

# Unitary Property Testing Lower Bounds by Polynomials

Adrian She ✉

University of Toronto, Canada

Henry Yuen ✉

Columbia University, New York, NY, USA

---

## Abstract

We study *unitary property testing*, where a quantum algorithm is given query access to a black-box unitary and has to decide whether it satisfies some property. In addition to containing the standard quantum query complexity model (where the unitary encodes a binary string) as a special case, this model contains “inherently quantum” problems that have no classical analogue. Characterizing the query complexity of these problems requires new algorithmic techniques and lower bound methods.

Our main contribution is a generalized polynomial method for unitary property testing problems. By leveraging connections with invariant theory, we apply this method to obtain lower bounds on problems such as determining recurrence times of unitaries, approximating the dimension of a marked subspace, and approximating the entanglement entropy of a marked state. We also present a unitary property testing-based approach towards an oracle separation between QMA and QMA(2), a long standing question in quantum complexity theory.

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

**Keywords and phrases** Quantum query complexity, polynomial method, unitary property testing, quantum proofs, invariant theory, quantum recurrence time, entanglement entropy, BQP, QMA, QMA(2)

**Digital Object Identifier** 10.4230/LIPIcs.ITCS.2023.96

**Related Version** *Full Version:* <https://arxiv.org/abs/2210.05885>

**Funding** *Adrian She:* NSERC Canada Graduate Scholarship.

*Henry Yuen:* AFOSR award FA9550-21-1-0040, NSF CAREER award CCF-2144219, and the Sloan Foundation.

**Acknowledgements** We thank Joshua Grochow for helpful conversations about invariant theory. We thank Shivam Nadimpalli for suggesting the problem about testing  $k$ -locality of a Hamiltonian. We thank Kunal Marwaha and Gregory Rosenthal for helpful discussions. We thank the reviewers for their feedback.

## 1 Introduction

The query model of quantum algorithms plays a central role in the theory of quantum computing. In this model, the algorithm queries (in superposition) bits of an unknown input string  $X$ , and after some number of queries decides whether  $X$  satisfies a property  $\mathcal{P}$  or not. We now have an extensive understanding of the query complexity of many problems; we refer the reader to Ambainis’s survey [9] for an extensive list of examples.

Although this query model involves quantum algorithms, the task being solved is *classical property testing*, that is, deciding properties of classical strings. This has been very useful for comparing the performance of classical versus quantum algorithms for the same task. In contrast, *quantum property testing* – deciding properties of quantum objects such as states and unitaries – has been studied much less but has been receiving more attention in recent years [35].



© Adrian She and Henry Yuen;

licensed under Creative Commons License CC-BY 4.0

14th Innovations in Theoretical Computer Science Conference (ITCS 2023).

Editor: Yael Tauman Kalai; Article No. 96; pp. 96:1–96:17

Leibniz International Proceedings in Informatics



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

In this paper we focus on *unitary property testing*, where the goal is to decide whether a unitary  $U$  satisfies a property  $\mathcal{P}$  by making as few queries to  $U$  as possible. This systematic study of this topic was initiated by Wang [46], and various aspects have been studied further in [35, 22, 7]. We continue explorations of this topic by developing a new lower bound technique, demonstrating its utility with several unitary property testing problems, and exploring intriguing connections between unitary property testing, invariant theory, and the complexity class QMA(2). Before presenting our findings in detail, we first explain the unitary property testing model.

## 1.1 Unitary Property Testing

The model of unitary property testing we consider is formally defined as follows. Fix a dimension  $d$  and let  $\mathcal{P}_{yes}, \mathcal{P}_{no}$  denote disjoint subsets (called *yes* and *no* instances respectively) of  $d$ -dimensional unitary operators. A *tester* for deciding the problem  $\mathcal{P} = (\mathcal{P}_{yes}, \mathcal{P}_{no})$  is a quantum algorithm that, given query access to a unitary  $U \in \mathcal{P}_{yes} \cup \mathcal{P}_{no}$  (called the *problem instance*), accepts with high probability if  $U \in \mathcal{P}_{yes}$  and otherwise accepts with low probability.<sup>1</sup>

This model includes the standard query model as a special case: a quantum query to a classical string  $X \in \{0, 1\}^d$  is defined to be a query to the unitary  $U$  that maps  $|i\rangle$  to  $(-1)^{X_i} |i\rangle$  for all  $i \in [d]$ . In other words, the unitary is self-adjoint and diagonal in the standard basis.

