

A Distribution Testing Oracle Separating QMA and QCMA

Anand Natarajan   

Massachusetts Institute of Technology, Cambridge, MA, USA

Chinmay Nirkhe   

IBM Quantum, Cambridge, MA, USA

Abstract

It is a long-standing open question in quantum complexity theory whether the definition of *non-deterministic* quantum computation requires quantum witnesses (QMA) or if classical witnesses suffice (QCMA). We make progress on this question by constructing a randomized classical oracle separating the respective computational complexity classes. Previous separations [3, 13] required a quantum unitary oracle. The separating problem is deciding whether a distribution supported on regular un-directed graphs either consists of multiple connected components (yes instances) or consists of one expanding connected component (no instances) where the graph is given in an adjacency-list format by the oracle. Therefore, the oracle is a distribution over n -bit boolean functions.

2012 ACM Subject Classification Theory of computation → Quantum computation theory

Keywords and phrases quantum non-determinism, complexity theory

Digital Object Identifier 10.4230/LIPIcs.CCC.2023.22

Related Version *Full Version:* [arXiv:2210.15380](https://arxiv.org/abs/2210.15380) [22]

Acknowledgements We thank Srinivasan Arunachalam, Andrew Childs, Elizabeth Crosson, Yi-Kai Liu, Aram Harrow, Zhiyang He, Robin Kothari, William Kretschmer, Yupan Liu, Kunal Marwaha, Mehdi Soleimanifar, Umesh Vazirani, and Elizabeth Yang for helpful discussions. Some of the early ideas of this result were developed while Chinmay Nirkhe was at the University of California, Berkeley. This work was partially completed while both authors were participants in the Simons Institute for the Theory of Computing program *The Quantum Wave in Computing: Extended Reunion*.

1 Introduction

There are two natural *quantum* analogs of the computational complexity class NP. The first is the class QMA in which a quantum polynomial-time decision algorithm is given access to a $\text{poly}(n)$ *qubit* quantum state as a witness for the statement. This class is captured by the QMA-complete local Hamiltonian problem [17] in which the quantum witness can be interpreted as the ground-state of the local Hamiltonian. The second is the class QCMA in which the quantum polynomial-time decision algorithm is given access instead to a $\text{poly}(n)$ *bit classical* state. While it is easy to prove that $\text{QCMA} \subseteq \text{QMA}$ as the quantum witness state can be immediately measured to yield a classical witness string, the question of whether $\text{QCMA} \stackrel{?}{=} \text{QMA}$, first posed by Aharonov and Naveh [4], remains unanswered. If $\text{QCMA} = \text{QMA}$, then every local Hamiltonian would have an efficient classical witness of its ground energy; morally, this can be thought of as an efficient classical description of its ground state. The relevance of local Hamiltonians to condensed matter physics makes this question a central open question in quantum complexity theory [2].



© Anand Natarajan and Chinmay Nirkhe;
licensed under Creative Commons License CC-BY 4.0
38th Computational Complexity Conference (CCC 2023).
Editor: Amnon Ta-Shma; Article No. 22; pp. 22:1–22:27



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



Because $P \subseteq QCMA \subseteq QMA \subseteq PSPACE$, any unconditional separation of the two complexity classes would imply $P \neq PSPACE$ and seems unlikely without remarkably ingenious new tools. A more reasonable goal is an oracle separation between the two complexity classes. The first oracle separation, by Aaronson and Kuperberg [3], showed that there exists a black-box unitary problem for which quantum witnesses suffice and yet no polynomial sized classical witness and algorithm can solve the problem with even negligible success probability. A second black-box separation was discovered a decade later by Fefferman and Kimmel [13]. The Fefferman and Kimmel oracle is a completely positive trace preserving (CPTP) map called an “in-place” permutation oracle. Both oracles [3, 13] are inherently quantum¹. Whereas, the “gold-standard” of oracle separations – namely black-box function separations (also known as classical oracle separations) – only require access to a *classical function* that can be queried in superposition².

1.1 Graph oracles

The major result of this work is to prove that there exists a distribution over black-box function problems separating QMA and QCMA. Each black-box function corresponds to the adjacency list of a $N \stackrel{\text{def}}{=} 2^n$ vertex constant-degree colored graphs³ $G = (V, E)$. Roughly speaking, a graph is a YES instance if the second eigenvalue of its normalized adjacency matrix is 1 (equivalently, if it has at least two connected components) and a graph is a NO instance if its second eigenvalue is at most $1 - \alpha$ for some fixed constant α (equivalently, the graph has one connected component and is expanding). We call this problem the *expander distinguishing problem*.

Distribution oracles

A distribution over functions (equivalently, a distribution over graphs) is a YES instance if it is entirely supported on YES graphs and a distribution over functions is a NO instance if it is entirely supported on NO graphs.

In this work, we construct, for every n , families of YES and NO distributions over graphs such that following hold for the promise problem of distinguishing a graph sampled from a YES distribution from a graph sampled from a NO distribution.

1. There is a QMA proof system that solves this problem, where the verifier runs in quantum polynomial time and has black-box query access to the sampled graph, and the honest prover’s (quantum) witness depends only on the distribution, not on the specific sample.
2. No QCMA proof system can solve this problem, provided the prover’s (classical) witness is only allowed to depend on the distribution, and not on the sample.

Our work is not the first to consider oracles that sample from distributions over functions. The in-place oracle separation of [13] between QMA and QCMA used oracles that sampled random permutations. For a somewhat different problem, of separating bounded-depth

¹ It might be reasonable to wonder if the unitary oracles can be converted into classical oracles by providing oracle access to the exponentially long classical descriptions of the respective matrices. This is not known to be true because it is unclear how to use access to the classical description to solve the QMA problem.

² One reason this model is natural is that if we were given a circuit of size C to implement this classical function, then we would automatically get a quantum circuit of size C to implement the oracle, simply by running the classical circuit coherently. This is not true for the “in-place” permutation oracle model, assuming that one-way functions exist.

³ A similar problem was previously conjectured to be an oracle separation for these complexity classes by Lutomirski [21].

■ **Table 1** List of known oracle separations.

Authors	Separating black box object	Proof techniques used
Aaronson & Kuperberg [3]	n -qubit unitaries	Adversary method
Fefferman & Kimmel [13]	n -qubit CPTP maps	Combinatorial argument, Adversary method
This work	Distributions over n -bit boolean functions	Combinatorial argument, Adversary method, Polynomial method
Conjectured	n -bit boolean function	?

quantum-classical circuits, [8] introduced a related notion called a “stochastic oracle” – the main difference between this and our model is that a stochastic oracle resamples an instance every time it is queried.

Comparison with previous oracle separations between QMA and QCMA

Table 1 summarizes our work in relation to previous oracle separations. In terms of results, we take a further step towards the standard oracle model – all that remains is to remove the randomness from our oracle. In terms of techniques, we combine the use of counting arguments and the adversary method from previous works with a BQP lower bound for a similar graph problem, due to [6]. This lower bound was shown using the polynomial method. We view the judicious combination of these lower bound techniques – as simple as it may seem – as one of the conceptual contributions of this paper.

Intuition for hardness

The expander distinguishing problem is a natural candidate for a separation between QMA and QCMA because it is an “oracular” version of the sparse Hamiltonian problem, which is complete for QMA [10, Problem H-4]. To see this, we recall some facts from spectral graph theory. The top eigenvalue of the normalized adjacency matrix A for regular graphs is always 1 and the uniform superposition over vertices is always an associated eigenvector. If the graph is an expander (the NO case of our problem), the second eigenvalue is bounded away from 1, but if the graph is disconnected (the YES case of our problem), then the second eigenvalue is exactly 1. Thus, our oracle problem is exactly the problem of estimating the minimum eigenvalue of $\mathbb{I} - A$ (a sparse matrix for a constant-degree graph), on the subspace orthogonal to the uniform superposition state. Viewing $\mathbb{I} - A$ as a sparse Hamiltonian, we obtain the connection between our problem and the sparse Hamiltonian problem.

One reason to show oracle separations between two classes is to provide a *barrier* against attempts to collapse the classes in the “real” world. We interpret our results as confirming the intuition that any QCMA protocol for the sparse Hamiltonian must use more than just black-box access to entries of the Hamiltonian: it must use some nontrivial properties of the ground states of these Hamiltonians. In this sense, it emulates the original quantum adversary lower bound of [9] which showed that any BQP-algorithm for solving NP-complete problems must rely on some inherent structure of the NP-complete problem as BQP-algorithms cannot solve unconstrained search efficiently.

Naturalness of the randomized oracle model

Some care must be taken whenever one proves a separation in a “nonstandard” oracle model – see for instance the “trivial” example in [1] of a randomized oracle separating MA_1 from MA . We believe that our randomized oracle model is natural for several reasons. Firstly, as mentioned above, randomization was used in the quantum oracle of [13] for essentially the

same reason: to impose a restriction on the witnesses received from the prover. Secondly, it is consistent with our knowledge that our oracle separates QMA from QCMA even when the randomness is removed (and indeed we conjecture this is the case, as described below.) Thirdly, the randomization still gives the prover access to substantial information about the graph: in particular, the prover knows the full connected component structure of the graph. As we show, this information is enough for the prover to give a *quantum* witness state, that in the YES case convinces the verifier with certainty. Our result shows that even given full knowledge of the component structure, the prover cannot construct a convincing classical witness – we believe this sheds light on how a QMA witness can be more powerful than a QCMA witness.

1.2 Overview of proof techniques

Quantum witnesses and containment in oracular QMA

A quantum witness for any YES instance graph is any eigenvector $|\xi\rangle$ of eigenvalue 1 that is orthogonal to the uniform superposition over vertices. The verification procedure is simple: project the witness into the subspace orthogonal to the uniform superposition over vertices, and then perform one step of a random walk along the graph, by querying the oracle for the adjacency matrix in superposition. Verify that the state after the walk step equals $|\xi\rangle$. This is equivalent to a 1-bit phase estimation of the eigenvalue. If a graph is a NO instance, then there does not exist any vector orthogonal to the uniform superposition (the unique eigenvector of value 1) that would pass the previous test.

Whenever, the graph has a connected component of $S \subsetneq V$, then an eigenvector orthogonal to the uniform superposition of eigenvalue 1 exists. When $|S| \ll N$, this eigenvector is very close to $|S\rangle$, the uniform superposition over basis vectors $x \in S$. Notice that this state only depends on the connected component S and not the specific edges of the graph. Furthermore, the state $|S'\rangle$ for any subset S' that approximates S forms a witness that is accepted with high probability.

Lower bound on classical witnesses

The difficulty in this problem lies in proving a *lower bound* on the ability for classical witnesses to distinguish YES and NO instances. To prove a lower bound, we argue that any quantum algorithm with access to a polynomial length classical witness must make an exponential number of (quantum) queries to the adjacency list of the graph in order to distinguish YES and NO instances. This, in turn, lower bounds the time complexity of any QCMA algorithm distinguishing YES and NO instances but is actually slightly stronger since we don't consider the computational complexity of the algorithm between queries.

Proving lower bounds when classical witnesses are involved is difficult because the witness could be based on any property of the graph. For example, the classical witness could describe cycles, triangles, etc. contained in the graph – while it isn't obvious why such a witness would be helpful, proving that any such witness is insufficient is a significant challenge. One way to circumvent this difficulty is to first show a lower bound *assuming* some structure about the witness⁴, and then “remove the training wheels” by showing that the assumption holds for any good classical witness.

