartite partially entangled state shared between two parties, A and B. Suppose we are given many copies of $|\psi\rangle_{AB}$ and we want to transform these states into copies of a maximally entangled state using only local operations and classical communication. Further, suppose that we wish this protocol to be universal, meaning it works when neither A nor B know the state $|\psi\rangle_{AB}$, unlike the original entanglement concentration protocol in [10]. Universal distortion-free (meaning zero error, but the entanglement yield is a random variable) entanglement concentration can be performed[14] by both parties performing Schur transforms on their n halves of $|\psi\rangle_{AB}$, measuring their $|\lambda\rangle$, discarding $|q\rangle$ and retaining $|p\rangle$. The two parties will now share a maximally entangled state of varying dimension depending on which λ was measured. This dimension is $2^{n(H\pm o(1))}$ with probability $1 - o(1)$, where H is the entropy of one of the parties' reduced mixed states, and in fact this protocol is optimal even in a non-asymptotic sense (i.e. for each finite value of n)[23]. While universal entanglement concentration can also be efficiently achieved without the Schur transform by using some of the copies to perform tomography, this introduces $\Omega(n^{-1/2})$ errors.

2.2.3 Universal Compression with Optimal Overflow Exponent Measuring $|\lambda\rangle$ weakly so as to cause little disturbance, together with appropriate relabeling, comprises a universal compression algorithm with optimal overflow exponent (rate of decrease of the probability that the algorithm will output a state that is much too large)[15, 16]. In the fixed-rate setting (where the entropy of the source is promised to be less than the rate), performing a projective measurement on λ will compress while incurring the optimal $\exp(-\Omega(n))$ error rate[36]. The only known efficient procedures not relying on the Schur transform have worse overflow exponents[37, 38] and introduce $\Omega(n^{-1/4})$ errors even in the fixed-rate setting.

2.2.4 Encoding and decoding into decoherence-free subsystems Further applications of the Schur transform include encoding into decoherence-free subsystems[18, 19, 20, 21]. Decoherence-free subsystems are subspaces of a system's Hilbert space which are immune to decoherence due to a symmetry of the system-environment interaction. For the case where the environment couples identically to all systems, information can be protected from decoherence by encoding into the $|p_\lambda\rangle$ basis. We can use the inverse Schur transform (which, as a circuit can be implemented by reversing the order of all gate elements and replacing them with their inverses) to perform this encoding: simply feed in the appropriate $|\lambda\rangle$ with the state to be encoded into the $|p\rangle$ register and any state into the $|q\rangle$ register into the inverse Schur transform. Decoding can similarly be performed using the Schur transform. Previously no efficient algorithms for encoding or decoding were known.

2.2.5 Communication without a shared reference frame An application of the concepts of decoherence-free subsystems comes about when two parties wish to communicate (in either a classical or quantum manner) when the parties do not share a reference frame. The effect of not sharing a reference frame is the same as the effect of collective decoherence (the same random unitary rotation has been applied to each subsystem). Thus encoding information into the $|p\rangle$ register will allow $n - O(\log n)$ qudits to be sent noiselessly with n uses of the channel in spite of the fact that the two parties do not share a reference frame[22]. Just as with decoherence-free subsystems, this encoding and decoding can be done with the Schur transform. Previously, the best known efficient procedure used m out of the n channel uses for tomography, resulting in $\Omega(1/m)$ overall error.

3 Subgroup adapted bases and the Schur basis

In the last section, we defined the Schur transform in a way that left the basis almost completely arbitrary. To construct a quantum circuit for the Schur transform, we need to explicitly specify the Schur basis. Since we want the Schur basis to be of the form $|\lambda, q, p\rangle$, this reduces to specifying orthonormal bases for $\mathcal{Q}_\lambda^d$ and $\mathcal{P}_\lambda$, which we will call Q_λ^d and P_λ, respectively. We will choose Q_λ^d and P_λ to both be a type of basis known as a *subgroup-adapted* basis, an idea first introduced to quantum information in [28, 29].

The key idea is to examine how an irrep $(\mathbf{r}, V)$ of a group $\mathcal{G}$ decomposes into $\mathcal{H}$-irreps when restricted

to $\mathcal{H}$ (denoted $(\mathbf{r}|_{\mathcal{H}}, V{\downarrow_{\mathcal{H}}})$). This behavior is called *branching* and when no irrep of $\mathcal{H}$ ever appears more than once in an irrep of $\mathcal{G}$, we call the branching *multiplicity-free*. Now consider a tower of groups $\mathcal{G} = \mathcal{G}_1 \supset \mathcal{G}_2 \supset \cdots \supset \mathcal{G}_{k-1} \supset \mathcal{G}_k = \{e\}$ where $\{e\}$ is the trivial subgroup consisting of only the identity element. We call this a *canonical* tower when the branching for each $\mathcal{G}_{i+1} \subset \mathcal{G}_i$ is multiplicity-free. Finally we can define a *subgroup-adapted basis* (unique up to an arbitrary choice of phase) in which basis vectors have the form $|\alpha_k, \ldots, \alpha_2\rangle$, where each $\alpha_i \in \hat{\mathcal{G}}_i$ and α_{i+1} appears in the decomposition of $V_{\alpha_i}{\downarrow_{\mathcal{G}_{i+1}}}$.

We now need only to specify canonical towers of subgroups for $\mathcal{U}_d$ and $\mathcal{S}_n$, which will give rise to subgroup-adapted bases for the irreps $\mathcal{Q}_\lambda^d$ and $\mathcal{P}_\lambda$, known as the Gel'fand-Zetlin basis[26] and the Young-Yamanouchi basis (or sometimes Young's orthogonal basis[27]), respectively.

The Gel'fand-Zetlin basis for $\mathcal{Q}_\lambda^d$ — For $\mathcal{U}_d$, it turns out that the chain of subgroups $\{1\} = \mathcal{U}_0 \subset \mathcal{U}_1 \subset \ldots \subset \mathcal{U}_{d-1} \subset \mathcal{U}_d$ is a canonical tower, where we have $\mathcal{U}_d - 1$ embedded in $\mathcal{U}_d$ by $\mathcal{U}_{d-1} := \{u \in \mathcal{U}_d : u|d\rangle = |d\rangle\}$. Since the branching from $\mathcal{U}_d$ to $\mathcal{U}_{d-1}$ is multiplicity-free, we obtain a subgroup-adapted basis Q_λ^d, which is known as the *Gel'fand-Zetlin* (GZ) basis. Our only free choice in a GZ basis is the initial choice of basis $|1\rangle, \ldots, |d\rangle$ for $\mathbb{C}^d$ which determines the canonical tower of subgroups $\mathcal{U}_1 \subset \ldots \subset \mathcal{U}_d$. Once we have chosen this basis, specifying Q_λ^d reduces to knowing which irreps $\mathcal{Q}_\mu^{d-1}$ appear in the decomposition of $\mathcal{Q}_\lambda^d{\downarrow_{\mathcal{U}_{d-1}}}$. Recall that the irreps of $\mathcal{U}_d$ are labeled by elements of $\mathcal{I}_{d,n}$ with n arbitrary. This set can be denoted by $\mathbb{Z}_{++}^d := \cup_n \mathcal{I}_{d,n} = \{\lambda \in \mathbb{Z}^d : \lambda_1 \geq \ldots \geq \lambda_d \geq 0\}$. For $\mu \in \mathbb{Z}_{++}^{d-1}, \lambda \in \mathbb{Z}_{++}^d$, we say that μ *interlaces* λ and write $\mu \precsim \lambda$ whenever $\lambda_1 \geq \mu_1 \geq \lambda_2 \ldots \geq \lambda_{d-1} \geq \mu_{d-1} \geq \lambda_d$. In terms of Young diagrams, this means that μ is a valid partition (i.e. a nonnegative, nonincreasing sequence) obtained from removing zero or one boxes from each column of λ. Thus a basis vector in Q_λ^d corresponds to a sequence of partitions $q = (q_d = \lambda, \ldots, q_1)$ such that $q_1 \precsim q_2 \precsim \ldots \precsim q_d$ and $q_j \in \mathbb{Z}_{++}^j$ for $j = 1, \ldots, d$.

In order to work with the Gel'fand-Zetlin basis vectors on a quantum computer, we will need an efficient method to write them down. If d is small compared to n (as in many information theory applications), we can write an element of $\mathcal{I}_{d,n}$ with $\lceil d \log(n+1) \rceil$ bits, since it consists of d integers between 0 and n. A Gel'fand-Zetlin basis vector then requires no more than $\lceil d^2 \log(n+1) \rceil$ bits, since it can be expressed as d partitions of integers no greater than n into $\leq d$ parts. Unless otherwise specified, our algorithms will use this encoding of the GZ basis vectors. However, another encoding, known as semi-standard Young tableaux, can represent a GZ basis vector using $n\lceil \log d \rceil$ bits. To encode $q = (q_d, \ldots, q_1)$, we fill the Young diagram of λ with n integers from $\{1, \ldots, d\}$ in a pattern that is nonincreasing from left to right, strictly increasing from top to bottom, and such that for each $d' < d$, removing all boxes with integers larger than d' leaves the Young diagram of $q_{d'}$.

The Young-Yamanouchi basis for $\mathcal{P}_\lambda$ — The situation for $\mathcal{S}_n$ is quite similar. Our chain of subgroups is $\{e\} = \mathcal{S}_1 \subset \mathcal{S}_2 \subset \ldots \subset \mathcal{S}_n$, where for $m < n$ we define $\mathcal{S}_m \subset \mathcal{S}_n$ to be the permutations in $\mathcal{S}_n$ which leave the last $n-m$ elements fixed. Recall that the irreps of $\mathcal{S}_n$ can be labeled by $\mathcal{I}_n = \mathcal{I}_{n,n}$: the partitions of n into $\leq n$ parts.