We can go beyond self-adjoint, diagonal unitaries and study quantum analogues of classical property testing problems, such as:

- Testing quantum juntas: if  $U$  is an  $n$ -qubit unitary, determining whether there is a  $k$ -sized subset  $S$  of qubits outside of which  $U$  acts as the identity. This is analogous to determining whether the input  $X$ , interpreted as a function on  $\{0, 1\}^n$ , only depends on a  $k$ -subset of coordinates. This was studied in [46, 22].
- Approximate dimension: promised that  $U$  applies a phase to all states  $|\psi\rangle$  in a subspace  $S$  of dimension either at  $w$  or  $2w$ , determine the dimension of the subspace. This is analogous to the classical problem of approximating the Hamming weight of an input  $X$ . This was studied in [4].

We can also study property testing problems that have no classical analogue at all, such as:

- Unitary recurrence times: Determining whether  $U^t = I$  or  $\|U^t - I\| \geq \epsilon$  (promised that one is the case) where  $t$  is a fixed integer.
- Hamiltonian properties: Promised that  $U = e^{-iH}$  for some Hamiltonian  $H$  with bounded spectral norm, determine properties of  $H$ , such as whether it is a sum of  $k$ -local terms, or the ground space is topologically ordered.
- Unitary subgroup testing: decide whether  $U$  belongs to some fixed subgroup of the unitary group (such as the Clifford subgroup). This was studied in [16].
- Entanglement entropy problem: Given access to a unitary  $U = I - 2|\psi\rangle\langle\psi|$  for some state  $|\psi\rangle \in \mathbb{C}^d \otimes \mathbb{C}^d$ , decide if the entanglement entropy of the state  $|\psi\rangle$  is low or high, promised one is the case.

These examples illustrate the rich variety of unitary property testing problems: some are motivated by well-studied classical problems in computer science (such as junta testing and approximate counting), whereas others are inspired by questions in quantum physics (e.g., identifying quantum chaos, topological order, or entanglement).

---

<sup>1</sup> We note that the concept of property testing discussed in this paper is more general than typically presented in the literature (see, e.g., [46, 24]), where *no* instances are defined to be  $\epsilon$ -far from the set of *yes* instances for some distance measure. In this paper we allow for other ways of defining *no* instances (as long as they are disjoint from *yes* instances), which may be more natural in many contexts.

## 1.2 A Generalized Polynomial Method and Its Applications

Our main contribution is a lower bound technique for unitary property testing that generalizes the well-known polynomial method in complexity theory. First introduced by Nisan and Szegedy as [36] in classical complexity theory and then adapted for quantum algorithms by Beals, et al. [10], the polynomial method is a powerful technique to lower bound the quantum query complexity of a variety of problems (see, e.g., [6, 18], and the references therein).

The polynomial method is based on the fact that a quantum algorithm making  $T$  queries to a boolean input  $X = (x_1, \dots, x_n)$  yields a real polynomial  $p : \mathbb{R}^n \rightarrow \mathbb{R}$  of degree at most  $2T$  such that  $p(x_1, \dots, x_n)$  is equal to the acceptance probability of the algorithm on input  $X$ . If the algorithm distinguishes between *yes* and *no* instances with some bias, so does the polynomial  $p$ . Thus, lower bounds on the degree of any such distinguishing polynomial directly translates into a lower bound on the quantum query complexity for the same task.

We generalize this to arbitrary unitary properties.