⁴ Assuming structure about a witness is a common technique in theoretical computer science and in particular lower bounds for classical witnesses of quantum statements. For example, lower bounds against natural proofs [19]. Another example is the NLTS statement [7] which is about lower bounds for classical witnesses for the ground energy of a quantum Hamiltonian of a particular form: constant-depth quantum circuits.

Lower bound against “subset witnesses”

One structure we can assume is that the witness only depends on the set of vertices contained in the connected component S . This is certainly the case for the ideal quantum witness state. Our result shows that any polynomial-length witness only depending on the vertices in S requires an exponential query complexity to distinguish YES and NO instance graphs.

The starting point for this statement is the exponential query lower bound *in the absence of a witness* (i.e. for BQP) for the expander distinguishing problem proven by Ambainis, Childs and Liu [6], using the polynomial method. In [6], the authors define two distributions over constant-degree regular colored graphs: the first is a distribution P_1 over random graphs with overwhelming probability of having a second normalized eigenvalue at most $1 - \epsilon_0$. The second is a distribution P_ℓ over random graphs with overwhelming probability of having ℓ connected components. Since, almost all graphs in P_1 are NO graphs and all graphs in P_ℓ are YES graphs, any algorithm distinguishing YES and NO instances must be able to distinguish the two distributions. We first show that a comparable query lower bound still holds even when the algorithm is given a witness consisting of polynomially many random points F from any one connected component.

Next, we show that if there were a QCMA algorithm where the optimal witness depends only on the set of vertices S in one of the connected components, by a counting argument, there must exist a combinatorial *sunflower* of subsets S that correspond to the same witness string. A *sunflower*, in this context, is a set of subsets such that each subset contains a core $F \subset V$ and every vertex of $V \setminus F$ occurs in a small fraction of subsets. This implies that there exists a BQP algorithm which distinguishes YES instances corresponding to the sunflower from all NO instances. Next, we show using an adversary bound [5], a quantum query algorithm cannot distinguish the distribution of YES instances corresponding to the sunflower from the uniform distribution of YES instances such that the core F is contained in a connected component (the ideal sunflower).

This indistinguishability, along with the previous polynomial method based lower bound, proves that QCMA algorithm – whose witness only depends on the vertices in the connected component – for the expander distinguishing problem must make an exponential number of queries to the graph.

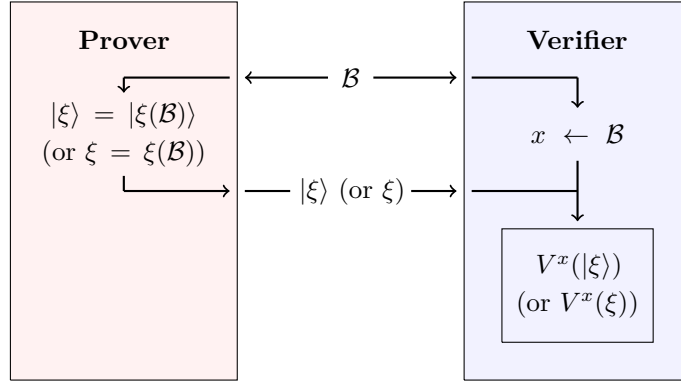
Removing the restriction over witnesses

Our proof, thus far, has required the restriction that the witness only depends on the vertices in the connected component. In some sense, this argues that there is an oracle separation between QMA and QCMA if the prover is restricted to being “near-sighted”: it cannot see the intricacies of the edge-structure of the graph, but can notice the separate connected components of the graph. If the near-sighted prover was capable of sending quantum states as witnesses, then she can still aid a verifier in deciding the expander distinguishing problem, whereas if she could only send classical witnesses, then she cannot aid a verifier.

It now remains to remove the restriction that the witness can only depend on the vertices in the connected component. We do this by introducing *randomness* into the oracle, precisely designed to “blind” the prover to the local structure of the graph. In the standard oracle setting, the verifier and prover both get access to an oracle $x \in \{0, 1\}^N$, and the prover provides either a quantum witness, $|\xi(x)\rangle \in (\mathbb{C}^2)^{\otimes \text{poly}(n)}$ or a classical witness, $\xi(x) \in \{0, 1\}^{\text{poly}(n)}$. The verifier then runs an efficient quantum algorithm V^x which takes as input $|\xi(x)\rangle$ (or $\xi(x)$, respectively) and consists of quantum oracle gates applying the unitary transform defined as the linear extension of

$$|i\rangle \mapsto (-1)^{x_i} |i\rangle \text{ for } i \in [N]. \quad (1)$$

We now modify this setup slightly. Instead of a single oracle x , we consider a distribution \mathcal{B} over oracles. The prover constructs a quantum witness $|\xi(\mathcal{B})\rangle$ (or a classical witness $\xi(\mathcal{B})$, respectively) based on the distribution \mathcal{B} . The verifier then samples a classical oracle $x \leftarrow \mathcal{B}$ from the distribution, and then runs the verification procedure V^x which takes as input $|\xi(\mathcal{B})\rangle$ (or $\xi(\mathcal{B})$, respectively) and applies quantum oracle gates corresponding to x . The success probability of the verifier is taken over the distribution \mathcal{B} and the randomness in the verification procedure.



■ **Figure 1** Cartoon of interaction between Prover and Verifier for a distribution over classical boolean functions.

From our previous observations, graphs with the same connected component S have the same ideal witness state. So, if the distribution \mathcal{B} is supported on all graphs with the same connected component S , then the ideal witness state suffices. Furthermore, in the case of the classical witness system, the witness can only depend on S and the previously stated lower bound applies. This motivates the oracle problem of distinguishing distributions, marked either YES or NO, over 2^n bit strings (or equivalently n -bit functions).

1.3 Statement of the result

► **Theorem 1.** *For every sufficiently large integer n that is a multiple of 200, there exist distributions over 100-regular 100-colored graphs on $N = 2^n$ vertices labeled either YES or NO such that*

- *Each YES distribution is entirely supported on YES instances of the expander-distinguishing problem and, likewise, each NO distribution is entirely supported on NO instance of the expander-distinguishing problem.*
- *There exists a $\text{poly}(n)$ time quantum algorithm V_q taking a witness state $|\xi\rangle$ as input and making $O(1)$ queries to the quantum oracle such that*

1. *For every YES distribution \mathcal{B} , there exists a quantum witness $|\xi\rangle \in (\mathbb{C}^2)^{\otimes n}$ such that*

$$\mathbf{E}_{x \leftarrow \mathcal{B}} \Pr[V_q^x(|\xi\rangle) \text{ accepts}] = 1. \quad (2)$$

2. *For every NO distribution \mathcal{B} , for all quantum witnesses $|\xi\rangle \in (\mathbb{C}^2)^{\otimes n}$,*

$$\mathbf{E}_{x \leftarrow \mathcal{B}} \Pr[V_q^x(|\xi\rangle) \text{ accepts}] \leq 0.01. \quad (3)$$

- Any quantum algorithm V_c accepting a classical witness of length $q(n)$ satisfying the following two criteria either requires $q(n)$ to be exponential or must make an exponential number of queries to the oracle⁵.

1. For every YES distribution \mathcal{B} , there exists a classical witness $\xi = \xi(\mathcal{B}) \in \{0, 1\}^{q(n)}$

$$\mathbf{E}_{x \leftarrow \mathcal{B}} \Pr[V_c^x(\xi) \text{ accepts}] \geq 0.99. \quad (4)$$

2. For every NO distribution \mathcal{B} , for all classical witnesses $\xi \in \{0, 1\}^{q(n)}$,

$$\mathbf{E}_{x \leftarrow \mathcal{B}} \Pr[V_c^x(\xi) \text{ accepts}] \leq 0.01. \quad (5)$$

Although our main theorem is formulated as a query lower bound, it can be converted to a separation between the relativized classes of QMA and QCMA via a standard diagonalization argument. Similarly, it was pointed out to us [14] that it proves a separation between the relativized classes of BQP/qpoly and BQP/poly, following the technique of [3].

1.4 Implications and future directions

There are several future questions raised by this work that we find interesting:

Oracle and communication separations

The most natural question is, of course, whether the oracle's randomness can be removed to obtain a separation in the standard model. We conjecture that our problem yields such a separation, but a new technique seems necessary to prove it. See Section 9 for more details on the technical barriers to derandomizing our construction.

Another natural question is to show a *communication complexity* separation between QMA and QCMA. This has been shown for one-way communication by Klauck and Podder [18] but their problem does not yield a separation for two-way communication. Could our query separation be lifted to the communication world by use of the appropriate gadget?

The class QMA(2) is another relative of QMA which is perhaps even more enigmatic than QCMA. In QMA(2), the witness state is promised to be unentangled between the first and second half of the qubits. We do not even know of a quantum (unitary) oracle separation between QMA(2) and QMA, nor do we have a natural candidate problem. Could we at least formulate such a candidate by considering “oracular” versions of QMA(2)-complete problems, in analogy to what we do in this work for QCMA.

Search-to-decision

In [15], Irani, Natarajan, Nirkhe, Rao and Yuen studied the complexity of generating a witness to a QMA problem (equivalently, generating a ground state of a local Hamiltonian) when given *oracle access* to a QMA oracle. This paradigm, called *search-to-decision*, is commonplace in classical complexity theory (for example, P, NP, MA, etc. all have search-to-decision reductions) and yet [15] gives evidence that QMA likely does not exhibit a search-to-decision reduction. They prove this by showing an oracle relative to which QMA search-to-decision reductions are provably impossible. The oracle used is identical to that of Aaronson and

⁵ We leave it open whether the completeness-soundness gap in this lower bound can be improved to an inverse polynomial. Naive gap amplification for QCMA do not work in the distribution testing oracle setting; see the note below Definition 4 for details.

Kuperberg [3] to separate QMA and QCMA. [15] acknowledge this noncoincidence and conjecture whether *any* QMA and QCMA separating oracle yields a QMA search-to-decision impossibility result. Similar to the reasons for why the gold-standard of oracle separation between QMA and QCMA is a n -bit boolean function, the ideal oracle for proving QMA search-to-decision impossibility is also a n -bit boolean function. Does the oracle presented here also yield a search-to-decision impossibility?

Implications for Quantum PCPs

The quantum PCP conjecture [4] is one of the biggest open questions in quantum complexity theory. In a recent panel [24] on the quantum PCP conjecture and the NLTS theorem [7], an interesting question was posed of whether MA or QCMA (lower or upper) bounds can be placed on the complexity of the promise-gapped local Hamiltonian problem. We recommend [23] for an introduction to the subject. Because the oracle presented in this result corresponds to a sparse Hamiltonian with a problem of deciding if the second eigenvalue of the Hamiltonian is 1 or $< 1 - \alpha/d = 1 - \Omega(1)$, one might wonder if this provides oracular evidence that quantum PCPs are at least QCMA-hard. Unfortunately, to the best of our knowledge, this is not a reasonable conclusion. While we give evidence that the promise-gapped *sparse* Hamiltonian problem is likely QCMA-hard, the reduction from the sparse Hamiltonian problem to the local Hamiltonian problem does not imply that the promise-gapped local Hamiltonian problem is likely QCMA-hard. The only algorithm known for checking a witness for the sparse Hamiltonian problem is Hamiltonian simulation on the witness which is not a local algorithm.

Connections to Stoquastic Hamiltonians

Since the oracles studied in this work correspond to the adjacency lists of graphs, they can be viewed as sparse access to a Hamiltonian H which is the Laplacian of a graph (recall that if the adjacency matrix is A , then the Laplacian is $\mathbb{I} - A/d$). Such Hamiltonians have a special structure not present in general Hamiltonians: they are *stoquastic*, meaning that the off-diagonal entries are nonpositive. The local Hamiltonian (LH) problem for stoquastic Hamiltonians is significantly easier than the general LH problem, and in some cases is even contained in MA as shown by Bravyi and Terhal [12]. It is worth noticing why this is not in tension with our result – in particular, why this does not imply that our oracle problem is contained in oracular MA.