Again, the branching from $\mathcal{S}_n$ to $\mathcal{S}_{n-1}$ is multiplicity-free, so to determine an orthonormal basis P_λ for the space $\mathcal{P}_\lambda$ we need only know which irreps occur in the decomposition of $\mathcal{P}_\lambda{\downarrow_{\mathcal{S}_{n-1}}}$. It turns out that the branching rule is given by finding all ways to remove one box from λ while leaving a valid partition. Denote the set of such partitions $\lambda - \square$. Formally, $\lambda - \square := \mathcal{I}_n \cap \{\lambda - e_j : j = 1, \ldots, n\}$, where e_j is the unit vector in $\mathbb{Z}^n$ with a one in the j^{th} position and zeroes elsewhere. Thus, the general branching rule is

$$\mathcal{P}_\lambda{\downarrow_{\mathcal{S}_{n-1}}} \overset{\mathcal{S}_{n-1}}{\cong} \bigoplus_{\lambda' \in \lambda - \square} \mathcal{P}_{\lambda'}. \tag{3.4}$$

This chain can be concisely labelled in $\lceil \log n! \rceil$ bits by writing the number j in the box of the Young frame that is removed when restricting from $\mathcal{S}_j$ to $\mathcal{S}_{j-1}$. However, for applications such as data compression[15, 16] we will need an encoding which gives us closer to the optimal $\lceil \log P_\lambda \rceil$ bits. First we note an exact (and efficiently computable) expression[31, 27] for $|P_\lambda| = \dim \mathcal{P}_\lambda$:

$$\dim \mathcal{P}_\lambda = \frac{\prod_{1 \leq i < j \leq d} (\lambda_i - \lambda_j + j - i)}{\lambda_1 + d - 1! \lambda_2 + d - 2! \cdots \lambda_d!} n!\,. \tag{3.5}$$

Now we would like to efficiently and reversibly map an element of P_λ (thought of as a chain of partitions $p = (p_n = \lambda, \ldots, p_1 = (1)) \in P_\lambda$, with $p_j \in p_{j+1} - \square$) to an integer in $[|P_\lambda|] := \{1, \ldots, |P_\lambda|\}$. We will construct this bijection $f_n : P_\lambda \to [|P_\lambda|]$ by defining an ordering on P_λ and setting $f_n(p) := |\{p' \in P_\lambda : p' \leq p\}|$. First fix an arbitrary, but easily computable, (total) ordering on partitions in $\mathcal{I}_n$ for each n; for example, lexicographical order. This induces an ordering on P_λ if we rank a basis vector

$p \in P_\lambda$ first according to p_{n-1}, using the order on partitions we have chosen, then according to p_{n-2} and so on. We skip p_n, since it is always equal to λ. In other words, for $p, p' \in P_\lambda$, $p > p'$ if $p_{n-1} > p'_{n-1}$ *or* $p_{n-1} = p'_{n-1}$ and $p_{n-2} > p'_{n-2}$ *or* $p_{n-1} = p'_{n-1}$, $p_{n-2} = p'_{n-2}$ and $p_{n-3} > p'_{n-3}$, and so on. Thus $f_n : P_\lambda \to [|P_\lambda|]$ can be easily verified to be

(3.6)
$$f_n(p) = f_n(p_1, \ldots, p_n) := 1 + \sum_{k=2}^{n} \sum_{\substack{\lambda \in p_k - \square \\ \lambda < p_{k-1}}} \dim \mathcal{P}_\lambda.$$

Thus f_n is an injective map from P_λ to $[|P_\lambda|]$ that is computable in time polynomial in n. Unfortunately, similar techniques for Q^d_λ take time $\Omega(n^d)$.

4 The Clebsch-Gordan transform and efficient circuits for the Schur transform

In this section, we describe an efficient circuit for the Schur transform $\mathbf{U}_{\mathrm{Sch}}$. The key building block will be the $\mathcal{U}_d$ Clebsch-Gordan (CG) transform, which decomposes a Kronecker product of $\mathcal{U}_d$-irreps $\mathcal{Q}^d_\mu \otimes \mathcal{Q}^d_\nu$ into a direct sum of other $\mathcal{U}_d$-irreps (described in Sec. 4.1). Then we will give efficient recursive constructions for the Schur transform and the CG transform. In Sec. 4.2, we will show how the Schur transform on $(\mathbb{C}^d)^{\otimes n}$ reduces to a Schur transform on $(\mathbb{C}^d)^{\otimes n-1}$ and a CG transform. Then in Sec. 4.3, we will show how the $\mathcal{U}_d$ CG transform reduces to a $\mathcal{U}_{d-1}$ CG transform and a efficiently-calculable $d \times d$ unitary matrix known as a reduced Wigner transform. Together this will yield poly-time algorithms for the CG and Schur transforms.

4.1 The Clebsch-Gordan Series and Transform The Clebsch-Gordan decomposition describes the reduction of a tensor product representation into irreps. We specialize to the case when the group is $\mathcal{U}_d$, one of the irreps is $\mathcal{Q}^d_\mu$ and the other is $\mathcal{Q}^d_{(1)} = \mathbb{C}^d$, the d-dimensional defining irrep of $\mathcal{U}_d$. The representation $\mathcal{Q}^d_\lambda \otimes \mathcal{Q}^d_{(1)}$ is generally reducible and decomposes as

$$\mathcal{Q}^d_\lambda \otimes \mathcal{Q}^d_{(1)} \overset{\mathcal{U}_d}{\cong} \bigoplus_{\lambda' \in \lambda + \square} \mathcal{Q}^d_{\lambda'}, \tag{4.7}$$

Here $\lambda + \square = \{\lambda + e_j : j \in [d]\} \cap \mathbb{Z}^d_{++}$ is the "add a single box" prescription for tensoring in a defining representation of $\mathcal{U}_d$: we add a single box to a Young diagram and if the new Young diagram is a valid Young diagram (i.e. corresponds to a valid partition), then this irrep appears in the Clebsch-Gordan series. By Schur duality, this statement is equivalent to the "remove a box" $\mathcal{S}_n \supset \mathcal{S}_{n-1}$ branching rule stated in Sec. 3.

We now seek to define the CG transform as a quantum circuit. One of the input irreps will always be the defining irrep, but we allow the other irrep to be specified by a quantum input. If we define $\mathbf{U}^{\lambda,(1)}_{\mathrm{CG}}$ to be the transform relating the two sides of (4.7), then the CG transform we are interested in is

$$\mathbf{U}_{\mathrm{CG}} = \sum_{\lambda \in \mathbb{Z}^d_{++}} |\lambda\rangle\langle\lambda| \otimes \mathbf{U}^{\lambda,(1)}_{\mathrm{CG}}. \tag{4.8}$$

This takes as input a state of the form $|\lambda\rangle|q\rangle|i\rangle$, for $\lambda \in \mathbb{Z}^d_{++}$, $|q\rangle \in Q^d_\lambda$ and $i \in [d]$. The output is a superposition over vectors $|\lambda\rangle|\lambda'\rangle|q'\rangle$, where $\lambda' \in \lambda + \square$ and $|q'\rangle \in Q^d_{\lambda'}$. Equivalently, we could output $|\lambda\rangle|j\rangle|q'\rangle$ or $|j\rangle|\lambda'\rangle|q'\rangle$ (where $\lambda' = \lambda + e_j$) since (λ, λ'), (λ, j) and (λ', j) are all trivially related via reversible classical circuits.

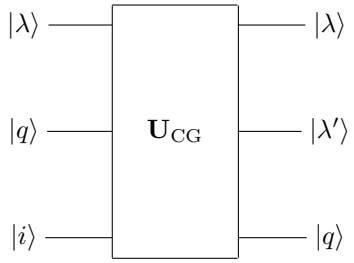

Figure 2: Schematic of the Clebsch-Gordan transform. Equivalently, we could replace either the λ output or the λ' output with j such that $\lambda' = \lambda + e_j$.

4.2 Constructing the Schur Transform from Clebsch-Gordan Transforms We now describe how to construct the Schur transform out of a series of Clebsch-Gordan transforms. Begin by decomposing $(\mathbb{C}^d)^{\otimes n}$ in two different ways. First, we Schur-decompose the first $n-1$ qudits,

$$(\mathbb{C}^d)^{\otimes n-1} \otimes \mathbb{C}^d \overset{\mathcal{U}_d \times \mathcal{S}_{n-1}}{\cong} \bigoplus_{\lambda \in \mathcal{I}_{d,n-1}} \mathcal{Q}^d_\lambda \otimes \mathcal{P}_\lambda \otimes \mathbb{C}^d \tag{4.9}$$

Next, combine $\mathcal{Q}^d_\lambda$ and $\mathbb{C}^d$ using the CG transform (4.7), then rearrange terms to obtain

(4.10)
$$(\mathbb{C}^d)^{\otimes n} \cong \bigoplus_{\substack{\lambda \in \mathcal{I}_{d,n-1} \\ \lambda' \in \lambda + \square}} \mathcal{Q}^d_{\lambda'} \otimes \mathcal{P}_\lambda = \bigoplus_{\lambda' \in \mathcal{I}_{d,n}} \mathcal{Q}^d_{\lambda'} \otimes \left(\bigoplus_{\lambda \in \lambda' - \square} \mathcal{P}_\lambda \right).$$

On the other hand, we have $(\mathbb{C}^d)^{\otimes n} \cong \bigoplus_{\lambda' \in \mathcal{I}_{d,n}} \mathcal{Q}^d_{\lambda'} \otimes \mathcal{P}_{\lambda'}$ from (2.1). These two decompositions can be

equated using (3.4), the branching rule for $\mathcal{S}_{n-1} \subset \mathcal{S}_n$.

Now we will turn the representation-theoretic arguments of the last paragraph into an algorithm. The Schur transform on $(\mathbb{C}^d)^{\otimes n}$ starts with inputs of the form $|i_1, \ldots, i_n\rangle \in (\mathbb{C}^d)^{\otimes n}$ and is implemented as follows:

1. Perform the Schur transform on the first $n-1$ registers (corresponding to (4.9)) to obtain $\left|\lambda^{(n-1)}\right\rangle \left|q^{(n-1)}\right\rangle \left|p^{(n-1)}\right\rangle |i_n\rangle$.
2. Perform the CG transform (as in (4.10)) on $\left|\lambda^{(n-1)}\right\rangle \left|q^{(n-1)}\right\rangle |i_n\rangle$ to obtain $\left|\lambda^{(n-1)}\right\rangle \left|\lambda^{(n)}\right\rangle \left|q^{(n)}\right\rangle$.
3. Set $|\lambda\rangle = \left|\lambda^{(n)}\right\rangle$ and $|q\rangle = \left|q^{(n)}\right\rangle$. Concatenate λ^{n-1} and $p^{(n-1)}$ to form the Young-Yamanouchi basis element $|p\rangle = \left|\lambda^{(n-1)}\right\rangle \left|p^{(n-1)}\right\rangle \in \mathcal{P}_\lambda$.

The base case of this recursion is simply the trivial $n = 1$ relabelling corresponding to $\mathcal{Q}^d_{(1)} \cong \mathbb{C}^d$ and $\mathcal{P}_{(1)} \cong \mathbb{C}$.

We can also express this algorithm for the Schur transform without the need for recursion. On input $|i_1, \ldots, i_n\rangle \in (\mathbb{C}^d)^{\otimes n}$, we combine each of $|i_1\rangle, \ldots, |i_n\rangle$ using the CG transform, one at a time. We start by inputting $\left|\lambda^{(1)}\right\rangle = |(1)\rangle$, $|i_1\rangle$ and $|i_2\rangle$ into $\mathbf{U}_{\mathrm{CG}}$ which outputs $\left|\lambda^{(1)}\right\rangle$ and a superposition of different values of $\left|\lambda^{(2)}\right\rangle$ and $|q_2\rangle$. Here $\lambda^{(2)}$ can be either $(2,0)$ or $(1,1)$ and $|q_2\rangle \in Q^d_{\lambda^{(2)}}$. Continuing, we apply $\mathbf{U}_{\mathrm{CG}}$ to $\left|\lambda^{(2)}\right\rangle |q_2\rangle |i_3\rangle$, and output a superposition of vectors of the form $\left|\lambda^{(2)}\right\rangle \left|\lambda^{(3)}\right\rangle |q_3\rangle$, with $\lambda^{(3)} \in \mathcal{I}_{d,3}$ and $|q_3\rangle \in Q^d_{\lambda^{(3)}}$. Each time we are combining an arbitrary irrep $\lambda^{(k)}$ and an associated basis vector $|q_k\rangle \in Q^d_{\lambda^{(k)}}$, together with a vector from the defining irrep $|i_{k+1}\rangle$. This is repeated for $k = 1, \ldots, n-1$ and the resulting circuit is depicted in Fig. 3.