► **Proposition 1.1** (Generalized polynomial method). *The acceptance probability of a quantum algorithm making  $T$  queries to a  $d \times d$  unitary  $U$  and its inverse  $U^*$  can be computed by a degree at most  $2T$  self-adjoint<sup>2</sup> polynomial  $p : \mathbb{C}^{2(d \times d)} \rightarrow \mathbb{C}$  evaluated at the matrix entries of  $U$  and  $U^*$ . Thus, degree lower bounds on such polynomials yields a query lower bound on the algorithm.*

**Proof.** Refer to Section 3.1 of the full version [41]. ◀

Furthermore, we say that a unitary property  $\mathcal{P} = (\mathcal{P}_{yes}, \mathcal{P}_{no})$  is closed under inversion if  $U \in \mathcal{P}_{yes}$  iff  $U^* \in \mathcal{P}_{yes}$ , and  $U \in \mathcal{P}_{no}$  iff  $U^* \in \mathcal{P}_{no}$ . All properties we will study in this paper will be closed under inversion, and hence the polynomial  $p$  satisfies a symmetry under this condition.

► **Proposition 1.2.** *Let  $\mathcal{P}$  be an property closed under inversion and suppose there is a  $T$ -query quantum algorithm for testing property  $\mathcal{P}$ . Let  $p$  be the polynomial from Proposition 1.1 that computes the acceptance property of the algorithm. Then, we may assume that  $p(U, U^*) = p(U^*, U)$ .*

**Proof.** Refer to Section 3.1 of the full version [41]. ◀

Hence, while establishing the existence of  $p$  is straightforward, proving lower bounds on its degree is another matter. The standard approach in quantum query complexity is to *symmetrize*  $p$  to obtain a related polynomial  $q$  whose degree is not too much larger than  $p$ , and acts on a much smaller number of variables (ideally a single variable). The choice of symmetrization method depends on the problem being analyzed. For example, for (classical) properties that only depend on the Hamming weight of the boolean string (these are called *symmetric* properties in the literature), the polynomial  $p$  is averaged over all binary strings with Hamming weight  $k$  in order to obtain a univariate polynomial  $q(k)$  (this is known as *Minsky-Papert symmetrization* [34]). Lower bounds on the degree of  $q$  can be then obtained by using Markov-Bernstein type inequalities from approximation theory.

---

<sup>2</sup> A self-adjoint polynomial is unchanged after complex conjugating every variable and every coefficient.

### 1.2.1 Lower Bounds for Unitarily Invariant Properties

To make Proposition 1.1 useful, we develop symmetrization techniques for unitary properties that are invariant under certain symmetries. We first study *unitarily invariant* properties<sup>3</sup>: these are properties  $\mathcal{P} = (\mathcal{P}_{yes}, \mathcal{P}_{no})$  such that conjugating an instance in  $\mathcal{P}_{yes} \cup \mathcal{P}_{no}$  by any  $g$  in the unitary group  $U(d)$  does not change whether it is a *yes* or *no* instance (in other words  $g\mathcal{P}_{yes}g^{-1} \subseteq \mathcal{P}_{yes}$  and  $g\mathcal{P}_{no}g^{-1} \subseteq \mathcal{P}_{no}$  for all  $g \in U(d)$ ). An example of such a property includes deciding whether a unitary  $U$  is a reflection about a subspace of  $\mathbb{C}^d$  of dimension at most  $w$  (the *no* instances) or dimension at least  $2w$  (the *yes* instances).

It is easy to see that whether a unitary  $U$  is a *yes* or *no* instance of a unitarily invariant property  $\mathcal{P}$  only depends on the multiset of eigenvalues of  $U$ . In fact, we can say something stronger; the following establishes a symmetrization method for polynomials that decide unitarily invariant properties.