- Crucially, the MA-containment for stoquastic LH holds *only* for the ground state: this is because of the Perron-Frobenius theorem, which implies that ground states of such Hamiltonians have nonnegative coefficients. However, in our case, we want the first excited state: the state of minimum energy for H restricted to the subspace orthogonal to the uniform superposition. It was shown by [16] that all excited state energies are QMA-hard to calculate for a stoquastic Hamiltonian.
- The MA containment also uses the locality of the Hamiltonian, which in turn imposes a strong structure on the adjacency matrix of the graph. The random graphs we consider will not have this structure. (While it was shown by [11] showed an AM algorithm for calculating the ground energy *stoquastic and sparse* Hamiltonians, again this does not apply to higher excited states.)
- At an intuitive level, in graph language, the LH problem for stoquastic Hamiltonians is to find a component of the graph where the average value of some *potential function* (given by the diagonal entries of H) is minimized. An MA verifier can solve this by executing a

random walk, given the right starting point by Merlin. In contrast, our problem is to determine whether the graph as a whole is connected – a global property which an MA verifier cannot determine.

2 Organization of the paper

The remainder of the paper is the proof of Theorem 1. The proof is divided into smaller components and these intermediate results are joined together in Section 8. In Section 3, we state some basic definitions and formally define the expander distinguishing problem. In Section 4, we describe the distributions over graphs that constitute YES and NO instances. In Section 5, we prove that there is an efficient QMA algorithm for the expander distinguishing problem. In particular, there is a single quantum witness that serves all the graphs in each of the YES distributions. In Section 6, we use the adversary method and counting arguments to prove that any QCMA algorithm for the expander distinguishing problem for the constructed distributions implies a BQP algorithm for distinguishing YES instances with a connected component corresponding to an ideal sunflower from a generic NO instance. In Section 7, we argue using the polynomial method that such an algorithm is impossible without an exponential query complexity. In Section 9, we present some concluding remarks about our construction and its relation to other notions of computational complexity.

3 Preliminaries

3.1 Notation and quantum information basics

We will assume that the reader is familiar with the basics of quantum computing and quantum information. We will use $N \stackrel{\text{def}}{=} 2^n$ throughout this paper and we will only consider graphs of N vertices. The adjacency list of a d -regular d -colored graph on N vertices takes dnN bits to describe. For any m , we abbreviate the set of integers $\{1, 2, \dots, m\}$ as $[m]$. For a set $A \subseteq [N]$, we will use $|A\rangle$ to denote the state $\frac{1}{\sqrt{|A|}} \sum_{j \in A} |j\rangle$, the subset state corresponding to A . Unless otherwise specified we assume $\|\cdot\|$ is the Euclidean norm $\|\cdot\|_2$ for a vector, and the spectral norm for a matrix, which is the largest singular value.

3.2 Expander graphs

► **Definition 2.** A graph G is a spectral α -expander (equiv. is α -expanding) if the second highest eigenvalue λ_2 of the normalized adjacency matrix of G satisfies $\lambda_2 \leq 1 - \alpha$. We say that a connected component S of the graph is α -expanding if the restricted graph to the vertices of S is α -expanding.

► **Lemma 3.** Let G be a d -regular α -expander. Consider the random walk that starts in any distribution over the vertices, and at each time step, stays in place with probability $1/2$, and moves along an edge of the graph with probability $1/2$. Then for any vertex v , after ℓ steps, the probability $\Pr[v]$ that the walk is in v satisfies

$$\left| \Pr[v] - \frac{1}{N} \right| \leq \left(1 - \frac{\alpha}{2}\right)^\ell. \quad (6)$$

In particular, when $\ell = O(c \log N/\alpha)$ we can get the RHS to be $1/N^c$.

Proof. Proof is available in the full version [22]. ◀

3.3 Non-deterministic oracle problems

► **Definition 4** (Quantum oracle problems). For a n -bit boolean function \mathcal{O} , we say an oracle decision problem $\mathcal{L}^{\mathcal{O}}$ is in $\text{QMA}^{\mathcal{O}}(\epsilon)$ if there exists a uniform family of quantum circuits $A^{\mathcal{O}}$ such that

1. For every YES instance \mathcal{O} , there exists a quantum state $|\xi\rangle$ of $\text{poly}(n)$ qubits such that $A^{\mathcal{O}}(|\xi\rangle)$ accepts with probability $\geq 1 - \epsilon$.
2. For every NO instance \mathcal{O} , for all quantum states $|\xi\rangle$ of $\text{poly}(n)$ qubits, $A^{\mathcal{O}}(|\xi\rangle)$ accepts with probability $\leq \epsilon$.

$\text{QCMA}^{\mathcal{O}}(c, s)$ is defined similarly, except the state $|\xi\rangle$ is promised to be classical. The classes $\text{QMA}^{\mathcal{O}}$ and $\text{QCMA}^{\mathcal{O}}$ are defined as $\text{QMA}^{\mathcal{O}}(1/3)$ and $\text{QCMA}^{\mathcal{O}}(1/3)$, respectively.

We note that due to parallel repetition, $\text{QMA}^{\mathcal{O}}(\epsilon = \frac{1}{2} - 1/\text{poly}(n)) = \text{QMA}^{\mathcal{O}} = \text{QMA}^{\mathcal{O}}(\epsilon = 2^{-\text{poly}(n)})$. Likewise, for $\text{QCMA}^{\mathcal{O}}$. This justifies removing the constant ϵ from the definition. We now define the same problem for oracles equaling distributions over n -bit boolean functions.

► **Definition 5** (Random classical oracles). A random oracle \mathcal{R} is a distribution over classical oracles $\{\mathcal{O}\}$. We say an oracle decision problem $\mathcal{L}^{\mathcal{R}}$ is in $\text{QMA}^{\mathcal{R}}(\epsilon)$ if there exists a uniform family of quantum circuits $A^{\mathcal{O}}$ such that

1. For every YES instance \mathcal{R} , there exists a quantum state $|\xi\rangle$ of $\text{poly}(n)$ qubits such that

$$\mathbf{E}_{\mathcal{O} \in \mathcal{R}} \Pr [A^{\mathcal{O}}(|\xi\rangle) \text{ accepts}] \geq 1 - \epsilon. \quad (7)$$

2. For every NO instance \mathcal{O} , for all quantum states $|\xi\rangle$ of $\text{poly}(n)$ qubits,

$$\mathbf{E}_{\mathcal{O} \in \mathcal{R}} \Pr [A^{\mathcal{O}}(|\xi\rangle) \text{ accepts}] \leq \epsilon. \quad (8)$$

$\text{QCMA}^{\mathcal{R}}(c, s)$ is defined similarly, except the state $|\xi\rangle$ is promised to be classical.

Ideally, we would define the classes $\text{QMA}^{\mathcal{R}}$ and $\text{QCMA}^{\mathcal{R}}$ are defined as $\text{QMA}^{\mathcal{R}}(1/3)$ and $\text{QCMA}^{\mathcal{R}}(1/3)$, respectively. However, the parallel repetition argument for boolean function oracles cannot be extended to distributions over boolean functions. This is because the ϵ error that an algorithm is the expectation of the success probability of the algorithm over the distribution. It is possible that the algorithm runs on every instance in the distribution with error ϵ or it is possible that the algorithm succeeds with 0 error on a $1 - \epsilon$ fraction of the distribution and fails on the remaining ϵ fraction. In the first case, the success of the algorithm can be improved with parallel repetition while it cannot in the second case⁶.

3.4 Graph oracles

► **Definition 6** (Colored Graphs). Given a d -colored d -regular graph $G = (V, E)$ on N vertices, we say G contains a triple $(j_1, j_2, \kappa) \in V^2 \times [d]$ if the edge (j_1, j_2) exists in G and is colored with color κ .

⁶ We note that this subtlety is overlooked in Fefferman and Kimmel [13] but we believe that their result without parallel repetition is correct. Furthermore, the adversary bounds used in [13] do not address this issue but can be rectified using the adversary bound stated in Theorem 13 which deals with *average-case* distinguishing.

► **Definition 7** (Adjacency graph oracles). *Let G be a d -colored d -regular undirected graph. The graph $G = (V, E)$ can be described by an adjacency function*

$$G : V \times [d] \rightarrow V \quad (9)$$

where the output of (j, κ) returns the neighbor of j along the edge colored with κ . Quantum access to the function G is provided by the following oracle unitary:

$$|j, \kappa, z\rangle \xrightarrow{G} |j, c, z \oplus G(j, \kappa)\rangle. \quad (10)$$

We call the function G the adjacency graph oracle corresponding to G .

► **Definition 8** (Expander distinguishing problem). *The (α, ζ) -expander distinguishing problem is a promise oracle language where the input is an oracle G for a d -colored d -regular undirected graph G on N vertices. The problem is to distinguish between the following two cases, promised that one holds:*

- YES: the graph G has a connected component S of size at most $|S| \leq \zeta$.
- NO: the graph G is an α -expander.

In this paper, we will think of α as a constant and $\zeta \sim N^{9/10}$. To simplify notation, since the oracles considered in this result always correspond to graphs G , we express the algorithm as \mathcal{A}^G rather than $\mathcal{A}^{\mathcal{O}}$.

4 Random distributions over graphs with many connected components

In this subsection, we describe distributions over graphs where the graphs with high probability consist of ℓ connected components. It should not be surprising that the distribution is almost identical to the distribution used by Ambainis, Childs, and Liu [6] in their proof that the expander distinguishing problem requires an exponential number of quantum queries for any quantum query algorithm in the absence of a proof. This is because we will reduce any QCMA algorithm to an efficient query algorithm for some expander distinguishing problem.

The lower bound in [6] is crucially a lower bound on the *polynomial degree* of any polynomial that distinguishes two graph distributions. From there, it isn't too much to argue that these graph distributions are very close to YES and NO instances as prescribed in the expander distinguishing problem; therefore any algorithm solving the expander distinguishing problem must be able to distinguish these two graph distributions. Our first goal is to amplify the argument of [6] to a more restricted class of graphs.

4.1 Graphs distributions inspired by [6]

The goal of the construction is a distribution which depends on an integer ℓ and a subset $F \subset V$. The integer ℓ will roughly correspond to the number of connected components (henceforth denoted C_1, \dots, C_ℓ) in the graph and we insist that $F \subset C_1$. Every $v \in V \setminus F$ appears in each subset C_i with equal probability of $1/\ell$. The actual construction will be slightly more complicated than this but, morally, this is what we hope to achieve from the distribution.

Formal construction

Let N be an integer and for integer $M \geq N$, integer ℓ dividing M and a subset $F \subset V$ define the distribution $P_{M, \ell}(F)$ over graphs on N vertices as follows:

1. Start by constructing a graph G' on M vertices: Partition V' into ℓ equally sized sets of vertices V_1, \dots, V_ℓ . On each subset V_k , create a random *colored* subgraph by randomly choosing d perfect matchings (each with a different color $1, \dots, d$) and taking their union.