We are left with a superposition of states of the form $\left|\lambda^{(1)}, \ldots, \lambda^{(n)}\right\rangle |q_n\rangle$, where $|q_n\rangle \in Q^d_{\lambda^{(n)}}$, $\lambda^{(k)} \in \mathcal{I}_{d,k}$ and each $\lambda^{(k)}$ is obtained by adding a single box to $\lambda^{(k-1)}$; i.e. $\lambda^{(k)} = \lambda^{(k-1)} + e_{j_k}$ for some $j_k \in [d]$. Again we relabel $\lambda = \lambda^{(n)}$, $|q\rangle = |q_n\rangle$ and $|p\rangle = \left|\lambda^{(1)}, \ldots, \lambda^{(n-1)}\right\rangle$, where we use the fact that our basis for $\mathcal{P}_\lambda$ is adapted to the subgroup tower $\mathcal{S}_1 \subset \ldots \subset \mathcal{S}_n$. Thus, we obtain the desired $|\lambda\rangle|q\rangle|p\rangle$. Finally, we can optionally compress the $|p\rangle$ register to $\lceil \log |P_\lambda| \rceil$ qubits using the techniques in Sec. 3, as is required for applications such as data compression and entanglement concentration, described in Sec. 2.2.

In the next section we will show that a single $\mathcal{U}_d$ transform can be performed to accuracy ϵ in time $\mathrm{poly}(\log n, d, \log 1/\epsilon)$. Thus the entire Schur transform requires time $n \cdot \mathrm{poly}(\log n, d, \log 1/\epsilon)$ plus an optional $\mathrm{poly}(n)$ to compress the $|p\rangle$ register.

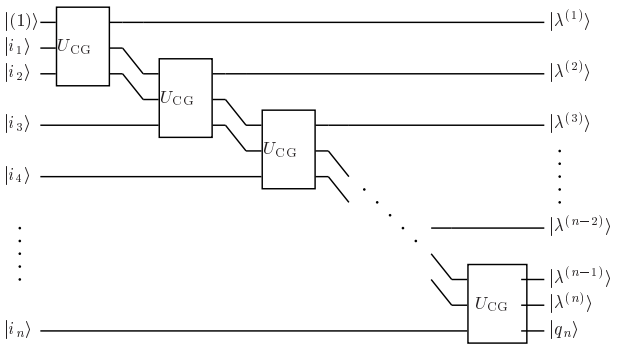

Figure 3: Cascading Clebsch-Gordan transforms to produce the Schur transform. Not shown are any ancilla inputs to the Clebsch-Gordan transforms. The structure of inputs and outputs of the Clebsch-Gordan transforms are the same as in Fig. 2.

Remark: If $d \gg n$, then a slight modification of the above algorithm can run in time $\mathrm{poly}(n, \log d, \log 1/\epsilon)$. First note that given oracle access to $u \in \mathcal{S}_d \subset \mathcal{U}_d$, we need only time $O(n \log d)$ to apply $\mathcal{Q}(u)$ or $\mathbf{q}^d_\lambda(u)$, assuming our GZ basis is written as semistandard Young tableaux. The algorithm first calculates a sorted list $a_1, \ldots, a_m$ of the $m \leq n$ distinct symbols occuring in $i_1, \ldots, i_n$. Next we will map $|a_j\rangle$ to $|j\rangle$ for each occurence of a_j in the input string. Now we can apply the Schur transform to $(\mathbb{C}^m)^{\otimes n}$, which runs in time $\mathrm{poly}(n, \log 1/\epsilon)$. Finally we apply the inverse map $|j\rangle \to |a_j\rangle$ to $|q\rangle$, and use $|q\rangle$ to uncompute $a_1, \ldots, a_m$.

4.3 Efficient circuits for the Clebsch-Gordan transform We now describe how to construct the $\mathcal{U}_d$ CG transform described in (4.7) and (4.8). The key to an efficient algorithm will be to reduce the $\mathcal{U}_d$ CG transform to a $\mathcal{U}_{d-1}$ transform. As with the Schur transform, our strategy will be to decompose a representation in two different ways, and then to find an operational method of equating them. This time we will decompose the $\mathcal{U}_d$-representation $\mathcal{Q}^d_\lambda \otimes \mathbb{C}^d$, which is input to the CG transform, both by using the $\mathcal{U}_{d-1} \subset \mathcal{U}_d$ branching rules and by applying the CG transform.

First, we might apply the $\mathcal{U}_d$ CG transform and then $\mathcal{U}_{d-1} \subset \mathcal{U}_d$ branching to obtain

$$\bigoplus_{\lambda' \in \lambda + \square} \mathcal{Q}^d_{\lambda'} \overset{\mathcal{U}_{d-1}}{\cong} \bigoplus_{\substack{\lambda' \in \lambda + \square \\ \mu' \precsim \lambda'}} \mathcal{Q}^{d-1}_{\mu'}. \tag{4.11}$$

Here we can write $\lambda' = \lambda + e_j$ for $j \in [d]$, and can map between $|\lambda, \lambda'\rangle$ and $|\lambda, j\rangle$.

On the other hand, if we restrict to $\mathcal{U}_{d-1}$ first and then apply the $\mathcal{U}_{d-1}$ CG, we obtain

$$\mathcal{Q}^d_\lambda \otimes \mathbb{C}^d \overset{\mathcal{U}_{d-1}}{\cong} \bigoplus_{\mu \precsim \lambda} \mathcal{Q}^{d-1}_\mu \otimes \left(\mathbb{C} \oplus \mathbb{C}^{d-1}\right) \tag{4.12a}$$

$$\overset{\mathcal{U}_{d-1}}{\cong} \bigoplus_{\mu \precsim \lambda} \mathcal{Q}^{d-1}_\mu \oplus \mathcal{Q}^{d-1}_\mu \otimes \mathbb{C}^{d-1} \tag{4.12b}$$

$$\overset{\mathcal{U}_{d-1}}{\cong} \bigoplus_{\mu \precsim \lambda} \left[\mathcal{Q}^{d-1}_\mu \oplus \bigoplus_{\mu' \in \mu + \square} \mathcal{Q}^{d-1}_{\mu'} \right] \tag{4.12c}$$

$$\overset{\mathcal{U}_{d-1}}{\cong} \bigoplus_{\substack{\mu \precsim \lambda \\ \mu' \in \{\mu\} \cup \mu + \square}} \mathcal{Q}^{d-1}_{\mu'}. \tag{4.12d}$$

In this last step, we can write $\mu' = \mu + e_{j'}$, where $j' \in \{0, 1, \ldots, d-1\}$ and we have defined $e_0 = 0$; moreover, we can readily map $|\mu, \mu'\rangle$ to and from $|\mu', j'\rangle$.

We want to perform $\mathbf{U}^{\lambda,(1)}_{\mathrm{CG}}$, which is a $\mathcal{U}_d$-invariant operator that maps the LHS of (4.12a) to the LHS of (4.11). Since all the maps in (4.12) and (4.11) are $\mathcal{U}_{d-1}$-invariant, if we pass $\mathbf{U}^{\lambda,(1)}_{\mathrm{CG}}$ through each of these isomorphisms we obtain a $\mathcal{U}_{d-1}$-invariant map from (4.12d) to the RHS of (4.11), which we call $\hat{T}^\lambda$. Next, the fact that $\hat{T}^\lambda$ commutes with the action of $\mathcal{U}_{d-1}$ means that $\hat{T}^\lambda$ must act as the identity on each subsystem $\mathcal{Q}^{d-1}_{\mu'}$ and nontrivially only on the multiplicity spaces of $\mathcal{Q}^{d-1}_{\mu'}$, conditioned on μ'. These multiplicity spaces are d-dimensional in both (4.12d) and (4.11); in the former they are labeled by $j' \in \{0, \ldots, d-1\}$ such that $\mu' = \mu + e_{j'}$, while in the latter they are labeled by $j \in [d]$ such that $\lambda' = \lambda + e_j$.

Let $\hat{T}^{\lambda,\mu'}$ denote the restriction of $\hat{T}^\lambda$ to the multiplicity space of $\mathcal{Q}^{d-1}_{\mu'}$. With a slight abuse of notation, we can write

$$\hat{T}^\lambda = \sum_\mu |\mu\rangle\langle\mu| \otimes I_{\mathcal{Q}^{d-1}_{\mu'}} \otimes \hat{T}^{\lambda,\mu'}. \tag{4.13}$$

We call $\hat{T}^{\lambda,\mu'}$ a *reduced Wigner operator*. Its matrix elements, given by

$$\hat{T}^{\lambda,\mu'} = \sum_{j=1}^{d} \sum_{j'=0}^{d-1} \hat{T}^{\lambda,\mu'}_{j,j'} |j\rangle\langle j'|, \tag{4.14}$$

are called *reduced Wigner coefficients*. At the end of this section, we will show how the $\hat{T}^{\lambda,\mu'}_{j,j'}$ can efficiently calculated, and thus $\hat{T}^{\lambda,\mu'}$ and $\hat{T}^\lambda$ can be efficiently implemented.