► **Theorem 1.3** (Symmetrization for unitarily invariant properties). *Let  $\mathcal{P} = (\mathcal{P}_{yes}, \mathcal{P}_{no})$  denote a  $d$ -dimensional unitarily invariant property. Suppose there is a  $T$ -query quantum algorithm that accepts *yes* instances with probability at least  $a$  and no instances with probability at most  $b$ . Then there exists a degree at most  $2T$  symmetric<sup>4</sup> self-adjoint polynomial  $q(z_1, \dots, z_d, z_1^*, \dots, z_d^*)$  satisfying*

- *If  $U \in \mathcal{P}_{yes}$  then  $q(z_1, \dots, z_d, z_1^*, \dots, z_d^*) \geq a$*
- *If  $U \in \mathcal{P}_{no}$  then  $q(z_1, \dots, z_d, z_1^*, \dots, z_d^*) \leq b$*

*where  $(z_1, \dots, z_d)$  and  $(z_1^*, \dots, z_d^*)$  are the eigenvalues of  $U$  and their complex conjugates, respectively.*

**Proof.** Refer to Section 3.1 of the full version [41]. ◀

The symmetrization of Theorem 1.3 may at first appear quite modest. The symmetrized polynomial  $q$  still acts on  $2d$  variables, which is fewer than the  $2d^2$  variables acted on by the original polynomial  $p$  from Proposition 1.1, but is a far cry from a univariate polynomial which approximation theory is best suited to handle. We have made some progress, however: as mentioned, the property  $\mathcal{P}$  only depends on the eigenvalue multiset of  $U$ , and the polynomial  $q$  is directly a function of the eigenvalues (whereas the original polynomial  $p$  is a function of the matrix entries of  $U$ , which are not obviously related to the property in a low-degree fashion). Furthermore, the polynomial  $q$  can be symmetrized further to obtain a univariate polynomial  $r$ , which we analyze using approximation theory. We illustrate this with two applications of Theorem 1.3.

#### Unitarily Invariant Subspace Properties

As a warmup, consider *subspace properties*, which consist of reflections about a subspace, i.e.,  $U = I - 2\Pi$  where  $\Pi$  is the projector onto some subspace  $S \subseteq \mathbb{C}^d$ . We say that  $U$  *encodes* the subspace  $S$ . An example of a unitarily invariant subspace property is the *Approximate Dimension* problem, which we parametrize by an integer  $w \in \{1, 2, \dots, d\}$ . The *yes* instances consist of (unitaries encoding) subspaces of dimension at least  $2w$ , and the *no* instances consist of subspaces of dimension at most  $w$ . This is a quantum generalization of the *Approximate Counting* problem, which is to determine whether the Hamming weight of an input string is at least  $2w$  or at most  $w$ .

<sup>3</sup> We acknowledge that the name “unitarily invariant unitary property” may seem redundant! The first “unitarily” refers to the symmetry; the second “unitary” refers to the type of property we are studying.

<sup>4</sup> Here, symmetric means that for all permutations  $\pi : [d] \rightarrow [d]$ , permuting the variables  $z_i \rightarrow z_{\pi(i)}$  and  $z_i^* \rightarrow z_{\pi(i)}^*$  leaves the polynomial  $q$  unchanged.

Since the eigenvalues of subspace unitaries are either 1 or  $-1$ , by Theorem 1.3 unitarily invariant subspace properties yield polynomials that compute acceptance probabilities on (a subset of)  $\{1, -1\}^d$ . Note that, after mapping  $\{1, -1\}$  to  $\{0, 1\}$ , these are the same kind of polynomials that arise when analyzing classical properties! In fact, there is a one-to-one correspondence between symmetric classical properties  $\mathcal{S}$  (properties that only depend on the Hamming weight of the input) and unitarily invariant subspace properties  $\mathcal{P}$ .