2. To construct the graph G on N vertices: We first choose an injective map $\iota : V \hookrightarrow V'$. First, we pick a function $k : V \rightarrow [\ell]$. We pick k as a uniformly random function conditioned on the fact that $k(j) = 1$ for each $j \in F$. Let $\iota(j)$ be a random vertex from $V_{k(j)}$ *without replacement* to satisfy injectivity. If all vertices from $V_{k(j)}$ have been selected with replacement, output the graph on N vertices with no edges (i.e. abort).
3. Induce a graph G on V from G' and the map ι – i.e. an edge $(j_1, j_2, \kappa) \in V^2 \times [d]$ exists if $(\iota(j_1), \iota(j_2), \kappa) \in V'^2 \times [d]$ is an edge.
4. For a vertex j and a color κ , if the previous induced edges did not introduce a κ -colored edge from j , then add edge (j, j, κ) .
5. The distribution over graphs G is henceforth called $P_{M,\ell}(F)$; when $F = \emptyset$, we write it as $P_{M,\ell}$.

Notationally, for edges $e = (u, v, \kappa) \in G$, we will use $\iota(e) = (\iota(u), \iota(v), \kappa) \in G'$. Furthermore, we will extend ι naturally to subgraphs and subsets of vertices and edges.

► Remark 9. For any F , $P_{M,1}(F) = P_{M,1}(\emptyset) \stackrel{\text{def}}{=} P_{M,1}$.

4.2 Setting of constants

The lower bounds we prove for the QCMA algorithm are by no means tight (up to constants). We make no attempt to perfect the choice of constants as our only goal is to prove an exponential lower bound on the size of the any quantum witness or the number of queries required to solve the expander distinguishing problem. For this reason, we pick the following constants:

Chosen constants

The degree of the graph G' is set to be $d = 100$. We assume $\ell = N^{1/10}$, $\gamma = N^{-1/10}$ and $M = (1 + \gamma)N$.

Induced constants

In Definition 8, we define an (α, ζ) -expander distinguishing problem. We will only consider $\alpha = 1/(2 \cdot 10^8)$ (which is a consequence of Lemma 10 and the chosen constants). Notationally, we will use $z \stackrel{\text{def}}{=} N/\ell = N^{9/10}$. We will use $\zeta = (1 + \gamma)z = M/\ell$.

Conventions

Typically, we will assume (for the purposes of contradiction) that $|F| \leq N^{1/100}$ but as that is a term we wish to bound, we explicitly state it each time. Anytime a set S is described, it will be of size ζ , but we will also state this.

4.3 Concentration bounds for random distributions over graphs

We will need the following concentration lemma about the generated distributions. The lemma proves that $P_{M,1}$ is approximately a YES instance and that $P_{M,\ell}(F)$ is approximately a NO instance. Overall, this lemma proves that any algorithm solving the expander distinguishing problem must do very well on identifying the distribution $P_{M,1}$ as a NO instance and identifying the distributions $P_{M,\ell}(F)$ as a YES instance. The proof of this lemma is provided in Appendix of the full version [22].

► **Lemma 10** (Adaptation of Lemma 16 of [6]). *Assume $|F| \leq N^{1/100}$. Then with probability at least $\geq 1 - O(N^{-3})$, a graph drawn from distribution $P_{M,\ell}(F)$ consists of exactly ℓ connected components each α -expanding and consisting of between $(1 - \gamma)z$ and $(1 + \gamma)z$ vertices.*

Likewise, the probability that a graph drawn from the distribution $P_{M,1}$ is α -expanding is $\geq 1 - O(N^{-3})$.

Note that being expanding necessarily implies connectivity. Note that when $F = \emptyset$ or $\ell = 1$, there are simpler proofs with tighter bounds but the bound proven here for the general statement is sufficient for our result.

The second concentration lemma that we will use is that $P_{M,\ell}$ is approximately equal to sampling a set F of size $\leq N^{1/100}$ and then sampling a graph from $P_{M,\ell}(F)$. The proof of this lemma is also provided in Appendix of the full version [22].

► **Lemma 11.** *Let $m \leq N^{1/100}$. Let \mathcal{D}_1 be the distribution on pairs (G, F) obtained by sampling $G \sim P_{M,\ell}$, choosing a uniformly random vertex $v \in G$, and then choosing F to be a uniformly random subset of the connected component of G containing v of size m . Let \mathcal{D}_2 be the distribution on pairs (G, F) obtained by first choosing F to be a uniformly random subset of V with size m , and then sampling $G \sim P_{M,\ell}(F)$. Then these distributions are close in statistical distance:*

$$\|\mathcal{D}_1 - \mathcal{D}_2\| \leq 3N^{-9/200}. \quad (11)$$

5 QMA protocol

In this section we show that the expander distinguishing problem (over a fixed graph – i.e., no distribution) can be solved with a polynomial number of queries (indeed, with just two queries) if a quantum witness is provided. Our algorithm has the added benefit of being time-efficient, so we have shown that this problem is contained in QMA^G . In Section 8, we prove that there still exists a QMA protocol if we consider distribution oracles.

► **Lemma 12.** *There is a QMA^G protocol \mathcal{A}_{QMA} that solves the (α, ζ) -expander distinguishing problem with the following properties:*

1. **Query complexity:** *the algorithm makes two queries to G .*
2. **Completeness:** *In the YES case, there exists a witness state that the verifier accepts with certainty.*
3. **Soundness:** *In the NO case, no witness state is accepted by probability greater than $1 - \alpha/4$.*
4. **Nice witnesses:** *In the YES case, if $S \subsetneq V$ is a connected component of the graph G , then the state*

$$|S\rangle = \frac{1}{\sqrt{|S|}} \sum_{v \in S} |v\rangle$$

is accepted with probability at least $1 - \sqrt{|S|/N}$. In particular, since there exists a connected component of size at most ζ , there is a state of this form that is accepted with probability $1 - \sqrt{\zeta/N}$.

Proof. At a high level, the verifier performs one-bit phase estimation of the adjacency matrix of G on the witness state; this verifies that the witness was an eigenvector of optimal eigenvalue of the graph's adjacency matrix. A complete proof is available in the full version [22]. ◀

6 Adversary method

In this section, we use the adversary method of Ambainis to argue that any successful QCMA algorithm implies a BQP algorithm for distinguishing the distribution $P_{M,\ell}(F)$ and $P_{m,1}$ for $|F| \leq N^{1/100}$. In Section 6.1, we state the adversary method result and in the following sections we prove the statement.

6.1 Ambainis' proof of the adversary method

The adversary method of Ambainis [5] is a convenient way of arguing lower bounds on the query complexity of oracular quantum algorithms. The adversary method lower bounds the complexity of any algorithm which (with high probability) computes $f(a)$ for a function $f : \{0,1\}^N \rightarrow \{0,1\}$. The quantum algorithm is allowed access to $a \in \{0,1\}^N$ by a oracle gate O which applies linearly the transform $|i\rangle \mapsto (-1)^{a_i} |i\rangle$ for $i \in \{0,1\}^n$ (here $N = 2^n$). In doing so, the adversary method is a convenient way of producing BQP (query) lower-bounds.

To use it in our distributional setting, we make two modifications to the adversary bound. The first is to relax the notion of correctness. The lower bound of Ambainis is for a lower-bound for any algorithm which, for *each* $a \in \{0,1\}^N$, outputs $f(a)$ correctly with probability $1 - \epsilon$ for $\epsilon < \frac{1}{2}$. We instead consider an *average-case* notion of success in which

$$\mathbf{E}_{a \in \{0,1\}^N} \Pr_{\mathcal{A}} [\mathcal{A}^a \neq f(a)] \leq \epsilon^2. \quad (12)$$

By Markov's inequality, this implies

$$\Pr_{a \in \{0,1\}^N} \left[\Pr_{\mathcal{A}} [\mathcal{A}^a \neq f(a)] \geq \epsilon \right] \leq \epsilon, \quad (13)$$

or in other words, most a are (with high probability) correctly identified.

The second modification is to *restrict* the set of locations that the algorithm is allowed to query the oracle. The reason for this is somewhat subtle. Essentially, the original lower bound of Ambainis was designed for decision problems with deterministic oracles, and relies on constructing a relation between two *disjoint* sets of oracle instances, one consisting only of YES instances and the other only of NO instances. However, in our setting, we are interested in distinguishing two distributions over oracles that may have overlapping support. In order to define disjoint YES and NO sets of instances even when the distributions overlap, we add to each oracle string a a set of flag bits b that indicate which of the two distributions the string a was sampled from. Naturally, any reasonable model cannot permit the algorithm to query the flag bits: otherwise, it would be easy to distinguish even two statistically close distributions with few queries.

More formally, we consider a generalization where the oracle string is a tuple $(a, b) \in \{0,1\}^N \times \{0,1\}^M$ and $f : \{0,1\}^{N+M} \rightarrow \{0,1\}$ but the algorithm can only query positions of a . In this model, with the average-case notion of success defined above, we obtain the following adversary lower bound for distributions:

► **Theorem 13.** *Let $f : \{0,1\}^{N+M} \rightarrow \{0,1\}$ be a function and let $X, Y \subset \{0,1\}^N \times \{0,1\}^M$ be two subsets such that $X \subset f^{-1}(0)$ and $Y \subset f^{-1}(1)$. Let $R \subset X \times Y$ be a relation such that*

1. *For every $x \in X$, let $R_x \subset Y$ equal $R_x = \{y : (x, y) \in R\}$ such that $\underline{m} \leq |R_x| \leq \overline{m}$.*
2. *For every $y \in Y$, let $R_y \subset X$ equal $R_y = \{x : (x, y) \in R\}$ such that $\underline{m}' \leq |R_y| \leq \overline{m}'$.*
3. *For every $x = (a, b) \in X$ and $i \in [N]$, let $\ell_{x,i}$ be the number of $y = (c, d) \in Y$ such that $(x, y) \in R$ and $a_i \neq c_i$. Likewise, for every $y = (c, d) \in Y$ and $i \in [N]$, let $\ell_{y,i}$ be the number of $x = (a, b) \in X$ such that $(x, y) \in R$ and $a_i \neq c_i$. Let ℓ_{\max} be the maximum product $\ell_{x,i} \ell_{y,i}$ over $(x, y) \in R$ and $i \in [N]$ such that $a_i \neq c_i$.*

Then any quantum algorithm \mathcal{A} which only queries the first N bits of the oracle and computes f such that

$$\mathbf{E}_{x=(a,b) \in X} \Pr_{\mathcal{A}} [\mathcal{A}^a \neq 0] \leq \epsilon^2 \quad \text{and} \quad \mathbf{E}_{y=(c,d) \in Y} \Pr_{\mathcal{A}} [\mathcal{A}^c \neq 1] \leq \epsilon^2 \quad (14)$$

uses

$$\geq \left(1 - 2\sqrt{\epsilon(1-\epsilon)}\right) \sqrt{\frac{(m - 2\epsilon\bar{m})(m' - 2\epsilon\bar{m}')}{\ell_{\max}}} \quad \text{queries.} \quad (15)$$

► **Corollary 14.** *Let X and Y be two subsets of $\{0, 1\}^{N+M}$ satisfying the three conditions listed in Theorem 13. Then, any query algorithm $(1 - \delta)$ -distinguishing the uniform distributions over X and Y , must use eq. (15) queries for $\epsilon = 2\delta$.*

The proofs of both statements are presented in the Appendix of the full version [22].

6.2 Setup from QCMA algorithm

In this subsection, we show that if there is a QCMA algorithm for solving the *expander distinguishing problem* then there exists a sunflower \mathfrak{S} (defined below) of YES instances which correspond to the same optimal witness wt^* . If we hardcode wt^* into the QCMA algorithm, we generate a quantum query algorithm that, with no access to a prover, accepts instances corresponding to \mathfrak{S} and rejects all NO instances.