First we show how performing $\hat{T}^{\lambda,\mu'}$ will allow us to implement $\mathbf{U}^{\lambda,(1)}_{\mathrm{CG}}$. Since we constructed $\hat{T}^\lambda$ by composing $\mathbf{U}^{\lambda,(1)}_{\mathrm{CG}}$ with the isomorphisms in (4.12) and (4.11), our algorithm for $\mathbf{U}^{\lambda,(1)}_{\mathrm{CG}}$ will simply be to apply the isomorphisms in (4.12), apply $\hat{T}^\lambda$ (or equivalently, $\hat{T}^{\lambda,\mu'}$ conditioned on μ'), and then reverse the isomorphism in (4.11). A more detailed description of the CG algorithm is as follows:

1. Start with input $|\lambda\rangle|q\rangle|i\rangle$ with $\lambda \in \mathcal{I}_{d,n}$, $|q\rangle \in \mathcal{Q}^d_\lambda$ and $i \in [d]$.
2. Unpack $|q\rangle$ into $|\mu\rangle|q_{d-1}\rangle$, with $\mu \precsim \lambda$ and $|q_{d-1}\rangle \in \mathcal{Q}^{d-1}_\mu$. This can be done efficiently because $|q\rangle$ is expressed in the GZ basis.
3. If $i \in \{1, \ldots, d-1\}$ then perform the $\mathcal{U}_{d-1}$ CG transform on $|\mu\rangle|q_{d-1}\rangle|i\rangle$ to obtain output $|\mu'\rangle|q'_{d-1}\rangle|j'\rangle$ with $|q'_{d-1}\rangle \in \mathcal{Q}^{d-1}_{\mu'}$ and $\mu' = \mu + e_{j'}$. Otherwise if $i = d$ then simply set $|\mu'\rangle = |\mu\rangle$, $|q'_{d-1}\rangle = |q_{d-1}\rangle$ and replace $|i\rangle = |0\rangle$ with $|j'\rangle = |0\rangle$. The "if/then/else" statement corresponds to (4.12a), while the conditional $\mathcal{U}_{d-1}$ CG transform is the isomorphism applied in (4.12c)
4. Perform the reduced Wigner transform $\hat{T}^\lambda$, by applying $\hat{T}^{\lambda,\mu'}$ conditional on μ' to map $|j'\rangle$ to $|j\rangle$, as per (4.13) and (4.14).
5. Map $|\lambda\rangle|j\rangle \mapsto |\lambda'\rangle|j\rangle$ with $\lambda' = \lambda + e_j$.
6. Pack $|\mu'\rangle$ and $|q'_{d-1}\rangle$ together into a GZ basis vector $|q'\rangle \in \mathcal{Q}^d_{\lambda'}$, as in (4.11).
7. Output $|\lambda'\rangle|q'\rangle|j\rangle$.

Finally we describe how to efficiently implement $\hat{T}^{\lambda,\mu'}$, starting with an efficiently-calculable formula for $\hat{T}^{\lambda,\mu'}_{j,j'}$ from Ref. [46]. First introduce the vectors $\tilde{\lambda} := \lambda + \sum_{i=1}^d (d-i) e_i$ and $\tilde{\mu} := \mu + \sum_{i'=1}^{d-1} (d-1-i') e_{i'}$ (where we recall that $\mu = \mu' - e_{j'}$). Also define $S_{j-j'}$ to be 1 if $j \geq j'$ and -1 if $j < j'$. Then according to Eq. (38) in Ref [46],

$$\hat{T}^{\lambda,\mu'}_{j,j'} = S_{j-j'} \left[\frac{\prod_{s \in [d-1] \setminus j} (\tilde{\lambda}_j - \tilde{\mu}_s) \prod_{t \in [d] \setminus j'} (\tilde{\mu}_{j'} - \tilde{\lambda}_t + 1)}{\prod_{s \in [d] \setminus j} (\tilde{\lambda}_j - \tilde{\lambda}_s) \prod_{t \in [d-1] \setminus j'} (\tilde{\mu}_{j'} - \tilde{\mu}_t + 1)} \right]^{\frac{1}{2}},$$

for $j' \in \{1, \ldots, d-1\}$, while for $j' = 0$ we have

$$\hat{T}^{\lambda,\mu'}_{j,0} = S_{j-d} \left[\frac{\prod_{s \in [d-1] \setminus j} (\tilde{\lambda}_j - \tilde{\mu}_s)}{\prod_{s \in [d] \setminus j} (\tilde{\lambda}_j - \tilde{\lambda}_s)} \right]^{\frac{1}{2}}.$$

The components of the partitions here are $O(n)$, so that each $\hat{T}^{\lambda,\mu'}_{j,j'}$ can be computed in time $\mathrm{poly}(d, \log n)$. We can also perform $\hat{T}^{\lambda,\mu'}$ (and thus $\hat{T}^{\lambda}$) to accuracy ϵ in time $\mathrm{poly}(d, \log n, \log 1/\epsilon)$ by (a) computing all d^2 matrix elements of $\hat{T}^{\lambda,\mu'}$ up to precision ϵ_1 , (b) decomposing this matrix into $d^2 \,\mathrm{poly}\log(d)$ elementary one and two-qubit operations[41], (c) approximating these operations to accuracy ϵ_2 with products of unitaries drawn from a fixed finite set (such as Clifford operators and a $\pi/8$ rotation) [44, 45], (d) applying these gates and (e) uncomputing all of the garbage bits produced by the classical computation along the way. By appropriate choice of ϵ_1 and ϵ_2 we achieve a total running time of $\mathrm{poly}(d, \log n, \log 1/\epsilon)$.

5 Conclusion

We have given an algorithm for the Schur transform with running time polynomial in the dimension d, the number of qudits, n, and the accuracy, $\log(1/\epsilon)$. This makes efficient a large set of quantum information protocols[13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23] whose computational efficiency has, prior to our work, been uncertain. Moreover, the existence of an efficient Schur transform raises the possibility of new quantum algorithms using it as a subroutine. Kuperberg's[39] subexponential algorithm for the dihedral hidden subgroup problem makes use of the effect of the CG transform for the dihedral group. Could similar techniques for $\mathcal{U}_d$ be useful? So far nonabelian quantum Fourier transforms [28, 29] have not had as many applications as their abelian counterparts, but perhaps the Schur transform will provide a fresh perspective.

Our techniques (exploiting $\mathcal{U}_d$-$\mathcal{S}_n$ duality and the structure of their subgroup-adapted bases) could also be applied to other pairs of groups, generally known as *dual reductive pairs*. Some candidates are discussed in [30, Sect 5.4], but no applications of these other transforms are yet known. In general, one can work with dual reductive pairs by studying either one of the component groups. This paper focussed on $\mathcal{U}_d$, but in a companion paper (to appear; see also [30, Chap 8]) we will explore connections between the $\mathcal{S}_n$ quantum Fourier transform and the Schur transform.

Acknowledgments: This work was partially funded by the NSF Institute for Quantum Information under grant number EIA-0086048 and the ARO under contracts DAAD19-01-1-06 and W911NF-04-R-0009. We thank Ashley Montanaro and Nolan Wallach for useful discussions.

References

[1] M. A. Nielsen and I. L. Chuang. *Quantum Computation and Quantum Information*. Cambridge University Press, New York, 2000.

[2] A. S. Holevo. The capacity of the quantum channel with general signal states. *IEEE Trans. Inf. Theory*, 44, 1998. quant-ph/9611023.

[3] B. Schumacher and M. D. Westmoreland. Sending classical information via noisy quantum channels. *Phys. Rev. A*, 56, 1997.

[4] C. H. Bennett, P. Hayden, D. W. Leung, P. W. Shor, and A. J. Winter. Remote preparation of quantum states. *IEEE Trans. Inf. Theory*, 51(1):56–74, 2005. quant-ph/0307100.

[5] I. Devetak and A.J. Winter. Relating quantum privacy and quantum coherence: an operational approach. *Phys. Rev. Lett.*, 93, 2004. quant-ph/0307053.

[6] B. Schumacher. Quantum coding. *Phys. Rev. A*, 51:2738–2747, 1995.

[7] R. Jozsa and B. Schumacher. A new proof of the quantum noiseless coding theorem. *J. Mod. Opt.*, 41:2343, 1994.

[8] R. Cleve and D.P. DiVincenzo. Schumacher's quantum data compression as a quantum computation. *Phys. Rev. A*, 54(4):2636–2650, 1996. quant-ph/9603009.

[9] P. Kaye and M. Mosca. Quantum networks for concentrating entanglement. *J. Phys. A*, 34:6939–6948, 2001. quant-ph/0101009.

[10] C. H. Bennett, H. J. Bernstein, S. Popescu, and B. Schumacher. Concentrating partial entanglement by local operations. *Phys. Rev. A*, 53:2046–2052, 1996.

[11] C. H. Bennett and G. Brassard. Quantum cryptography: Public key distribution and coin tossing. In *Proceedings of IEEE International Conference on Computers, Systems, and Signal Processing*, pages 175–179. IEEE, 1984.

[12] A. Ambainis and A. Smith. Small pseudo-random families of matrices: Derandomizing approximate quantum encryption. In K. Jansen, S. Khanna, J.D.P. Rolim, and D. Ron, editors, *APPROX-RANDOM*, volume 3122 of *Lecture Notes in Computer Science*, pages 249–260. Springer, 2004. quant-ph/0404075.

[13] M. Keyl and R. F. Werner. Estimating the spectrum of a density operator. *Phys. Rev. A*, 64:052311, 2001. quant-ph/0102027.

[14] M. Hayashi and K. Matsumoto. Universal distortion-free entanglement concentration, 2002. quant-ph/0209030.

[15] M. Hayashi and K. Matsumoto. Quantum universal variable-length source coding. *Phys. Rev. A*, 66(2):022311, 2002. quant-ph/0202001.

[16] M. Hayashi and K. Matsumoto. Simple construction of quantum universal variable-length source coding.

Quantum Inform. Compu., 2:519–529, 2002. quant-ph/0209124.

[17] M. Hayashi. Optimal sequence of quantum measurements in the sense of stein's lemma in quantum hypothesis testing. *J. Phys. A*, 35:10759–10773, 2002. quant-ph/0208020.

[18] P. Zanardi and M. Rasetti. Error avoiding quantum codes. *Mod. Phys. Lett. B*, 11(25):1085–1093, 1997.

[19] E. Knill, R. Laflamme, and L. Viola. Theory of quantum error correction for general noise. *Phys. Rev. Lett.*, 84:2525–2528, 2000.

[20] J. Kempe, D. Bacon, D. A. Lidar, and K. B. Whaley. Theory of decoherence-free fault-tolerant quantum computation. *Phys. Rev. A*, 63:042307–1–042307–29, 2001. quant-ph/0004064.

[21] D. Bacon. *Decoherence, Control, and Symmetry in Quantum Computers*. PhD thesis, University of Calfornia at Berkeley, Berkeley, CA, 2001. quant-ph/0305025.

[22] S.D. Bartlett, T. Rudolph, and R.W. Spekkens. Classical and quantum communication without a shared reference frame. *Phys. Rev. Lett.*, 91:027901, 2003.

[23] M. Hayashi and K. Matsumoto. Universal entanglement concentration, 2005. quant-ph/0509140.

[24] M. Hayashi. Two quantum analogues of fisher information from a large deviation viewpoint of quantum estimation. *J. Phys. A*, 35(36):7689–7727, 2002. quant-ph/0202003.

[25] C.H. Bennett, I. Devetak, A.W. Harrow, P.W. Shor, and A. Winter. The quantum reverse Shannon theorem, 2006. In preparation.

[26] I.M. Gelfand and M.L. Zetlin. Matrix elements for the unitary groups. *Dokl. Akad. Nauk.*, 71:825, 1950.

[27] G. D. James and A. Kerber. *The representation theory of the symmetric group*. Addison-Wesley, Reading, Mass., 1981.

[28] R. Beals. Quantum computation of fourier transforms over symmetric groups. In *Proceedings of the 29th Annual ACM Symposium on Theory of Computing*, pages 48–53, New York, NY, May 1997. ACM Press.

[29] Cristopher Moore, Daniel Rockmore, and Alexander Russell. Generic quantum Fourier transforms. In *SODA '04: Proceedings of the fifteenth annual ACM-SIAM symposium on Discrete algorithms*, pages 778–787, Philadelphia, PA, USA, 2004. Society for Industrial and Applied Mathematics. quant-ph/0304064.