This implies that the polynomial  $q$  given by Theorem 1.3 (associated to a unitarily invariant subspace property  $\mathcal{P}$ ), also distinguishes between the *yes* and *no* instances of the associated classical symmetric property  $\mathcal{S}$ . This yields a lower bound method for the query complexity of  $\mathcal{P}$ :

► **Proposition 1.4.** *Let  $\mathcal{P}$  be a unitarily invariant subspace property and let  $\mathcal{S}$  be the associated symmetric classical property. The query complexity of distinguishing between *yes* and *no* instances of  $\mathcal{P}$  is at least the minimum degree of any polynomial that distinguishes between the *yes* and *no* instances of  $\mathcal{S}$ .*

**Proof.** Refer to Section 4.1 of the full version [41]. ◀

Therefore, degree lower bounds on polynomials that decide a classical symmetric property  $\mathcal{S}$ , automatically yield query complexity lower bounds for the quantum property  $\mathcal{P}$ . Approximate degree lower bounds on symmetric boolean functions are well-studied in complexity theory [37, 25]; these can be automatically “lifted” to the unitary property testing setting.

We note that there is another way of seeing this reduction: any  $T$ -query tester for  $\mathcal{P}$  is automatically a  $T$ -query tester for  $\mathcal{S}$ ; thus lower bounds on  $\mathcal{S}$  imply lower bounds on  $\mathcal{P}$ . Here our motivation is to present Proposition 1.4 as a simple application of Theorem 1.3.

For example, the Approximate Dimension problem contains the Approximate Counting problem as a special case. Any polynomial that decides with bounded error the Approximate Counting problem must have degree at least  $\Omega(\sqrt{d/w})$ , which implies the same lower bound for the query complexity of the Approximate Dimension problem.

## Recurrence Time of Unitaries

Not all unitarily invariant properties reduce to classical lower bounds. For instance, we analyze a problem related to the recurrence times of unitaries.

In general, the recurrence time of a dynamical system is the time that the system takes to return to a state that is close to its initial state (if it exists). The recurrence statistics of dynamical systems have been extensively studied in the physics literature, where they have been used as indicators of chaotic behaviour within a dynamical system [40]. As the time evolution of a quantum system is governed by a unitary operator, the recurrence times of unitary matrices is of particular interest. The Poincaré recurrence theorem guarantees that the recurrence time exists for certain quantum mechanical systems [15]. For example, the expected recurrence times of a Haar-random unitary were studied in [32]. We now define these concepts more formally.

► **Definition 1.5** (Recurrence Time Problem). *The  $(t, \epsilon)$ -Recurrence Time problem is to decide, given oracle access to a unitary  $U$ , whether  $U^t = I$  (*yes case*) or  $\|U^t - I\| \geq \epsilon$  in the spectral norm (*no case*), promised that one is the case.*

Note that the instances of this problem are generally not self-adjoint; their eigenvalues can be any complex number on the unit circle. There is no obvious classical analogue of the unitary Recurrence Time problem, and thus it does not seem to naturally reduce to a classical lower bound. We instead employ Theorem 1.3 to prove the following lower bound on the Recurrence Time problem:

► **Theorem 1.6.** *Let  $\epsilon \leq \frac{1}{2\pi}$ . Any quantum query algorithm solving the  $(t, \epsilon)$ -Recurrence Time problem for  $d$ -dimensional unitaries with error  $\epsilon$  must use  $\Omega(\max(\frac{t}{\epsilon}, \sqrt{d}))$  queries.*

**Proof.** Refer to Section 4.2 of the full version [41]. ◀

We prove the lower bound by observing that the Recurrence Time problem is testing a unitarily invariant property, and hence Theorem 1.3 applies to give a polynomial  $q$  representing the acceptance probability of any algorithm solving the problem in terms of the eigenvalues of the input unitary  $U$ . We then symmetrize the polynomial  $q$  by constructing a distribution  $D(p, z)$  on unitaries with exactly two eigenvalues  $\{1, z = e^{i\theta}\}$  such that the expected acceptance probability  $r(p, z)$  of the algorithm over  $D(p, z)$  remains a polynomial in  $p$  and  $z$ . Afterwards, Markov-Bernstein type inequalities are used to prove a lower bound on the degree of  $r$ .