► **Definition 15 (Sunflower).** *A collection of subsets $\mathfrak{S} \subset \binom{V}{\zeta}$ is (μ, ζ, t) -sunflower if there exists a subset $F \subset V$ with $|F| \leq t$ satisfying the following two conditions:*

1. For all $S \in \mathfrak{S}$, $F \subseteq S$.
2. For all $x \in \left(\bigcup_{S \in \mathfrak{S}} S\right) \setminus F$, the $\Pr_{S \in \mathfrak{S}} [x \in S] \leq \left(\frac{\zeta}{N}\right)^{1-\mu}$.

We call the set F the *core* of the sunflower.

YES instances corresponding to subsets

For any graph G and subset S of size ζ , define $G \triangleleft S$ if G has a connected component $C_i \subseteq S$. Let $\mathcal{S}_{\triangleleft}$ be the set of G such that $G \triangleleft S$. For each subset S of size ζ , define B_S to be the restriction of the distribution $P_{M,\ell}$ to graphs in $\mathcal{S}_{\triangleleft}$. The intuition is that the ideal witness will be a good witness for B_S since the connected components of $P_{M,\ell}$ are of a size concentrated around z .

There is a small complication, which we address now, in that the distribution $P_{M,\ell}$ is not a uniform distribution over a set of graphs. To rectify this, we can always assume that the oracle corresponding to a graph G sampled to $P_{M,\ell}$ consists of a queryable component corresponding to the adjacency list of G and a non-queryable component corresponding to the random coins r_G that were flipped in order to generate G according to $P_{M,\ell}$. We will also define B_S as the restriction of the extended oracle. Therefore, both $P_{M,\ell}$ and B_S are uniform distributions over some support.

Lastly, the distributions B_S are not exactly YES distributions since their support is not *entirely* on YES graphs of the expander distinguishing problem. However, similar to Lemma 10, we will show that B_S is almost entirely supported on YES graphs. Therefore, it suffices to use B_S as a *proxy* for YES instances until the very end where we handle this subtlety.

► **Corollary 16.** For every graph $G \in \mathcal{S}_d$ such that G has a connected component C_i with $|C_i| \geq (1 - \gamma)z$,

$$\Pr[\mathcal{A}_{\text{QMA}}^G(|S\rangle) = 1] \geq 1 - 3\sqrt{\gamma}. \quad (16)$$

Proof. Proof is available in the full version [22]. ◀

► **Corollary 17.** Let $M = (1 + \gamma)N$ and $\gamma = N^{-1/10}$ and $\ell = N^{-1/10}$. For every S of size $(1 + \gamma)z$, the distribution B_S is a YES instance and

$$\mathbf{E}_{G \leftarrow B_S} [\Pr[\mathcal{A}_{\text{QMA}}^G(|S\rangle) = 1]] \geq 1 - 3\sqrt{\gamma} - O(N^{-3}). \quad (17)$$

Proof. Proof is available in the full version [22]. ◀

QCMA algorithm implies a quantum low-query algorithm for some sunflower ☼

► **Lemma 18.** For some $\epsilon > 0$, assume there exists a k -query non-deterministic quantum algorithm which accepts a q -length classical witness and accepts every distribution B_S for subset S of size ζ with probability $1 - \epsilon$ and accepts any NO distribution B_{NO} with probability at most ϵ . Then for $\mu > 0$, there exists a $(\mu, \zeta, 2q/(\mu \log \ell))$ -sunflower ☼ and a k -query quantum algorithm that accepts every distribution B_S for $S \in \text{☼}$ and accepts any NO distribution B_{NO} with probability at most ϵ .

Proof. Proof is available in the full version [22]. ◀

6.3 Query lower bound for distinguishing sunflowers and fixed distributions

Let $\textcircled{F} \stackrel{\text{def}}{=} \binom{V}{\zeta} \cap \{S : F \subseteq S\}$. This is the *ideal sunflower* with a core of F . We will show by an adversary bound that the sunflower ☼ and the ideal sunflower \textcircled{F} are indistinguishable by quantum query algorithms with few queries. Consider the distribution $H_{\text{☼}}$ defined by sampling an $S \in \text{☼}$ and then sampling a graph from B_S . Similarly, define the distribution $H_{\text{Ⓢ}}$ but by first sampling an $S \in \text{Ⓢ}$. We want to show that any quantum query algorithm requires exponentially many queries to distinguish $H_{\text{☼}}$ and $H_{\text{Ⓢ}}$. The main result of this subsection is the following lemma.

► **Lemma 19.** For $\delta < 1/4$, any quantum query algorithm $(1 - \delta)$ -distinguishing the distributions $H_{\text{☼}}$ and $H_{\text{Ⓢ}}$ where ☼ is a (μ, ζ, t) -sunflower and F is the corresponding core requires

$$\geq \frac{1}{2} \left(1 - 2\sqrt{2\delta(1 - 2\delta)}\right) (1 - 4\delta) \cdot \sqrt{\left(\frac{N}{\zeta}\right)^{1-\mu}} \text{ queries.} \quad (18)$$

6.3.1 A warmup lemma for distinguishing graphs

The main challenge in proving Lemma 19 is the complicated structure inherent in graphs. However, if we work instead directly with the *sets* S , the problem is much simpler, and was already solved in [13, Lemma 11]. They showed that given membership query access (equivalently, the indicator function for the set), it requires exponentially many quantum queries to distinguish a sample from ☼ from a sample from Ⓢ .

We will work up to the result we wish to prove by gradually adding more structure to the objects being queried until we reach graphs. We will start by working with *permutations* that map the set S to a known set, and show that any algorithm with query access to the permutation and its inverse requires exponentially many queries.

To be precise, let $U = [\zeta]$. Let $\Pi_{\mathfrak{S}}$ be the set of all permutations and inverses (π, π^{-1}) such that $\pi(S) = U$ for some $S \in \mathfrak{S}$. Similarly, define $\Pi_{\mathfrak{F}}$. We shall abuse notation and also use $\Pi_{\mathfrak{S}}$ and $\Pi_{\mathfrak{F}}$ to refer to the uniform distributions over these sets of permutations. We first claim that no quantum query algorithm can distinguish the distributions $\Pi_{\mathfrak{S}}$ and $\Pi_{\mathfrak{F}}$ without an exponential number of queries. Note that the algorithm is allowed to query both the permutation and its inverse⁷.

► **Lemma 20.** *Any quantum query algorithm $(1 - \delta)$ -distinguishing the distributions $\Pi_{\mathfrak{S}}$ and $\Pi_{\mathfrak{F}}$ where \mathfrak{S} is a (μ, ζ, t) -sunflower and F is the corresponding core requires*

$$\geq \left(\frac{1}{2} - 2\sqrt{2\delta(1-2\delta)} \right) (1-4\delta) \cdot \sqrt{\left(\frac{N}{\zeta} \right)^{1-\mu}} \text{ queries.} \quad (19)$$

Proof. Proof is available in the full version [22]. ◀

A short corollary of Lemma 20 is that there is a similar query lower bound for distinguishing distributions over graphs. Let \mathcal{G} be a distribution over graphs with a connected component of $U = [\zeta]$. Let $\mathcal{G}_{\mathfrak{S}}$ be the distribution over graphs formed by sampling a permutation pair (π, π^{-1}) from $\Pi_{\mathfrak{S}}$, a graph G from \mathcal{G} and outputting the graph $\pi^{-1}(G)$. By construction, $\mathcal{G}_{\mathfrak{S}}$ is a distribution over graphs with a connected component of S for $S \in \mathfrak{S}$. Likewise, define the distribution $\mathcal{G}_{\mathfrak{F}}$.

► **Corollary 21.** *For $\delta < 1/4$, any quantum query algorithm $(1 - \delta)$ -distinguishing the distributions $\mathcal{G}_{\mathfrak{S}}$ and $\mathcal{G}_{\mathfrak{F}}$ where \mathfrak{S} is a (μ, ζ, t) -sunflower and F is the corresponding core requires*

$$\geq \frac{1}{2} \left(1 - 2\sqrt{2\delta(1-2\delta)} \right) (1-4\delta) \cdot \sqrt{\left(\frac{N}{\zeta} \right)^{1-\mu}} \text{ queries.} \quad (20)$$

Proof. The intuition is that any algorithm \mathcal{A} for distinguishing $\mathcal{G}_{\mathfrak{S}}$ and $\mathcal{G}_{\mathfrak{F}}$, can be used as a subroutine in a (not necessarily time-efficient) algorithm \mathcal{A}' for distinguishing $\Pi_{\mathfrak{S}}$ and $\Pi_{\mathfrak{F}}$ in twice as many queries. A full proof is available in the full version [22]. ◀

6.3.2 Improving to more general permutations

While Lemma 20 and Corollary 21 are simple enough to prove, they are insufficient at proving indistinguishability for the graph distributions $H_{\mathfrak{S}}$ and $H_{\mathfrak{F}}$ defined at the start of this section. This is because, unlike the distribution $\mathcal{G}_{\mathfrak{S}}$, the distribution $H_{\mathfrak{S}}$ cannot be defined in terms of independently sampling a graph G and a set S . For one, the sizes of the connected components in $H_{\mathfrak{S}}$ do not exactly equal z ; instead, they concentrate tightly around z . It was precisely the independence of the graphs and sets that made Corollary 21 easy to prove.

To fix the argument, we prove the following variations of Lemma 20 and Corollary 21. For a sunflower \mathfrak{S} with core F and any k such that $|F| \leq k \leq \zeta$, let $\Pi_{\mathfrak{S}}^{(k)}$ be the distribution formed by the following procedure:

1. Sample a set S from \mathfrak{S} .
2. Sample uniformly randomly a subset $C \subset S$ of size k .
3. Sample uniformly randomly a permutation $\pi : V \rightarrow V$ such that $\pi(C) = [k]$.
4. Output (π, π^{-1}) .

Define the distribution $\Pi_{\mathfrak{F}}^{(k)}$ similarly where we change the first step to sampling from \mathfrak{F} .

⁷ This can be equivalently modeled by having a separate in-place oracle for the permutation and its inverse, or having a single “standard” oracle for the permutation.

► **Lemma 22.** *Any quantum query algorithm $(1 - \delta)$ -distinguishing the distributions $\Pi_{\mathfrak{F}}^{(k)}$ and $\Pi_{\mathfrak{F}}^{(k)}$ where \mathfrak{F} is a (μ, ζ, t) -sunflower and F is the corresponding core requires*

$$\geq \left(1 - 2\sqrt{2\delta(1 - 2\delta)}\right)(1 - 4\delta) \cdot \sqrt{\left(\frac{N}{\zeta}\right)^{1-\mu}} \text{ queries.} \quad (21)$$

Proof. This proof is equivalent to that of Lemma 20 except we use $U = [k]$. Note, the listed bound has no dependence on k ; this is because $k \leq \zeta$ and we express here the weaker bound with ζ . ◀

Likewise, a short corollary of Lemma 22 is the following. Construct the distribution $\mathcal{G}_{\mathfrak{F}}^{(k)}$ by the following procedure:

1. Sample a graph G from the restriction of the distribution $P_{M,\ell}$ to graphs with a connected component of exactly $[k]$.
2. Sample a permutation (π, π^{-1}) from $\Pi_{\mathfrak{F}}^{(k)}$.
3. Output $(\pi^{-1}(G), r_G)$ where r_G is the random coin flips that would have generated G when sampling according to $P_{M,\ell}$. The oracle will be divided into a queryable component of $(\pi^{-1}(G))$ and a un-queryable component of r_G .