[30] A. W. Harrow. *Applications of coherent classical communication and Schur duality to quantum information theory*. PhD thesis, M.I.T., Cambridge, MA, 2005.

[31] R. Goodman and N.R. Wallach. *Representations and Invariants of the Classical Groups*. Cambridge University Press, 1998.

[32] J. Chen, J. Ping, and F. Want. *Group Representation Theory for Physicists*. World Scientific, New Jersey, 2002.

[33] M. Hayashi and K. Matsumoto. Asymptotic performance of optimal state estimation in quantum two level system, 2004. quant-ph/0411073.

[34] M. Hayashi. Error exponents in quantum hypothesis testing, 2006. quant-ph/0607009.

[35] D. Bacon, I. Chuang, and A. Harrow. Efficient quantum circuits for quantum information theory. quant-ph/0407082, 2004.

[36] M. Hayashi and K. Matsumoto. Exponents of quantum fixed-length pure state source coding. *Phys. Rev. A*, 66:032321, 2002. quant-ph/0202002.

[37] R. Jozsa and S. Presnell. Universal quantum information compression and degrees of prior knowledge. *Proc. Roy. Soc. London Ser. A*, 459:3061–3077, October 2003. quant-ph/0210196.

[38] C.H. Bennett, A.W. Harrow, and S. Lloyd. Universal quantum data compression via gentle tomography. *Phys. Rev. A*, 73, 2006. quant-ph/0403078.

[39] G. Kuperberg. A subexponential-time quantum algorithm for the dihedral hidden subgroup. quant-ph/0302112, 2003.

[40] J. D. Louck. Recent progress toward a theory of tensor operators in unitary groups. *Am. J. Phys.*, 38(1):3, 1970.

[41] V.V. Shende, S.S. Bullock, and I.L. Markov. Synthesis of quantum logic circuits, 2004. quant-ph/0406176.

[42] M. Reck, A. Zeilinger, H.J. Bernstein, and P. Bertani. Experimental realization of any discrete unitary operator. *Phys. Rev. Lett.*, 73:58–61, 1994.

[43] A. Barenco. A universal two-bit gate for quantum computation. *Proc. Roy. Soc. London Ser. A*, 449:679–683, 1995.

[44] C.M. Dawson and M.A. Nielsen. The Solovay-Kitaev algorithm. quant-ph/0505030, 2005.

[45] A. Yu Kitaev, A. H. Shen, and M. N. Vyalyi. *Classical and Quantum Computation*, volume 47 of *Graduate Studies in Mathematics*. AMS, 2002.

[46] L. C. Biedenharn and J. D. Louck. A pattern calculus for tensor operators in the unitary groups. *Commun. Math. Phys.*, 8:89–131, 1968.

[47] A. Messiah. *Quantum Mechanics, Vol. 2*, chapter Representation of Irreducible Tensor Operators: Wigner-Eckart Theorem, pages 573–575. North-Holland, Amsterdam, Netherlands, 1962.

A.7 STACS Paper

Weak Fourier-Schur Sampling, the Hidden Subgroup Problem, and the Quantum Collision Problem

Andrew M. Childs[1], Aram W. Harrow[2], and Paweł Wocjan[3]

[1] Institute for Quantum Information, California Institute of Technology, Pasadena, CA 91125, USA
amchilds@caltech.edu
[2] Department of Computer Science, University of Bristol, Bristol, BS8 1UB, UK
a.harrow@bris.ac.uk
[3] School of Electrical Engineering and Computer Science, University of Central Florida, Orlando, FL 32816, USA
wocjan@cs.ucf.edu

Abstract. Schur duality decomposes many copies of a quantum state into subspaces labeled by partitions, a decomposition with applications throughout quantum information theory. Here we consider applying Schur duality to the problem of distinguishing coset states in the standard approach to the hidden subgroup problem. We observe that simply measuring the partition (a procedure we call *weak Schur sampling*) provides very little information about the hidden subgroup. Furthermore, we show that under quite general assumptions, even a combination of weak Fourier sampling and weak Schur sampling fails to identify the hidden subgroup. We also prove tight bounds on how many coset states are required to solve the hidden subgroup problem by weak Schur sampling, and we relate this question to a quantum version of the collision problem.

1 Introduction

The hidden subgroup problem (HSP) is a central challenge for quantum computation. On the one hand, many of the known fast quantum algorithms are based on the efficient solution of the abelian HSP [21, 22, 38, 41]. On the other hand, the *nonabelian* HSP has potential applications: in particular, the graph isomorphism problem can be reduced to the HSP in the symmetric group [8, 14], and the shortest lattice vector problem can be reduced to a variant of the HSP in the dihedral group [36]. Unfortunately, no efficient algorithms are known for these two instances of the nonabelian HSP. However, some partial progress has been made: there is a subexponential time algorithm for the dihedral HSP [31, 37], and it is known how to solve the HSP efficiently for a variety of other nonabelian groups [2, 16, 17, 19, 25, 28, 33].

In the HSP for a group G, we have black-box access to a function $f : G \to S$, where S is some finite set. We say that f hides a subgroup $H \leq G$ provided

W. Thomas and P. Weil (Eds.): STACS 2007, LNCS 4393, pp. 598–609, 2007.

$f(g) = f(g')$ iff $g^{-1}g' \in H$. The goal is to determine H (say, in terms of a generating set) as quickly as possible. In particular, we say that an algorithm for the HSP in G is efficient if it runs in time $\mathrm{poly}(\log |G|)$.

Nearly all quantum algorithms for the HSP use the so-called *standard method*, in which we query f on a uniform superposition of group elements and then discard the function value, giving a *coset state* $|gH\rangle := |H|^{-1/2} \sum_{h \in H} |gh\rangle$ for some unknown, uniformly random $g \in G$. This state is described by the density matrix

$$\rho_H := \frac{1}{|G|} \sum_{g \in G} |gH\rangle\langle gH| = \frac{1}{|G|} \sum_{h \in H} R(h) \tag{1}$$

(called a *hidden subgroup state*), where R is the *right regular representation* of G, satisfying $R(g)|g'\rangle = |g'g^{-1}\rangle$ for all $g, g' \in G$. Now the HSP is reduced to the problem of distinguishing the states ρ_H for the possible $H \leq G$.

The symmetry of ρ_H can be exploited using Fourier analysis. In particular, the group algebra $\mathbb{C}G$ decomposes under the commuting left and right multiplication actions of G as

$$\mathbb{C}G \overset{G \times G}{\cong} \bigoplus_{\sigma \in \hat{G}} \mathcal{V}_\sigma \otimes \mathcal{V}_\sigma^* \tag{2}$$

where $\hat{G}$ denotes a complete set of irreducible representations (or *irreps*) of G, and $\mathcal{V}_\sigma$ and $\mathcal{V}_\sigma^*$ are the (row and column, respectively) subspaces acted on by $\sigma \in \hat{G}$. The unitary transformation that relates the standard basis for $\mathbb{C}G$ and the basis for the spaces $\mathcal{V}_\sigma \otimes \mathcal{V}_\sigma^*$ is the Fourier transform, which can be carried out efficiently for most groups of interest [7, 12, 23, 32].

Since ρ_H is invariant under the left multiplication action of G, the decomposition (2) shows that it is block diagonal in the Fourier basis, with blocks labeled by the irreps $\sigma \in \hat{G}$. For each σ, there is a $\dim \mathcal{V}_\sigma \times \dim \mathcal{V}_\sigma$ block that appears $\dim \mathcal{V}_\sigma$ times (or in other words, the state is maximally mixed in the row space). Thus, without loss of information, we can measure the irrep name σ and discard the information about which σ-isotypic block occurred.

The process of measuring the irrep name σ is referred to as *weak Fourier sampling*. For most nonabelian groups (including the symmetric group [19, 25] and the dihedral group), weak Fourier sampling alone produces insufficient information to identify the hidden subgroup H. To obtain further information about H, we must perform a refined measurement inside the resulting subspace. This is referred to as *strong Fourier sampling*, and there are many possible ways to do it, especially if G has large irreps.

Of course, with either weak or strong Fourier sampling, a single hidden subgroup state is not sufficient to determine H: we must repeat the sampling procedure to obtain statistics. However, repeating strong Fourier sampling a polynomial number of times is not sufficient for some groups (such as the symmetric group), even if measurements can be chosen adaptively and unlimited classical processing is allowed [34]. To solve the HSP in general, we must perform a joint measurement on $k = \mathrm{poly}(\log |G|)$ copies of $\rho_H^{\otimes k}$. In fact, there are groups (again including the symmetric group) for which the measurement must be entangled

across $\Omega(\log|G|)$ copies [24]. Thus the difficulty of the general HSP may be attributed at least in part to that fact that highly entangled measurements are required. While $O(\log|G|)$ copies are always information-theoretically sufficient [15] (so that, in particular, the query complexity of the HSP is polynomial), there are many groups for which it is not known how to efficiently extract the identity of the hidden subgroup.

Although previous work on the HSP has focused almost exclusively on Fourier sampling, there is another measurement that can also be performed without loss of information. The idea is to exploit the symmetry of $\rho_H^{\otimes k}$ under permutations of the k registers. Thus, we should consider the decomposition of $(\mathbb{C}G)^{\otimes k}$ afforded by *Schur duality* [18], which decomposes k copies of a d-dimensional space as

$$(\mathbb{C}^d)^{\otimes k} \overset{\mathcal{S}_k \times \mathcal{U}_d}{\cong} \bigoplus_{\lambda \vdash k} \mathcal{P}_\lambda \otimes \mathcal{Q}_\lambda^d \tag{3}$$

where the symmetric group $\mathcal{S}_k$ acts to permute the k registers and the unitary group $\mathcal{U}_d$ acts identically on each register. The subspaces $\mathcal{P}_\lambda$ and $\mathcal{Q}_\lambda^d$ correspond to irreps of $\mathcal{S}_k$ and $\mathcal{U}_d$, respectively. They are labeled by partitions λ of k (denoted $\lambda \vdash k$), i.e., $\lambda = (\lambda_1, \lambda_2, \ldots)$ where $\lambda_1 \geq \lambda_2 \geq \ldots$ and $\sum_j \lambda_j = k$. (We can restrict our attention to partitions with at most d parts, since $\dim \mathcal{Q}_\lambda^d = 0$ if $\lambda_{d+1} > 0$.)