We also establish the following upper bound:

► **Theorem 1.7.** *The  $(t, \epsilon)$ -Recurrence Time problem can be solved using  $O(t\sqrt{d}/\epsilon)$  queries.*

**Proof.** Refer to Section 4.2 of the full version [41]. ◀

It is an interesting question to determine whether the upper bound or lower bound (or neither) is tight.

### 1.2.2 Beyond Unitarily Invariant Properties

For unitarily invariant properties, there was a natural candidate for how to symmetrize the polynomials  $p$  we get from Proposition 1.1 by using viewing  $p$  as a polynomial in the eigenvalues of the matrix  $U$  rather than the matrix entries of  $U$ . However, a symmetrization technique for other properties is less unclear. In this direction, we develop symmetrization techniques based on invariant theory.

Invariant theory studies the action of a group  $G$  on a polynomial ring  $\mathbb{C}[x_1, \dots, x_n]$ . We denote the action of  $g \in G$  on  $f \in \mathbb{C}[x_1, \dots, x_n]$  by  $g \cdot f$ . The ring of invariant polynomials  $\mathbb{C}[x_1, \dots, x_n]^G$  is then the subring of  $\mathbb{C}[x_1, \dots, x_n]$  consisting of polynomials satisfying  $g \cdot f = f$  for all  $g \in G$ , that is  $f \in \mathbb{C}[x_1, \dots, x_n]^G$  is left unchanged by the action of  $g$  for all group elements.

There are many natural questions about the invariant ring  $\mathbb{C}[x_1, \dots, x_n]^G$  one can ask, such as construction of a generating set for the invariant ring. For example, one classical example is the action of a permutation  $\sigma \in S_n$  acting on a polynomial  $p(x_1, \dots, x_n)$  by permuting the variables by  $\sigma \cdot p = p(x_{\sigma(1)}, \dots, x_{\sigma(n)})$ . The invariant ring is known as the ring of symmetric polynomials, for which there are many well-known generating sets. One example of a generating set is the power sum symmetric polynomials given by  $p_i = \sum_{j=1}^n x_j^i$ , which generate the symmetric polynomial ring as an algebra. In other words, for any symmetric polynomial  $f$ , there exists a polynomial  $g$  for which  $f = g(p_1, \dots, p_n)$ . There are similar characterizations of the invariant ring for numerous other group actions.

To connect invariant theory with our Proposition 1.1, we prove the following result for testing  $G$ -invariant unitary properties. Since we are studying properties of general unitaries, not just boolean strings, we consider symmetries coming from subgroups of the unitary group  $U(d)$ . Let  $G \subseteq U(d)$  be a compact subgroup equipped with a Haar measure  $\mu$  (i.e., a measure over  $G$  that is invariant under left-multiplication by elements of  $G$ ).

► **Definition 1.8** ( $G$ -invariant property). *Let  $G \subseteq U(d)$  be a compact group. A  $d$ -dimensional unitary property  $\mathcal{P} = (\mathcal{P}_{yes}, \mathcal{P}_{no})$  is  $G$ -invariant if for every  $g \in G$  we have  $g\mathcal{P}_{yes}g^{-1} \subseteq \mathcal{P}_{yes}$  and  $g\mathcal{P}_{no}g^{-1} \subseteq \mathcal{P}_{no}$ .*

► **Definition 1.9** (Invariant rings). Let  $\mathbb{C}[X, Y]_d$  be the ring of complex polynomials in matrix variables  $X = (x_{i,j})_{1 \leq i, j \leq d}$  and  $Y = (y_{i,j})_{1 \leq i, j \leq d}$ . Observe that there is an action of  $G$  on any  $f(X, Y) \in \mathbb{C}[X, Y]$  by simultaneous conjugation  $g \cdot f(X, Y) = f(gXg^{-1}, gYg^{-1})$ .