► **Corollary 23.** *For $\delta < 1/4$, any quantum query algorithm $(1 - \delta)$ -distinguishing the distributions $\mathcal{G}_{\mathfrak{F}}^{(k)}$ and $\mathcal{G}_{\mathfrak{F}}^{(k)}$ where \mathfrak{F} is a (μ, ζ, t) -sunflower and F is the corresponding core requires*

$$\geq \frac{1}{2} \left(1 - 2\sqrt{2\delta(1 - 2\delta)}\right)(1 - 4\delta) \cdot \sqrt{\left(\frac{N}{\zeta}\right)^{1-\mu}} \text{ queries.} \quad (22)$$

Proof. The corollary follows from Lemma 22 via a reduction from permutations to graphs exactly as in the proof of Corollary 21 from Lemma 20. ◀

6.3.3 Completing the proof

Proof of Lemma 19. Notice that for any $k \neq k'$, the support of $\mathcal{G}_{\mathfrak{F}}^{(k)}$ is disjoint from the support of $\mathcal{G}_{\mathfrak{F}}^{(k')}$, and likewise for $\mathcal{G}_{\mathfrak{F}}^{(k)}$ and $\mathcal{G}_{\mathfrak{F}}^{(k')}$. Let us again abuse notation and use $\mathcal{G}_{\mathfrak{F}}^{(k)}$ to denote the support of the corresponding distribution. For each k , the lower bound from Corollary 23 is shown via an adversary bound with a relation R_k , and parameters m, m', ℓ_{\max} , and moreover these parameters are the same for all k . Thus, we may construct a relation R between $\bigcup_k \mathcal{G}_{\mathfrak{F}}^{(k)}$ and $\bigcup_k \mathcal{G}_{\mathfrak{F}}^{(k)}$ by simply taking the union $R = \bigcup_k R_k$. This relation maintains the same parameters m, m', ℓ_{\max} due to the disjointness of supports for different k . Lastly, notice that $\bigcup_k \mathcal{G}_{\mathfrak{F}}^{(k)}$ is equal to the support of $H_{\mathfrak{F}}$ as described in the statement of Lemma 19. Likewise, for $H_{\mathfrak{F}}$. Since $H_{\mathfrak{F}}$ and $H_{\mathfrak{F}}$ are uniform distributions over their support, by Corollary 14 using the relation R that we have constructed, the distributions are indistinguishable without the stated number of queries. ◀

6.4 Statistical indistinguishability between random distributions

The final step of this section is to show that no algorithm can distinguish the distributions $H_{\mathfrak{F}}$ and $P_{M,\ell}(F)$ with more than a negligible probability. This will be because these distributions are statistically close and this can be proven by a Chernoff tail bound.

► **Lemma 24.** *The statistical distance between $H_{\mathfrak{F}}$ and $P_{M,\ell}(F)$ is $O(N^{-3})$.*

Proof. Notice that the distribution $H_{\mathbb{F}}$ is equivalent to sampling a graph from $P_{M,\ell}(F)$ conditioned on consisting of ℓ connected components each with size $\in [(1-\gamma)z, (1+\gamma)z]$. By Lemma 10, with all but $O(N^{-3})$ probability, a graph from $P_{M,\ell}(F)$ satisfies this condition. Therefore, the statistical distance between these distributions is bounded by $O(N^{-3})$. ◀

7 Polynomial method lower bound

In this section, we prove that any quantum query algorithm cannot distinguish the graph distributions $P_{M,1}$ and $P_{M,\ell}(F)$. When $F = \emptyset$, this is equivalent to the problem studied by [6] in their quantum query lower bound:

► **Theorem 25** (Restatement of Theorem 2 of [6]). *For any sufficiently small constant $\epsilon_1 > 0$, any deterministic quantum query algorithm \mathcal{A} distinguishing the distributions $P_{M,1}$ and $P_{M,\ell}$ for any $1 < \ell < N^{1/4}$ by probability ϵ_1 . I.e.*

$$\mathbf{E}_{G \leftarrow P_{M,1}} \left[\Pr_{\mathcal{A}} [\mathcal{A}^G = 1] \right] - \mathbf{E}_{G \leftarrow P_{M,\ell}} \left[\Pr_{\mathcal{A}} [\mathcal{A}^G = 1] \right] \geq \epsilon_1 \quad (23)$$

must make at least $\Omega(N^{1/4}/\log N)$ queries. Here the Ω notation hides a dependence on ϵ_1 .

The proof used in that result is very technical and builds on the polynomial method. Fortunately, we can show our query lower bound via a *reduction* to the [6] result. The reduction requires taking a short walk which mixes well by the expander mixing lemma.

► **Lemma 26.** *Suppose there exists some F_0 and a q_1 -query quantum algorithm that ϵ_1 -distinguishes the distributions $P_{M,1} = P_{M,1}(F_0)$ and $P_{M,\ell}(F_0)$ for $\ell > 1$. Then there exists a q_2 -query quantum algorithm that ϵ_2 -distinguishes the distributions $P_{M,1}$ and $P_{M,\ell}$ with $q_2 = q_1 + O(N^{3/100})$ and $\epsilon_2 = \epsilon_1 - O(N^{-9/200})$.*

Intuitively, what this lemma says is that the set of points F_0 (which are in the same connected component) is not a helpful witness. Concretely, such a witness is negligibly more helpful than no witness at all. This is because, in the case of $P_{M,1}$ or $P_{M,\ell}$, the connected components are expanding and therefore the verifier can easily select a random subset of the points from a single connected component without any assistance from the prover. This can be shown via an application of the expander mixing lemma. Therefore, if a query algorithm exists for distinguishing $P_{M,1}$ and $P_{M,\ell}(F)$, it can be used as a subroutine for distinguishing $P_{M,1}$ and $P_{M,\ell}$ without any witness.

Furthermore, due to Ambainis, Childs, and Liu [6], we know Theorem 25 – i.e. that distinguishing the distributions without witnesses has a query lower bound. Therefore, the problem has a query lower bound even when a set of points F from a connected component are provided:

► **Corollary 27.** *For any F_0 with $|F_0| \leq N^{1/100}$, any sufficiently small constant ϵ_1 , and any ℓ with $1 < \ell < N^{1/4}$, any quantum query algorithm to ϵ_1 -distinguish $P_{M,1}$ and $P_{M,\ell}(F_0)$ must make $\Omega(N^{1/4}/\log N)$ queries.*

Proof. Suppose an algorithm making $q = o(N^{1/4}/\log N)$ queries existed. Then by Lemma 26 there exists an algorithm making $q' = q + O(N^{3/100}) = o(N^{1/4}/\log N)$ queries that distinguishes between $P_{M,1}$ and $P_{M,\ell}$ as well. However, this is impossible by Theorem 25. ◀

The remainder of this section is the proof of Lemma 26.

Let \mathcal{A}_0 be the hypothesized algorithm making q_1 queries to ϵ_1 -distinguish $P_{M,1}$ and $P_{M,\ell}(F_0)$. We first claim that for any F with $|F| = |F_0|$, there exists an algorithm \mathcal{A}_1 that, given as classical input a list of all the vertices in F , and as oracle input an oracle G where G is a sample from either $P_{M,1}$ or $P_{M,\ell}(F)$, can ϵ_1 -distinguish between these two cases using q_1 queries to G . The algorithm \mathcal{A}_1 is as follows:

1. Given F , compute a permutation π on V that maps F to F_0 . (This step is not efficient in terms of runtime, but makes no queries to the oracle G .)
2. Run \mathcal{A}_0 with every query to G replaced by a query to $\pi(G)$. Return the answer given by \mathcal{A}_0 .

The correctness of the algorithm follows from the fact that π maps the distribution $P_{M,\ell}(F)$ exactly to $P_{M,\ell}(F_0)$. Therefore, for all F such that $|F| = |F_0|$,

$$\mathbf{E}_{G \leftarrow P_{M,\ell}(F)} \left[\Pr_{\mathcal{A}_1}[\mathcal{A}_1(F, G) = 1] \right] - \mathbf{E}_{G \leftarrow P_{M,1}} \left[\Pr_{\mathcal{A}_1}[\mathcal{A}_1(F, G) = 1] \right] \geq \epsilon_1. \quad (24)$$

As this holds for all such F ,

$$\mathbf{E}_F \mathbf{E}_{G \leftarrow P_{M,\ell}(F)} \left[\Pr_{\mathcal{A}_1}[\mathcal{A}_1(F, G) = 1] \right] - \mathbf{E}_F \mathbf{E}_{G \leftarrow P_{M,1}} \left[\Pr_{\mathcal{A}_1}[\mathcal{A}_1(F, G) = 1] \right] \geq \epsilon_1. \quad (25)$$

Next, we will show that the input of F can be removed from the algorithm: given just access to G , it is possible to compute a suitable F without making too many queries to the oracle. Specifically, we define the algorithm \mathcal{A}_2 to distinguish between $P_{M,1}$ and $P_{M,\ell}$ given only oracle access to G .

1. For a choice of t to be defined later, construct a set F_1 by starting at a random vertex v_0 and taking a $100t \cdot N^{1/100}$ -step random walk along the graph as described in Lemma 3. If $|F_1| \geq |F|$, pick the first $|F|$ points from F_1 as the set F' . If not, output 0 (i.e. abort).
2. Run \mathcal{A}_1 on input F' with oracle access to G .

We will argue that for an appropriately chosen t , this algorithm achieves the success probability and query complexity claimed in the theorem. To do so, we will argue in two stages.

1. First, we argue that the distribution of F' chosen by random walk is very close to F' chosen uniformly at random from subsets of a connected component of G . This analysis uses the expander mixing lemma.
2. Second, we argue that the distribution over pairs (G, F') obtained after the first step of \mathcal{A}_2 is statistically indistinguishable from the distribution over pairs (G, F) sampled by first choosing a uniformly random $F \subseteq V$ and then choosing a random $G \leftarrow P_{M,\ell}(F)$. This will make use of Lemma 11, shown in the appendix of the full version [22] of this paper. By eq. (25), the algorithm \mathcal{A}_1 can ϵ_1 -distinguish inputs distributed in this manner, and thus the second step of \mathcal{A}_2 can ϵ_2 -distinguish inputs of $P_{M,1}$ and $P_{M,\ell}$ for ϵ_2 just slightly smaller than ϵ_1 .

In our analysis, we will denote probabilities over the distribution of (G, F') generated by \mathcal{A}_2 by $\Pr_{\mathcal{A}_2}[\cdot]$ and probabilities over the distribution of (G, F) obtained by first sampling $F \subseteq V$, and then sampling $G \leftarrow P_{M,\ell}(F)$ by $\Pr_{F \text{ then } G}[\cdot]$. The notation $\Pr_{\text{unif}}[\cdot]$ denotes the distribution over F obtained by first picking a uniformly random vertex v in G , and then picking F to be a uniformly random subset of the connected component of G containing v with size $|F_0|$.

7.1 From random walk sampling to uniform sampling

Henceforth, define *expander walk sampling* as the sampling procedure of selecting a uniformly random vertex as the initial vertex v_1 , then subsequently taking t steps of a lazy random walk (as defined in the expander mixing lemma, Lemma 3) to choose v_2 , and so forth. In this case, the graph and the integer t will be clear from context.

We start by showing a sequence of claims that establish that if the expander walk sampling procedure for generating F' starts in a connected component C of G with size $|C| = K$ and expansion α , then the distribution over sets F' generated by the random walk is close to uniformly sampling points from C . Our main result here will be Claim 30.