Since $\rho_H^{\otimes k}$ is invariant under the action of $\mathcal{S}_k$, the decomposition (3) shows that it is block diagonal in the Schur basis with blocks labeled by $\lambda \vdash k$. For each λ, there is a $\dim \mathcal{Q}_\lambda^{|G|} \times \dim \mathcal{Q}_\lambda^{|G|}$ block that appears $\dim \mathcal{P}_\lambda$ times (or in other words, the state is maximally mixed in the permutation space). Thus, no information is lost if we measure the partition λ and discard the permutation register. By analogy to weak Fourier sampling, we refer to the process of measuring λ as *weak Schur sampling*. This is a natural measurement to consider not only because it can be performed without loss of information, but also because it is a joint measurement of all k registers, and we know that some measurement of this kind is required to solve the general HSP. Unfortunately, we will see in Section 2 (and see also Corollary 4 below) that weak Schur sampling with $k = \mathrm{poly}(\log|G|)$ provides insufficient information to solve the HSP unless the hidden subgroup is very large (in which case the problem is easy, even for a classical computer).

In fact, since both weak Fourier sampling and weak Schur sampling can be performed without loss of information, it is possible to perform both measurements simultaneously (with the caveat that we must discard the irrelevant information about the order in which the irreps of G appear). Even though the statistics of the irrep name σ and the partition λ do not provide enough information to identify the hidden subgroup, this does not preclude the possibility that their joint distribution is more informative. However, we will see in Section 3 that unless we are likely to see the same representation more than once under weak Fourier sampling (which is typically not the case), the Fourier and Schur distributions are nearly uncorrelated. Formally, we have

Theorem 1 (Failure of weak Fourier-Schur sampling). *The probability that weak Fourier-Schur sampling (defined in Section 3) applied to $\rho_H^{\otimes k}$ (defined*

in (1)) provides a result that depends on $|H|$ is at most $k^2 d_{\max}^2 |H|/|G|$, where $d_{\max}$ is the largest dimension of an irrep of G.

This implies that k needs to be large for most cases of interest, including the dihedral and symmetric groups.

Corollary 2 (Weak Fourier-Schur sampling on $\mathcal{D}_N$ and $\mathcal{S}_n$). *(a) Weak Fourier-Schur sampling on the dihedral group $\mathcal{D}_N$ cannot distinguish the trivial subgroup from a hidden reflection with constant advantage (i.e., success probability $\frac{1}{2} + \Omega(1)$) unless $k = \Omega(\sqrt{N})$. (b) Weak Fourier-Schur sampling on the symmetric group $\mathcal{S}_n$ or on the wreath product $\mathcal{S}_n \wr \mathbb{Z}_2$ cannot distinguish the trivial subgroup from an order 2 subgroup with constant advantage unless $k = \exp(\Omega(\sqrt{n}))$.*

The proof that weak Schur sampling fails is based on the simple observation that distinguishing the trivial subgroup from a subgroup of order $|H|$ in this way requires us to distinguish 1-to-1 from $|H|$-to-1 functions on G, i.e., to solve the $|H|$-collision problem for a list of size $|G|$. Since there is an $\Omega(\sqrt[3]{|G|/|H|})$ quantum lower bound for this problem [1], $\mathrm{poly}(\log |G|)$ registers are insufficient. In fact, the problem resulting from the HSP is potentially harder, since the basis in which the collisions occur is inaccessible to the Schur measurement. This naturally leads to the notion of a *quantum* collision problem, and raises the question of how quickly it can be solved on a quantum computer, which we discuss in Section 4.

We first consider a sampling version of the quantum r-collision problem. Using results on the asymptotics of the Plancherel measure on the symmetric group, we prove that $k = \Theta(d/r)$ registers are necessary and sufficient to solve this problem. In particular, we have

Theorem 3 (Quantum collision sampling problem). *Given $\rho^{\otimes k}$, distinguishing between [case A] $\rho = I/d$ and [case B] $\rho^2 = \rho/\frac{d}{r}$ (i.e., ρ is proportional to a projector of rank d/r) is possible with success probability $1 - \exp(-\Theta(kr/d))/2$. In particular, constant advantage is possible iff $k = \Omega(d/r)$.*

In addition to providing the first results on estimation of the spectrum of a quantum state in the regime where $k \ll d^2$, this gives tight estimates of the effectiveness of weak Schur sampling, which we see requires an exponentially large (in $\log |G|$) number of copies to be successful.

Corollary 4 (Failure of weak Schur sampling). *Applying weak Schur sampling to $\rho_H^{\otimes k}$ (where ρ_H is defined in (1)), one can distinguish the case $|H| \geq r$ from the case $H = \{1\}$ with constant advantage iff $k = \Omega(|G|/r)$.*

The connection between Theorem 3 and Corollary 4 is explained in Section 2.

In Section 4 we also introduce a black box version of the quantum collision problem. We show that it can be solved using $O(\sqrt[3]{d/r} \log d/r)$ queries, nearly matching the query lower bound from the classical problem.

2 Weak Schur Sampling

We begin by considering only the permutation symmetry of $\rho_H^{\otimes k}$, without taking into account symmetry resulting from the group G. In other words, we consider only the Schur decomposition (3), and we perform *weak Schur sampling*, i.e., a measurement of the partition λ.

The projector onto the subspace labeled by a particular $\lambda \vdash k$ is

$$\Pi_\lambda := \frac{\dim \mathcal{P}_\lambda}{k!} \sum_{\pi \in \mathcal{S}_k} \chi_\lambda(\pi)\, P(\pi) \tag{4}$$

(see e.g. [40, Theorem 8]), where χ_λ is the character of the irrep of $\mathcal{S}_k$ labeled by λ and P is the (reducible) representation of $\mathcal{S}_k$ that acts to permute the k registers, i.e., $P(\pi)|i_1\rangle \dots |i_k\rangle = |i_{\pi^{-1}(1)}\rangle \dots |i_{\pi^{-1}(k)}\rangle$ for all $i_1, \dots, i_k \in \{1, \dots, d\}$. For any d^k-dimensional density matrix γ, the distribution under weak Schur sampling is

$$\Pr(\lambda|\gamma) = \operatorname{tr}(\Pi_\lambda \gamma)\,. \tag{5}$$

To use weak Schur sampling in a quantum algorithm, it is important that the measurement of λ can be done efficiently. The simplest implementation of the complete Schur transform [5], which fully resolves the subspaces $\mathcal{P}_\lambda$ and Q_λ^d, runs in time $\operatorname{poly}(k, d)$, and thus is inefficient when d is exponentially large, as in the HSP. It can be modified to run in time $\operatorname{poly}(k, \log d)$ either by a relabeling trick [26, footnote in Section 8.1.2] or by *generalized phase estimation* [4, 26] (which may be viewed as a generalization of the well-known swap test [6, 10]). Generalized phase estimation only allows us to measure λ, but for weak Schur sampling this is all we need. In this procedure, we prepare an ancilla register in the state $\frac{1}{\sqrt{k!}} \sum_{\pi \in \mathcal{S}_k} |\pi\rangle$, use it to perform a conditional permutation $P(\pi)$ on the input state γ, and then perform an inverse Fourier transform over $\mathcal{S}_k$ [7] on the ancilla register. Measurement of the ancilla register will then yield $\lambda \in \hat{\mathcal{S}}_k$, interpreted as a partition of k, distributed according to (5).

The distribution of λ according to weak Schur sampling is invariant under the actions of the permutation and unitary groups, since these groups act only within the subspaces $\mathcal{P}_\lambda$ and $\mathcal{Q}_\lambda^d$, respectively. In other words, for any $U \in \mathcal{U}_d$, any $\pi \in \mathcal{S}_k$, and any d^k-dimensional density matrix γ, we have $\Pr(\lambda|\gamma) = \Pr(\lambda|P(\pi)U^{\otimes k}\,\gamma\, U^{\dagger\otimes k} P(\pi)^\dagger)$. In particular, the invariance under $U^{\otimes k}$ implies that for $\gamma = \rho^{\otimes k}$, the distribution according to weak Schur sampling depends only on the spectrum of ρ.

Now it is easy to see that weak Schur sampling on $k = \operatorname{poly}(\log |G|)$ copies of ρ_H provides insufficient information to solve the HSP. The state ρ_H is proportional to a projector of rank $|G|/|H|$, since

$$\rho_H^2 = \frac{1}{|G|^2} \sum_{h,h' \in H} R(hh') = \frac{|H|}{|G|}\rho_H\,. \tag{6}$$

Because the distribution of measurement outcomes $\Pr(\lambda|\rho_H^{\otimes k})$ depends only on the spectrum of ρ_H, and this spectrum depends only on $|H|$, different subgroups

of the same order cannot be distinguished by weak Schur sampling. In fact, even distinguishing the trivial hidden subgroup from a hidden subgroup of order $|H| \geq 2$ (which would suffice for, e.g., graph isomorphism) requires an exponential number of hidden subgroup states.

Suppose that weak Schur sampling could distinguish between hidden subgroup states corresponding to $H = \{1\}$ and some particular H of order $|H| \geq 2$. Since the distribution of λ depends only on the spectrum, this would mean that we could distinguish k copies of the maximally mixed state $I_{|G|}/|G|$, where I_d is the $d \times d$ identity matrix, from k copies of the state $J_{|G|/|H|}/(|G|/|H|)$, where $J_{d'}$ is a projector onto an arbitrary subspace of dimension d'. This in turn would imply that we could distinguish 1-to-1 functions from $|H|$-to-1 functions using k queries of the function. Then the quantum lower bound for the $|H|$-collision problem [1] shows that $k = \Omega(\sqrt[3]{|G|/|H|})$ copies are required.

Of course, this does not mean that $O(\sqrt[3]{|G|/|H|})$ copies are sufficient. In fact, it turns out that a linear number of copies is both necessary and sufficient, as we will show by a more careful analysis in Section 4. There we will sketch the proof of Theorem 3, which by the arguments of this section implies Corollary 4.

3 Weak Fourier-Schur Sampling

In the previous section, we showed that weak Schur sampling provides insufficient information to efficiently solve the HSP. However, even though weak Fourier sampling typically also does not provide enough information, it is conceivable that the joint distribution of the two measurements could be substantially more informative. In this section, we will see that this is not the case: provided weak Fourier sampling fails, so does weak Fourier-Schur sampling.

Since neither measurement constitutes a loss of information, it is in principle possible to perform both weak Fourier sampling and weak Schur sampling simultaneously. If we perform weak Fourier sampling in the usual way, measuring the irrep label for each register, then we will typically obtain a state that is no longer permutation invariant. However, since the irrep labels are identically distributed for each register, the order in which the irreps appear carries no information. Only the *type* of the irreps, i.e., the number of times each irrep appears, is relevant. Thus, it suffices to perform what we might call *weak Fourier type sampling*, in which we only measure the irrep type. Equivalently, we could perform complete weak Fourier sampling and then either randomly permute the k registers, or perform weak Schur sampling and discard the $\mathcal{P}_\lambda$ register.