The invariant ring  $\mathbb{C}[X, Y]_d^G$  is the subring of polynomials in  $\mathbb{C}[X, Y]_d$  satisfying  $g \cdot f = f$  for all  $g \in G$ .

The general theory of invariant theory guarantees the existence and finiteness of a generating set for the invariant ring  $\mathbb{C}[X, Y]^G$  for all compact groups, which includes all finite groups, the unitary group, and products of unitary groups as special cases. Furthermore, the following proposition connects the invariant ring to property testers for  $G$ -invariant properties.

► **Proposition 1.10** (Symmeterization for  $G$ -invariant properties). Suppose  $\mathcal{P} = (\mathcal{P}_{yes}, \mathcal{P}_{no})$  is a  $G$ -invariant  $d$ -dimensional unitary property. If there is a  $T$ -query tester for  $\mathcal{P}$  that accepts yes instances with probability at least  $a$  and no instances with probability at most  $b$ , then there exists a self-adjoint degree- $2T$  polynomial  $q$  in the invariant ring  $\mathbb{C}[X, X^*]_d^G$  satisfying

- If  $U \in \mathcal{P}_{yes}$ , then  $q(U, U^*) \geq a$ .
- If  $U \in \mathcal{P}_{no}$ , then  $q(U, U^*) \leq b$ .

**Proof.** Refer to Section 3.1 of the full version [41]. ◀

While Proposition 1.10 at first may seem difficult to apply, the invariant ring has been characterized in numerous cases. Depending on the group, the associated invariant ring may have a much simpler description than the full polynomial ring, making it easier to prove degree lower bounds. For instance, in the case where  $G$  is the full unitary group, the invariant polynomials are exactly symmetric polynomials in the eigenvalues of  $U$  and the adjoint  $U^*$ .

We illustrate this connection to invariant theory by considering property testing questions related to *entanglement* of quantum states, which is a central concept in quantum information theory. Recall that a state  $|\psi\rangle \in \mathbb{C}^d \otimes \mathbb{C}^d$  is *entangled* if it cannot be written as a tensor product of two states  $|\psi_1\rangle \otimes |\psi_2\rangle$  where  $|\psi_1\rangle, |\psi_2\rangle \in \mathbb{C}^d$ . The property of being entangled is invariant under the *local unitary* group instead of the full unitary group.

► **Definition 1.11** (Local Unitary Group). Let  $d_1, d_2 \geq 2$ . The local unitary group  $\text{LU}(d_1, d_2)$  is the subgroup  $U(d_1) \times U(d_2)$  of  $U(d_1 d_2)$  consisting of all unitaries of the form  $g \otimes h$  where  $g \in U(d_1), h \in U(d_2)$ .

Furthermore, the *entanglement entropy* of a state  $|\psi\rangle \in \mathbb{C}^d \otimes \mathbb{C}^d$  can be used as a measurement of how entangled the state is. Numerous definitions of entanglement entropy have been proposed in the physics and quantum information literature; we use the following definition of entanglement entropy in this work.

► **Definition 1.12** (Rényi 2-entropy). Given a state  $|\psi\rangle \in \mathbb{C}^d \otimes \mathbb{C}^d$  with reduced density matrix  $\rho$ , the Rényi 2-entropy of  $|\psi\rangle$  is defined as  $H_2(|\psi\rangle) = -\log \text{Tr}(\rho^2)$ .

We now define the Entanglement Entropy problem as the task of distinguishing between high and low entropy states.