▷ **Claim 28.** Let $\delta = (1 - \alpha/2)^t$ and let r be a natural number with $rK\delta < 1$. Then for any sequence of r vertices v_1, \dots, v_r , the probability $\Pr_{\text{unif}}[\cdot]$ that this sequence was obtained by iid random sampling and the probability $\Pr_{\text{walk}}[\cdot]$ that it was obtained by expander walk sampling differ by

$$\left| \Pr_{\text{unif}}[v_1, \dots, v_r] - \Pr_{\text{walk}}[v_1, \dots, v_r] \right| \leq \left(\frac{1}{K} \right)^r \cdot \left(rK\delta + (rK\delta)^2 \frac{1}{1 - rK\delta} \right). \quad (26)$$

Proof. Proof is available in the full version [22]. ◁

The following claim will be used to bound the probability that the expander walk sampling procedure aborts, by instead bounding the probability that iid sampling fails to generate enough distinct points.

▷ **Claim 29.** The probability that $T \geq 100|F|$ iid samples from C contain fewer than $|F|$ distinct vertices is at most $\exp(-T/16)$.

Proof. Proof is available in the full version [22]. ◁

We now combine these two claims and apply them to our setting. Define the event $[F' \leftarrow G]$ if F' is the set of vertices selected from the graph G . Define the distribution $\Pr_{\text{unif}}[\cdot]$ corresponding to first choosing a connected component C with probability proportional to $|C|$, taking r uniform iid samples from the connected component C and setting F' to be the first $|F|$ distinct sampled points. Likewise, define the distribution $\Pr_{\mathcal{A}_2}$ corresponding to choosing a random vertex v in G , taking C to be the connected component containing v , taking r samples according to an expander random walk in C initialized at v with t steps between samples, and then setting F' to be the first $|F|$ distinct sampled points. Set K to be the maximum size of a connected component in G and let $\delta = (1 + \alpha/2)^t$. Then we have the following distance bound between the distributions.

▷ **Claim 30.** Suppose r, K, δ are such that $rK\delta \leq 10/11$. For any F' of size $|F'| = |F|$, let $\delta_C(F') = \Pr_{\text{unif}}[F' \leftarrow G] - \Pr_{\mathcal{A}_2}[F' \leftarrow G]$. Then

$$|\delta_C(F')| \leq \left(rK\delta + (rK\delta)^2 \frac{1}{1 - rK\delta} \right) \leq 10rK\delta. \quad (27)$$

Moreover, $\Pr_{\mathcal{A}_2}[\text{abort}] \leq 10rK\delta + \exp(-T/16)$.

Proof. Proof is available in the full version [22]. ◁

7.2 From $\Pr_{\mathcal{A}_2}[\cdot]$ to $\Pr_{F \text{ then } G}[\cdot]$

We will now proceed to the main argument showing that the pairs (G, F) sampled by \mathcal{A}_2 are distributed close to the distribution expected by \mathcal{A}_1 .

G has expanding components with high probability

To start off, first note that by Lemma 10, with probability at least $1 - O(N^{-3})$ a graph drawn from $P_{M,\ell}(F)$, for any F of size $\leq N^{1/100}$, will consist of ℓ connected-components which are α -expanders and have size between $[(1 - \gamma)z, (1 + \gamma)z]$. Since ϵ_1 is a constant, for sufficiently large N , we can restrict to the situation that the graph is of this form and account for this factor in the end. Henceforth set $K_0 = (1 + \gamma)z$; we are guaranteed that every component has size at most K_0 .

Relating the probabilities

In the case that each connected component is an α -expander, observe that the probability of every valid pair (G, F') is approximately a constant p independent of G and F' . Also recall that the event $F' \leftarrow G$ is the event that F' is the set of vertices selected from G . Moreover, recall the distributions \mathcal{D}_1 and \mathcal{D}_2 from Lemma 11, and notice that \mathcal{D}_2 is exactly the distribution $\Pr_{F \text{ then } G}$ defined above. We define

$$\delta_{1,2}(G, F') \stackrel{\text{def}}{=} \Pr_{\mathcal{D}_1}[(G, F')] - \Pr_{\mathcal{D}_2}[(G, F')], \quad (28)$$

$$\delta_G(F') \stackrel{\text{def}}{=} \Pr_{\mathcal{A}_2}[F' \leftarrow G] - \Pr_{\text{unif}}[F' \leftarrow G]. \quad (29)$$

We will now start with the $\Pr_{\mathcal{A}_2}$ distribution and bound its distance from $\Pr_{F \text{ then } G}$.

$$\Pr_{\mathcal{A}_2}[(G, F')] = \Pr_{P_{M,\ell}}[G] \cdot \Pr_{\mathcal{A}_2}[F' \leftarrow G] \quad (30)$$

$$= \Pr_{P_{M,\ell}}[G] \cdot \left(\Pr_{\text{unif}}[F' \leftarrow G] + \delta_G(F') \right) = \Pr_{\mathcal{D}_1}[(G, F')] + \delta_G(F') \cdot \Pr_{P_{M,\ell}}[G] \quad (31)$$

$$= \Pr_{\mathcal{D}_2}[(G, F')] + \delta_{1,2}(G, F') + \delta_G(F') \cdot \Pr_{P_{M,\ell}}[G] \quad (32)$$

$$= \Pr_{F \text{ then } G}[(G, F')] + \delta_{1,2}(G, F') + \delta_G(F') \cdot \Pr_{P_{M,\ell}}[G]. \quad (33)$$

We may now bound the total variational distance between the two sides.

$$\left\| \Pr_{\mathcal{A}_2}[\cdot] - \Pr_{F \text{ then } G}[\cdot] \right\| = \frac{1}{2} \sum_{(G, F')} \left| \Pr_{\mathcal{A}_2}[(G, F')] - \Pr_{F \text{ then } G}[(G, F')] \right| \quad (34)$$

$$\leq \frac{1}{2} \sum_{(G, F')} \left(|\delta_{1,2}(G, F')| + \Pr_{P_{M,\ell}}[G] \cdot |\delta_G(F')| \right) \quad (35)$$

$$\leq \|\mathcal{D}_1 - \mathcal{D}_2\| + \frac{1}{2} \left(\frac{K_0}{|F'|} \right) \cdot \max_{G, F'} |\delta_G(F')| \quad (36)$$

$$\leq 3N^{-9/200} + K_0^{|F'|} \cdot (10rK_0\delta) \quad (37)$$

$$= 3N^{-9/200} + ((1 + \gamma)z)^{|F'|+1} \cdot (10 \cdot (100|F|)) \cdot (1 - \alpha/2)^t \quad (38)$$

$$\leq 3N^{-9/200} + \left(2N^{9/10} \right)^{N^{1/100}+1} \cdot 1000N^{1/100} \cdot (1 - \alpha/2)^t \quad (39)$$

$$= 3N^{-9/200} + 2000 \cdot 2^{N^{0.01}} \cdot N^{0.9N^{0.01}+0.91} \cdot (1 - \alpha/2)^t. \quad (40)$$

A total distance bound of $O(N^{-9/200})$ can be achieved if

$$2^{N^{0.01}} \cdot N^{0.9N^{0.01}+0.91} \cdot (1 - \alpha/2)^t \leq N^{-9/200} \quad (41)$$

$$N^{0.01} + (0.9N^{0.01} + 0.91) \cdot \log N + t \cdot \log(1 - \alpha/2) \leq -\frac{9}{200} \log N \quad (42)$$

$$\left(N^{0.01} \left(\frac{1}{\log N} + 0.9 \right) + 0.955 \right) \frac{\log N}{\log(1/(1-\alpha/2))} \leq t. \quad (43)$$

So setting $t = \Theta(N^{0.02})$ is sufficient.

Given this choice of t , let us now calculate the chance that the sampling of F' aborts. By Claim 30, this is at most $10rK\delta + \exp(-T/16) = O(N^{-9/200}) + \exp(-100N^{0.01}/16) = O(N^{-9/200})$.

Thus, the total error probability of \mathcal{A}_2 equals the error probability of \mathcal{A}_1 up to

$$\underbrace{O(N^{-3})}_{\text{Sample a bad graph}} + \underbrace{O(N^{-9/200})}_{\text{Changing } \mathcal{A}_2 \text{ to } F \text{ then } G} + \underbrace{O(N^{-9/200})}_{\text{Sampling } F' \text{ aborts}}, \quad (44)$$

yielding $\epsilon_2 = \epsilon_1 - O(N^{-9/200})$ as claimed in theorem. And the total query complexity assuming not aborting can be calculated as follows. Recall that t was chosen to be $\Theta(N^{0.02})$. The total number of additional queries over \mathcal{A}_1 is thus the number of steps in the walk which is $100N^{1/100} \cdot t \leq O(N^{0.03})$. Thus, this algorithm has total query complexity $q + O(N^{0.03})$ and distinguishes with probability ϵ_2 as claimed.

8 Wrapping up the proof of Theorem 1

First, we need to note that the distributions B_S and $P_{M,1}$ which we used as proxies for YES and NO instances are not fully supported on YES and NO instance graphs, respectively. However, they are very close. For every $S \subset [N]$ of size ζ , let \tilde{B}_S be the restriction of the distribution B_S (defined in Section 6.2) to graphs with ℓ connected components each consisting of between $(1-\gamma)z$ and $(1+\gamma)z$ vertices. By Corollary 17, the statistical distance between B_S and \tilde{B}_S is $O(N^{-3})$. The YES instances for Theorem 1 are the $\{\tilde{B}_S\}$.

We consider a single NO instance of $\tilde{P}_{M,1}$ where $\tilde{P}_{M,1}$ is the restriction of $P_{M,1}$ to graphs which are α -expanders. The statistical distance between these two distributions is $O(N^{-3})$ by Lemma 10.

Furthermore, we can verify that the supports of $\tilde{P}_{M,1}$ and \tilde{B}_S are far apart in Hamming distance. Consider graphs G_1 and G_ℓ from either support, respectively. Consider a connected component C from G_ℓ . In the graph G_ℓ , all the edges on C stay within G_ℓ , but since G_1 is an α -expander and also a $1/10^2$ -edge expander (see proof of Lemma 10), then in G_1 , a $\geq 1/10^4$ fraction of the edges emanating from C leave C . As this holds for all components C since $|C| \ll N/2$, then the Hamming distance between the adjacency lists of G_1 and G_ℓ is $\Omega(N)$. As this holds for all graphs G_1 and G_ℓ , then the Hamming distance bound between the supports hold.

8.1 QMA algorithm

For completeness, from Corollary 17, we know that the algorithm \mathcal{A}_{QMA} with witness state $|S\rangle$ answers distribution \tilde{B}_S with probability at least $\geq 1 - O(N^{-1/20})$. For soundness, from Lemma 10, we know that $\tilde{P}_{M,1}$ is an $1/(2 \cdot 10^8)$ -expander with probability $\geq 1 - O(N^{-3})$. Therefore, by Lemma 12, the algorithm \mathcal{A}_{QMA} accepts with probability at most

$$\leq 1 - \frac{1}{4} \cdot \frac{1}{(2 \cdot 10^8)} + O(N^{-3}) \leq 1 - \frac{1}{9 \cdot 10^8}. \quad (45)$$

By parallel repetition $9 \cdot 10^6 = O(1)$ times, we yield a quantum algorithm with ≤ 0.01 soundness.