We begin by performing weak Fourier sampling. The hidden subgroup state ρ_H defined in (1) has the following block structure in the Fourier basis:

$$\rho_H \cong \frac{1}{|G|} \bigoplus_{\sigma \in \hat{G}} I_{\dim \mathcal{V}_\sigma} \otimes \sum_{h \in H} \sigma(h)^* =: \sum_{\sigma \in \hat{G}} \Pr(\sigma) \frac{I_{\dim \mathcal{V}_\sigma}}{\dim \mathcal{V}_\sigma} \otimes \rho_{H,\sigma} \,. \tag{7}$$

Here the probability of observing the irrep σ under weak Fourier sampling is $\Pr(\sigma) = (\dim \mathcal{V}_\sigma/|G|) \sum_{h \in H} \chi_\sigma(h)^*$ and the state conditioned on this observation is $\rho_{H,\sigma} = \left(\sum_{h \in H} \chi_\sigma(h)\right)^{-1} \sum_{h \in H} |\sigma\rangle\langle\sigma| \otimes \sigma(h)^*$

Repeating weak Fourier sampling k times, we get $\rho_{H,\underline{\sigma}} = \rho_{H,\sigma_1} \otimes \cdots \otimes \rho_{H,\sigma_k}$, where $\underline{\sigma} := (\sigma_1, \sigma_2, \ldots, \sigma_k) \in \hat{G}^k$ may be viewed either as the actual outcome of the k instances of weak Fourier sampling, or merely as a representative of the irrep type, as discussed above. Given this state, the conditional probability of observing the partition λ is

$$\Pr(\lambda|\underline{\sigma}) = \operatorname{tr}(\Pi_\lambda \, \rho_{H,\underline{\sigma}}) = \frac{\dim \mathcal{P}_\lambda}{k!} \sum_{\pi \in \mathcal{S}_k} \chi_\lambda(\pi) \operatorname{tr}[P(\pi) \, \rho_{H,\underline{\sigma}}] \,. \tag{8}$$

Note that $\operatorname{tr}[P(\pi) \, \rho_{H,\underline{\sigma}}] = 0$ if $\pi(\underline{\sigma}) \neq \underline{\sigma}$, where $\pi(\underline{\sigma}) = (\sigma_{\pi^{-1}(1)}, \ldots, \sigma_{\pi^{-1}(k)})$.

Proof (Theorem 1). Assume that $\underline{\sigma}$ is multiplicity-free, i.e., that all the σ_i's are different. In this case the traces are zero for all $\pi \neq 1$ (the identity of $\mathcal{S}_k$). Then $\Pr(\lambda|\underline{\sigma}) = \frac{\dim \mathcal{P}_\lambda}{k!} \chi_\lambda(1) \operatorname{tr} \rho_{H,\underline{\sigma}} = \frac{(\dim \mathcal{P}_\lambda)^2}{k!}$, which is nothing but the Plancherel distribution over $\hat{\mathcal{S}}_k$, and which in particular is independent of the hidden subgroup H. This shows that we cannot extract any information about H provided that we have obtained a multiplicity-free $\underline{\sigma}$.

Finally, we can use $|\chi_\sigma(h)| \leq \dim \mathcal{V}_\sigma$ to show that the probability of any σ is $\leq d_{\max}^2 |H|/|G|$, and then use a union bound to prove that $\underline{\sigma}$ is multiplicity-free with probability $\geq 1 - \binom{k}{2} d_{\max}^2 |H|/|G|$.

In [11] two of us considered an alternative approach to graph isomorphism based on the nonabelian hidden shift problem. It can be shown that weak Fourier-Schur sampling fails for similar reasons when applied to hidden shift states instead of hidden subgroup states.

4 The Quantum Collision Problem

In Section 2, we saw that weak Schur sampling cannot efficiently solve the HSP since this would require solving the collision problem. In fact, the problem faced by weak Schur sampling is considerably harder, since no information is available about the basis in which collisions occur. This motivates quantum generalizations of the usual (i.e., classical) collision problem, which we study in this section.

Let us briefly review the classical problem. The classical r-collision problem is the problem of determining whether a black box function with d inputs (where r divides d) is 1-to-1 or r-to-1. This problem has classical (randomized) query complexity $\Theta(\sqrt{d/r})$—as evidenced by the well-known birthday problem—and quantum query complexity $\Theta(\sqrt[3]{d/r})$ [1,9]. The classical algorithm is quite simple: after querying the function on $O(\sqrt{d/r})$ random inputs, there is a reasonable probability of seeing a collision, provided one exists. The quantum algorithm is slightly more subtle, making use of Grover's algorithm for unstructured search [20]. In particular, while the classical algorithm queries the black box nonadaptively, it is essential for the quantum algorithm to make adaptive queries.

Here we first consider a sampling version of the quantum collision problem, which is closely connected to the weak Schur sampling approach to the HSP, and then study a full-fledged black box version of the problem.

The quantum collision sampling problem. The *quantum r-collision sampling problem* is the problem of deciding whether one has k copies of the d-dimensional maximally mixed state or of a state that is maximally mixed on an unknown subspace of dimension d/r. This is exactly the problem faced by the weak Schur sampling approach to the HSP, so our results on the quantum collision sampling problem give tight bounds on the effectiveness of weak Schur sampling. It turns out that $k = \Theta(d/r)$ copies are necessary and sufficient to distinguish these two cases with constant advantage, as stated by Theorem 3.

Proof sketch (Theorem 3). Weak Schur sampling is the optimal strategy to distinguish states ρ with [case A] $\rho = I/d$ or [case B] $\rho^2 = \rho/\frac{d}{r}$. We call the resulting distribution of $\lambda \vdash k$ arising in case A the *Schur distribution*, $\mathrm{Schur}(k, d)$, with

$$\Pr(\lambda) = \frac{\dim \mathcal{P}_\lambda \dim \mathcal{Q}_\lambda^d}{d^k} = \frac{(\dim \mathcal{P}_\lambda)^2}{k!} \prod_{(i,j)\in\lambda} \left(1 + \frac{j-i}{d}\right). \tag{9}$$

The second equality follows from Stanley's formula for $\dim \mathcal{Q}_\lambda^d$ [42], interpreting λ as a Young diagram, where $(i, j) \in \lambda$ iff $1 \leq j \leq \lambda_i$. The outcomes in case B are also Schur-distributed (by a simple representation-theoretic argument), but here the distribution is $\mathrm{Schur}(k, d/r)$.

Our first goal is to show that the distributions $\mathrm{Schur}(k, d)$ and $\mathrm{Schur}(k, d/r)$ are close when $k \ll d/r$. We do this by showing that when $k \ll d$, $\mathrm{Schur}(k, d)$ is close to the *Plancherel distribution* of $\lambda \vdash k$, $\mathrm{Planch}(k)$, for which

$$\Pr(\lambda) = \frac{(\dim \mathcal{P}_\lambda)^2}{k!}. \tag{10}$$

Using (9) and (10), the ℓ_1 distance $\Delta_{k,d} := \| \mathrm{Schur}(k, d) - \mathrm{Planch}(k)\|_1$ is

$$\Delta_{k,d} = \mathop{\mathbf{E}}_{\lambda\vdash k} \left| \prod_{(i,j)\in\lambda} \left(1 + \frac{j-i}{d}\right) - 1 \right| \tag{11}$$

where the expectation is over $\mathrm{Planch}(k)$. Using Cauchy-Schwartz and the inequality $1 + x \leq e^x$, we can upper bound (11) by

$$\Delta_{k,d}^2 \leq \mathop{\mathbf{E}}_{\lambda\vdash k} \exp\left(2 \sum_{(i,j)\in\lambda} \frac{j-i}{d}\right) = \sum_{m=1}^{\infty} \frac{2^m}{m!\, d^m} \mathop{\mathbf{E}}_{\lambda\vdash k} v_1(\lambda)^m, \tag{12}$$

where $v_1(\lambda) := \sum_{(i,j)\in\lambda}(j-i)$. Finally, we use calculations of the moments of v_1 obtained by Kerov in the course of describing the asymptotically Gaussian fluctuations about the limiting shape of the typical diagram under the Plancherel distribution [29]. This establishes $\Delta_{k,d} \leq \sqrt{2}(k/d)$, and it follows from the triangle inequality that $\mathrm{Schur}(k, d)$ and $\mathrm{Schur}(k, d/r)$ are close when $k \ll d/r$.

Conversely, we would like to show that if $k \gg d/r$, then $\mathrm{Schur}(k, d)$ is far from $\mathrm{Schur}(k, d/r)$. We do this by first proving a lower bound on $\Delta_{k,d}$ (using similar techniques as in the upper bound on $\Delta_{k,d}$, as well as a one-sided Chebyshev inequality showing $v_1(\lambda)^2 \geq \Omega(k^2)$ with constant probability). Then we

combine this with the upper bound on $\Delta_{k,d}$ and use a monotonicity argument ($\|\mathrm{Schur}(k,d_1) - \mathrm{Schur}(k,d_2)\|_1 \geq \|\mathrm{Schur}(k,rd_1) - \mathrm{Schur}(k,rd_2)\|_1$) to separate the Schur distributions. This completes the proof sketch.

To put Theorem 3 in context, we can compare it to results on spectrum estimation. When $k \to \infty$ with d fixed, applying the measurement $\{\Pi_\lambda\}_{\lambda \vdash k}$ to $\rho^{\otimes k}$ and outputting $\bar{\lambda} := \lambda/k$ has long been known to be a valid estimator of the spectrum of ρ [30]. Indeed, if $r_1 \geq \ldots \geq r_d$ are the eigenvalues of ρ, then $\operatorname{tr} \Pi_\lambda \rho^{\otimes k} \leq (k+1)^{d(d-1)/2} \exp\left(-kD(\bar{\lambda}\|r)\right)$, where $D(p\|q) := \sum_i p_i \log(p_i/q_i)$ is the (classical) relative entropy [13,27]. This inequality is usually only interesting when $k = \Omega(d^2)$, so our Theorem 3 can be viewed as the first positive result for spectrum estimation in the regime where $k = o(d^2)$.

A black box for the quantum collision problem. A complete definition of the quantum collision problem requires us to specify a unitary black box that hides the function, and that allows us to make adaptive queries. We now propose one such definition, and show that the resulting quantum r-collision problem can be solved in $O(\sqrt[3]{d/r}\log d/r)$ queries, nearly matching the $\Omega(\sqrt[3]{d/r})$ lower bound from the classical collision problem.

Consider a quantum oracle that implements the isometry $|i\rangle \mapsto |i\rangle|\psi_{f(i)}\rangle$, where $\mathcal{B} := \{|\psi_1\rangle, \ldots, |\psi_d\rangle\}$ is an arbitrary (unknown) orthonormal basis of $\mathbb{C}^d$ and f is either a 1-to-1 function or an r-to-1 function. The goal is to determine which is the case using as few queries as possible. We assume that the isometry is extended to a unitary operator R acting on $\mathbb{C}^d \otimes \mathbb{C}^d$ by $|i\rangle|y\rangle \mapsto |i\rangle\, U|y \oplus f(i)\rangle$, where $U := \sum_i |\psi_i\rangle\langle i|$ is the unitary matrix effecting a transformation from the standard basis to $\mathcal{B}$. We also assume we can perform its inverse $R^\dagger$.