8.2 QCMA algorithm

We argue now that any QCMA algorithm with completeness 0.99 and soundness 0.01 either requires an exponentially long proof or an exponential number of quantum queries. This is done by arguing that any algorithm with a short proof and few queries cannot have such a large completeness and soundness gap. Assume, therefore, that there exists a QCMA algorithm with a

$$q \leq \frac{n \cdot N^{1/100}}{2000}\text{-bit proof and } f = O(N^{1/50}) \text{ quantum queries} \quad (46)$$

and a completeness and soundness gap of ≥ 0.98 . By Lemma 18, there exists a

$$\left(\frac{1}{100}, \zeta, \frac{2000q}{n}\right)\text{-sunflower } \clubsuit \quad (47)$$

with core F and a f -query deterministic quantum algorithm \mathcal{A} that accepts each distribution \tilde{B}_S for $S \in \clubsuit$ with probability ≥ 0.99 and accepts $\tilde{P}_{M,1}$ with at most ≤ 0.01 probability. It also accepts that accepts each distribution B_S for $S \in \clubsuit$ with probability $\geq 0.99 - O(N^{-3})$ and accepts $P_{M,1}$ with at most $\leq 0.01 + O(N^{-3})$ probability. Then with the assumed number of queries, we can apply Lemma 19 with $\delta = 1/10$ to argue that \mathcal{A} must accept the distribution H_{Ω_F} with probability $\geq 0.09 - O(N^{-3})$. Next, by Lemma 24, \mathcal{A} must accept the distribution $P_{M,\ell}(F)$ with probability $\geq 0.09 - 2 \cdot O(N^{-3}) \geq 0.08$. We conclude by applying Corollary 27. Therefore, \mathcal{A} must accept the distribution $P_{M,1}$ with probability > 0.02 , a contradiction.

9 Concluding remarks

9.1 Relation to the Fefferman and Kimmel [13] construction

One can think of the result stated in this work as applying the QCMA lower bounding techniques developed by Fefferman and Kimmel [13] to the expander distinguishing problem originally studied by Ambainis, Childs, and Liu [6].

At a high level, in the *in-place permutation oracle* QMA and QCMA separation of [13], the goal was to distinguish between permutations $\pi : [N] \rightarrow [N]$ such that $\pi^{-1}([\sqrt{N}])$ is mostly (2/3) supported on odd numbers from permutations mostly supported on even numbers. The original idea in Fefferman and Kimmel was that if the oracle π was provided as a classical oracle (an Nn -bit list $[\pi(1), \pi(2), \dots, \pi(N)]$) then the subset state $|\xi_{\text{ideal}}\rangle = |\pi^{-1}([\sqrt{N}])\rangle$ would be a good quantum witness. By measuring the last qubit of a witness $|\xi\rangle$, the verifier can decide if the set $\pi^{-1}([\sqrt{N}])$ is supported mostly on either odd numbers or even numbers. What remains to verify is that the witness $|\xi\rangle$ provided is indeed $|\xi_{\text{ideal}}\rangle$. The hope would be to use the oracle for π to verify the statement as the state $|\xi_{\text{ideal}}\rangle$ can be easily verified by measuring in the Hadamard basis.

However, due to the index-erasure problem, a classical oracle for verifying that $|\xi\rangle = |\xi_{\text{ideal}}\rangle$ would need to allow implementation of both π and π^{-1} . However, if the oracle π^{-1} is provided, then there is a BQP algorithm for this problem. Simply, pick a random $j \in [\sqrt{N}]$ and then check if $\pi^{-1}(j)$ is odd or even. The solution in [13] was to define the oracle instead as an “in-place oracle” for π , meaning a unitary defined as $\sum_j |\pi(j)\rangle\langle j|$. Then the verifier can verify that $|\xi\rangle = |\xi_{\text{ideal}}\rangle$ and yet the BQP algorithm no longer holds.

Fefferman and Kimmel had to make one more modification to prove a QMA and QCMA oracle separation: they considered distributions over in-place oracles which mapped to the same ideal quantum witness $|\xi_{\text{ideal}}\rangle$. This was because it seems to be beyond current techniques

to prove classical lower bounds without forcing a large structured set of permutations to all share the same witness – otherwise, for all we know, there might be a mathematical fact about permutations which yields a short classical certificate for any individual permutation. So the oracle is defined as a distribution over unitaries – i.e. a completely positive trace preserving (CPTP) map.

Notice that this work takes much inspiration from [13]; the quantum witnesses for both our work and [13] are subset states and we also consider distributions over oracles with the same (or similar) ideal quantum witness. This is because we are unsure how to prove that there is no property of a specific regular graph which yields a short classical witness. We elaborate on why such an impossibility result is hard to prove in the next subsection. What our result principally improves on is that the underlying oracle can be a classical string instead of a unitary.

9.2 Difficulties in proving stronger statements

Recall that our QMA upper bound does not require the setup of distributions over oracles – it was only included to prove the QCMA lower bound. How much harder is it (or is it even possible) to prove a QCMA lower bound without considering distributions?

As pointed out to us by William Kretschmer [20], if one considers *average-case* algorithms instead of *worst-case* algorithms, then this problem is $\in \text{RNP}^G$, the average-case analog of NP^G . This is because the average-case version of the expander distinguishing problem is to distinguish the distributions $P_{M,\ell}$ and $P_{M,1}$. And there is a simple randomized algorithm for this problem with a classical witness. Let us recall that for a d -regular graph, the expected number of triangles in a connected component is $\Theta(d^3)$ independent of the number of vertices in the component. A similar analysis can be done for $P_{M,\ell}$ and $P_{M,1}$, to show that a random graph from $P_{M,\ell}$ has $\Theta(\ell d^3)$ triangles whereas $P_{M,1}$ has $\Theta(d^3)$ triangles. Therefore, a *classical* witness for the statement that the graph (with high probability) is drawn from $P_{M,\ell}$ (instead of $P_{M,1}$) is a list of $100 \cdot \Theta(d^3)$ triangles from the graph. This witness is easily verifiable and correctly distinguishes with high probability.

Notice that this RNP^G algorithm does not solve the expander distinguishing problem in the worst-case; since graphs exist in both distributions which are triangle-free (with constant probability). Furthermore, it cannot distinguish the distributions considered in Theorem 1 because the proof relies on finding triangles which is property of the graph not deducible from only knowing the connected components.

But it does highlight a principal roadblock in extending Theorem 1 to distinguishing oracles that are not distributions. It is entirely possible that there exists a property of graphs revealed by looking at the edges that distinguishes graphs with many connected components from graphs with a single expanding connected component. To the best of our knowledge, we do not know of any such property but proving that none exist is beyond the techniques shown here.

Lastly, if we consider the expander distinguishing problem when in the YES case we are promised that every connected component has size at most $0.99N$, then this problem is in coAM^G . When the graph is a NO instance, the verifier can select two random points and the prover can always find a path of length $O(\log N) = O(n)$ between the two. However, when the graph is disconnected and no component is too big, with probability $\geq 1/50$, no path exists.

Therefore, our constructed oracle very finely separates the classes QMA and QCMA in the sense that small perturbations of the problem might be very easy.

References

- 1 Scott Aaronson. On perfect completeness for QMA. *Quantum Information & Computation*, 9(1):81–89, 2009. [arXiv:0806.0450](#).
- 2 Scott Aaronson. Open problems related to quantum query complexity. *ACM Transactions on Quantum Computing*, 2(4), December 2021. [doi:10.1145/3488559](#).
- 3 Scott Aaronson and Greg Kuperberg. Quantum versus classical proofs and advice. In *Twenty-Second Annual IEEE Conference on Computational Complexity (CCC'07)*, pages 115–128, 2007. [doi:10.1109/CCC.2007.27](#).
- 4 Dorit Aharonov and Tomer Naveh. Quantum NP – A survey, 2002. [arXiv:arXiv:quant-ph/0210077](#).
- 5 Andris Ambainis. Quantum lower bounds by quantum arguments. *J. Comput. Syst. Sci.*, 64(4):750–767, June 2002. [doi:10.1006/jcss.2002.1826](#).
- 6 Andris Ambainis, Andrew M. Childs, and Yi-Kai Liu. Quantum property testing for bounded-degree graphs. In Leslie Ann Goldberg, Klaus Jansen, R. Ravi, and José D. P. Rolim, editors, *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques*, pages 365–376, Berlin, Heidelberg, 2011. Springer Berlin Heidelberg.
- 7 Anurag Anshu, Nikolas Breuckmann, and Chinmay Nirkhe. NLTS hamiltonians from good quantum codes, 2022. [arXiv:2206.13228](#).
- 8 Atul Singh Arora, Alexandru Gheorghiu, and Uttam Singh. Oracle separations of hybrid quantum-classical circuits. *CoRR*, 2022. [arXiv:2201.01904](#).
- 9 Charles H. Bennett, Ethan Bernstein, Gilles Brassard, and Umesh V. Vazirani. Strengths and weaknesses of quantum computing. *SIAM J. Comput.*, 26(5):1510–1523, 1997. [doi:10.1137/S0097539796300933](#).
- 10 Adam D. Bookatz. QMA-complete problems. *CoRR*, 2012. [doi:10.48550/arXiv.1212.6312](#).
- 11 Sergey Bravyi, David P. Divincenzo, Roberto Oliveira, and Barbara M. Terhal. The complexity of stoquastic local hamiltonian problems. *Quantum Info. Comput.*, 8(5):361–385, May 2008.
- 12 Sergey Bravyi and Barbara Terhal. Complexity of stoquastic frustration-free hamiltonians. *SIAM J. Comput.*, 39(4):1462–1485, November 2009.
- 13 Bill Fefferman and Shelby Kimmel. Quantum vs. classical proofs and subset verification. In Igor Potapov, Paul G. Spirakis, and James Worrell, editors, *43rd International Symposium on Mathematical Foundations of Computer Science, MFCS 2018, August 27-31, 2018, Liverpool, UK*, volume 117 of *LIPICs*, pages 22:1–22:23. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2018. [doi:10.4230/LIPICs.MFCS.2018.22](#).
- 14 Honghao Fu. Personal Communication, October 2022.
- 15 Sandy Irani, Anand Natarajan, Chinmay Nirkhe, Sujit Rao, and Henry Yuen. Quantum search-to-decision reductions and the state synthesis problem, 2021. [arXiv:2111.02999](#).
- 16 Stephen P Jordan, David Gosset, and Peter J Love. Quantum-merlin-arthur-complete problems for stoquastic hamiltonians and markov matrices. *Physical Review. A*, 81(3), March 2010. [doi:10.1103/PHYSREVA.81.032331](#).
- 17 A.Yu. Kitaev. Fault-tolerant quantum computation by anyons. *Annals of Physics*, 303(1):2–30, 2003. [doi:10.1016/S0003-4916\(02\)00018-0](#).
- 18 Hartmut Klauck and Supartha Podder. Two results about quantum messages. In *International Symposium on Mathematical Foundations of Computer Science*, pages 445–456. Springer, 2014.
- 19 Dexter Kozen. Lower bounds for natural proof systems. In *Proceedings of the 18th Annual Symposium on Foundations of Computer Science, SFCS '77*, pages 254–266, USA, 1977. IEEE Computer Society. [doi:10.1109/SFCS.1977.16](#).
- 20 William Kretschmer. Personal Communication, July 2022.
- 21 Andrew Lutomirski. Component mixers and a hardness result for counterfeiting quantum money, 2011. [arXiv:1107.0321](#).
- 22 Anand Natarajan and Chinmay Nirkhe. A distribution testing oracle separation between QMA and QCMA. *CoRR*, 2023. [arXiv:2210.15380](#).

- 23 Chinmay Nirkhe. *Lower bounds on the complexity of quantum proofs*. PhD thesis, EECS Department, University of California, Berkeley, November 2022. URL: <http://www2.eecs.berkeley.edu/Pubs/TechRpts/2022/EECS-2022-236.html>.
- 24 Chinmay Nirkhe. NLTS Hamiltonians from codes, 2022. Simons Institute for the Theory of Computing Quantum Colloquium. Panel Umesh Vazirani, Dorit Aharonov, Matthew Hastings, Anand Natarajan, and Chinmay Nirkhe. URL: <https://simons.berkeley.edu/events/quantum-colloquium-nlts-hamiltonians-codes>.