By considering the case where the basis $\mathcal{B}$ (or equivalently U) is known, it is clear that the quantum lower bound for the usual collision problem implies an $\Omega(\sqrt[3]{d/r})$ lower bound for the quantum collision problem as well. We present an algorithm for this problem that uses only $O(\sqrt[3]{d/r}\log d/r)$ queries. The basic idea is to adapt the quantum algorithm for the classical collision problem [9]. That algorithm is not directly applicable to the quantum problem since we cannot check equality of quantum states. However, the swap test can determine whether two states are identical or orthogonal with one-sided error of 1/2. With $O(\log d)$ copies of each state, this error (and the resulting state disturbance) can be reduced to $1/\operatorname{poly}(d)$. We use this *amplified swap test* to prove

Theorem 5. *The query complexity of the quantum r-collision problem for a list of size d is $O(\sqrt[3]{d/r}\log d/r)$.*

Proof. We first outline the quantum algorithm of [9] for the classical collision problem. The algorithm builds a table of a random set of $\sqrt[3]{d/r}$ items and uses Grover's algorithm to search the remaining items for a collision with an entry of the table. The entries of the table are distinct with high probability. If f is r-to-1, there are $(r-1)(d/r)^{1/3}$ solutions among $< d$ items, for a total query complexity of $O(\sqrt{d/[r(d/r)^{1/3}]}) = O((d/r)^{1/3})$.

Now we adapt this algorithm to the quantum problem. Using the amplified swap test, we can effectively test equality using $m := 2 + 2\log d/r$ copies of the

quantum states, increasing the query complexity only by a factor of $O(\log d/r)$. For this to work, it is important that we can reuse the states corresponding to the entries in the table, so we will need m copies of each state in the table as well. Iterating this swap test, we find that the error after ℓ Grover iterations is at most $\ell \cdot 2^{1-m/2} \leq \ell r/d$. Since the number of Grover iterations is $\ell = O((d/r)^{1/3})$, the total error is asymptotically negligible, and we obtain nearly the same performance as in the classical collision problem.

5 Discussion

We have shown that weak Fourier-Schur sampling typically provides insufficient information to solve the hidden subgroup problem. Nevertheless, it remains possible that Schur duality could be a useful tool for the HSP. Just as weak Fourier sampling refines the space into smaller subspaces in which we can perform strong Fourier sampling, even when it alone fails to solve the HSP, so we can use weak Fourier-Schur sampling to decompose the space even further. The Schur decomposition has the additional complication that the refined subspaces are no longer simply tensor products of single-copy subspaces, but this may actually be an advantage since entangled measurements are known to be necessary for some groups. Also, Schur sampling may be useful for implementing optimal measurements, which are typically entangled [2, 3].

In principle, strong Fourier-Schur sampling is guaranteed to provide enough information to solve the HSP, simply because the hidden subgroup states are always distinguishable with $k = \mathrm{poly}(\log |G|)$ copies. However, it would be interesting to find a new efficient quantum algorithm for some HSP based on strong Fourier-Schur sampling. Perhaps a first step in this direction would be to analyze the performance of measurement in a random basis, as has been studied extensively in the case of weak Fourier sampling [19, 33, 35, 39].

Moving away from our original motivation of the HSP, the quantum collision problem may be of independent interest. As discussed in Section 4, our results on the quantum collision sampling problem can be viewed as an exploration of spectrum estimation with $k = o(d^2)$ copies, but much remains unknown about that regime. Many open problems also remain regarding variants of the black box version of the quantum collision problem.

Acknowledgments. We thank Scott Aaronson, Andris Ambainis, Masahito Hayashi, Keiji Matsumoto, Pranab Sen, and Umesh Vazirani for helpful discussions. We also thank Patrick Hayden for organizing a Bellairs Research Institute workshop on representation theory in quantum information, at which the seeds for this work were planted. This work was supported in part by the National Science Foundation under grant PHY-456720, by the Army Research Office under grant W9111NF-05-1-0294, by the European Commission under Marie Curie grants ASTQIT (FP6-022194) and QAP (IST-2005-15848), and by the U.K. Engineering and Physical Science Research Council through "QIP IRC."

References

[1] S. Aaronson and Y. Shi, *Quantum lower bounds for the collision and the element distinctness problems*, J. ACM **51** (2004), no. 4, 595–605.

[2] D. Bacon, A. M. Childs, and W. van Dam, *From optimal measurement to efficient quantum algorithms for the hidden subgroup problem over semidirect product groups*, Proc. 46th FOCS, 2005, pp. 469–478.

[3] ———, *Optimal measurements for the dihedral hidden subgroup problem*, Chicago J. Th. Comp. Sci. (2006), no. 2.

[4] D. Bacon, I. L. Chuang, and A. W. Harrow, *Efficient quantum circuits for Schur and Clebsch-Gordan transforms*, quant-ph/0407082.

[5] ———, *The quantum Schur transform: I. Efficient qudit circuits*, to appear in Proc. 18th SODA, 2007, available at quant-ph/0601001.

[6] A. Barenco, A. Berthiaume, D. Deutsch, A. Ekert, R. Jozsa, and C. Macchiavello, *Stabilisation of quantum computations by symmetrisation*, SIAM J. Comput. (1997), 1541–1557.

[7] R. Beals, *Quantum computation of Fourier transforms over symmetric groups*, Proc. 29th STOC, 1997, pp. 48–53.

[8] R. Boneh and R. Lipton, *Quantum cryptanalysis of hidden linear functions*, Proc. Advances in Cryptology, LNCS **963**, 1995, pp. 424–437.

[9] G. Brassard, P. Høyer, and A. Tapp, *Quantum cryptanalysis of hash and claw-free functions*, Proc. 3rd Latin American Symposium on Theoretical Informatics, LNCS **1380**, 1998, pp. 163–169.

[10] H. Buhrman, R. Cleve, J. Watrous, and R. de Wolf, *Quantum fingerprinting*, Phys. Rev. Lett. **87** (2001), 167902.

[11] A. M. Childs and P. Wocjan, *On the quantum hardness of solving isomorphism problems as nonabelian hidden shift problems*, quant-ph/0510185.

[12] D. Coppersmith, *An approximate Fourier transform useful in quantum factoring*, Technical Report RC 19642, IBM Research Division, Yorktown Heights, NY, 1994, quant-ph/0201067.

[13] M. Christandl and G. Mitchison, *The spectra of density operators and the Kronecker coefficients of the symmetric group*, Commun. Math. Phys. **261** (2006), no. 3, 789–797.

[14] M. Ettinger and P. Høyer, *A quantum observable for the graph isomorphism problem*, quant-ph/9901029.

[15] M. Ettinger, P. Høyer, and E. Knill, *Hidden subgroup states are almost orthogonal*, quant-ph/9901034.

[16] K. Friedl, G. Ivanyos, F. Magniez, M. Santha, and P. Sen, *Hidden translation and orbit coset in quantum computing*, Proc. 35th STOC, 2003, pp. 1–9.

[17] D. Gavinsky, *Quantum solution to the hidden subgroup problem for poly-near-Hamiltonian groups*, Quant. Inf. Comp. **4** (2004), 229–235.

[18] R. Goodman and N. R. Wallach, *Representations and Invariants of the Classical Groups*, Cambridge University Press, Cambridge, 1998.

[19] M. Grigni, L. Schulman, M. Vazirani, and U. Vazirani, *Quantum mechanical algorithms for the nonabelian hidden subgroup problem*, Combinatorica **24** (2004), 137–154.

[20] L. K. Grover, *A fast quantum mechanical algorithm for database search*, Proc. 28th STOC, 1996, pp. 212–219.

[21] S. Hallgren, *Polynomial-time quantum algorithms for Pell's equation and the principal ideal problem*, Proc. 34th STOC, 2002, pp. 653–658.

[22] ———, *Fast quantum algorithms for computing the unit group and class group of a number field*, Proc. 37th STOC, 2005, pp. 468–474.
[23] L. Hales and S. Hallgren, *An improved quantum Fourier transform algorithm and applications*, Proc. 41st FOCS, 2000, pp. 515–525.
[24] S. Hallgren, C. Moore, M. Rötteler, A. Russell, and P. Sen, *Limitations of quantum coset states for graph isomorphism*, Proc. 38th STOC, 2006, pp. 604–617.
[25] S. Hallgren, A. Russell, and A. Ta-Shma, *The hidden subgroup problem and quantum computation using group representations*, Proc. 32nd STOC, 2000, pp. 627–635.
[26] A. W. Harrow, *Applications of coherent classical communication and the Schur transform to quantum information theory*, Ph.D. thesis, MIT, 2005.
[27] M. Hayashi and K. Matsumoto, *Quantum universal variable-length source coding*, Phys. Rev. A **66** (2002), 022311.
[28] G. Ivanyos, F. Magniez, and M. Santha, *Efficient quantum algorithms for some instances of the non-abelian hidden subgroup problem*, Int. J. Found. Comp. Sci. **14** (2003), 723–739.
[29] S. Kerov, *Gaussian limit for the Plancherel measure of the symmetric group*, Comptes Rendus Acad. Sci. Paris, Sér. I **316** (1993), 303–308.
[30] M. Keyl and R. F. Werner, *Estimating the spectrum of a density operator*, Phys. Rev. A **64** (2001), 052311.
[31] G. Kuperberg, *A subexponential-time quantum algorithm for the dihedral hidden subgroup problem*, SIAM J. Comput. **35** (2005), 170–188.
[32] C. Moore, D. N. Rockmore, and A. Russell, *Generic quantum Fourier transforms*, Proc. 15th SODA, 2004, pp. 778–787.
[33] C. Moore, D. N. Rockmore, A. Russell, and L. J. Schulman, *The hidden subgroup problem in affine groups: Basis selection in Fourier sampling*, Proc. 15th SODA, 2004, pp. 1113–1122.
[34] C. Moore, A. Russell, and L. J. Schulman, *The symmetric group defies strong Fourier sampling*, Proc. 46th FOCS, 2005, pp. 479–490.
[35] J. Radhakrishnan, M. Rötteler, and P. Sen, *On the power of random bases in Fourier sampling: Hidden subgroup problem in the Heisenberg group*, Proc. 32nd ICALP, LNCS **3580**, 2005, pp. 1399–1411.
[36] O. Regev, *Quantum computation and lattice problems*, Proc. 43rd FOCS, 2002, pp. 520–529.
[37] ———, *A subexponential time algorithm for the dihedral hidden subgroup problem with polynomial space*, quant-ph/0406151.
[38] A. Schmidt and U. Vollmer, *Polynomial time quantum algorithm for the computation of the unit group of a number field*, Proc. 37th STOC, 2005, pp. 475–480.
[39] P. Sen, *Random measurement bases, quantum state distinction and applications to the hidden subgroup problem*, quant-ph/0512085.
[40] J. P. Serre, *Linear Representations of Finite Groups*, Graduate Texts in Mathematics, vol. 42, Springer, New York, 1977.
[41] P. W. Shor, *Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer*, SIAM J. Comput. **26** (1997), 1484–1509.
[42] R. P. Stanley, *Theory and application of plane partitions*, Studies in Appl. Math. **1** (1971), 167–187 and 259–279.