► **Definition 1.13** (Entanglement Entropy Problem). Let  $0 < a < b \leq \log d$ . Given oracle access to a reflection oracle  $U = I - 2|\psi\rangle\langle\psi|$  where  $|\psi\rangle \in \mathbb{C}^d \otimes \mathbb{C}^d$ , decide whether or not the state  $|\psi\rangle$  satisfies one of the following two conditions, promised one of the following is the case:

- Low entropy case:  $H_2(|\psi\rangle) \leq a$

■ *High entropy case:  $H_2(|\psi\rangle) \geq b$*

Since entanglement entropy of a state is an LU-invariant quantity (i.e.  $H_2((g \otimes h)|\psi\rangle) = H_2(|\psi\rangle)$  for all unitaries  $g$  and  $h$ ), the Entanglement Entropy problem corresponds to an LU-invariant unitary property, opening the door to exploiting well-known results from invariant theory to prove query lower bounds. Indeed, leveraging a characterization of LU-invariant polynomials by Procesi [38] and Brauer [17], and specializing it to the Entanglement Entropy problem, we obtain the following lower bound using our generalized polynomial method and Proposition 1.10:

► **Theorem 1.14.** *Assume  $a \geq 5$ . Given parameters  $a < b \leq \log d$ , any tester must make  $\Omega(\exp(a/4))$  queries to distinguish between the low and high entropy cases in the Entanglement Entropy problem.*

**Proof.** Refer to Section 5.2 of the full version [41]. ◀

We hope that this connection between invariant theory and quantum query complexity can be used as a general framework to prove new lower bounds.

### 1.3 Property Testing with Quantum Proofs

We also study unitary property testing with *quantum proofs*: in addition to getting query access to the instance  $U$ , the tester also receives an additional quantum state called a *proof* that supposedly certifies that  $U$  is a *yes* instance. On one hand, having access to a quantum proof can significantly reduce the number of queries to  $U$  needed. On the other hand, the proof state is not trusted and must be verified: if  $U$  is a *no* instance it must be rejected with high probability no matter what proof was provided. Since this definition is analogous to the definition of the complexity class QMA, we call this the “QMA property testing model”. Similarly we call the standard definition of unitary property testing (without quantum proofs) the “BQP property testing model”.

An illustration of QMA property testing is that of unstructured search, where the goal is to determine whether  $U|\psi\rangle = -|\psi\rangle$  for some state  $|\psi\rangle$  (and acts as the identity everywhere else) or whether  $U = I$ , promised that one is the case. If  $U$  is  $d$ -dimensional, then the generalized polynomial method (see Section 4.2 of [41]) implies the BQP query complexity of this problem is  $\Theta(\sqrt{d})$ , but on the other hand the QMA query complexity of this problem is 1: given a proof state  $|\theta\rangle$ , the tester can verify using a single query whether  $U$  applies a nontrivial phase to  $|\theta\rangle$ , in which case the tester would accept, and otherwise reject. Thus quantum proofs can dramatically reduce the query complexity of a problem.

The QMA property testing model motivates the following questions: which problems admit query speedups when quantum proofs are provided? For which problems are quantum proofs useless? In addition to the BQP property testing lower bounds mentioned above, we prove QMA property testing lower bounds for the Approximate Dimension, Recurrence Time, and Entanglement Entropy problems.

We note that in the BQP setting, our lower bounds can also be obtained by other methods, such as the “hybrid method” of [13]. However, it is unclear how to apply this method in the QMA setting, and hence the polynomial method appears necessary to prove non-trivial QMA lower bounds. Furthermore, even in the BQP setting, we believe that the polynomial method provides a clean and simple method to prove lower bounds compared to other methods.

The QMA lower bound for Approximate Dimension is obtained by observing that any QMA tester for Approximate Dimension is also a QMA tester for the classical Approximate Counting problem (i.e., counting the Hamming weight of an input string). Thus, the lower bound follows immediately from the QMA lower bound on Approximate Counting proved by Aaronson, et al. [4]:

► **Theorem 1.15** (QMA lower bound for Approximate Dimension). *Suppose there is a  $T$ -query algorithm that solves the Approximate Dimension problem (i.e. deciding whether a  $d$ -dimensional unitary encodes a subspace of dimension at least  $2w$  or at most  $w$ ) with the help of a  $m$ -qubit proof. Then either  $m = \Omega(w)$ , or  $T \geq \Omega(\sqrt{\frac{d}{w}})$ .*

**Proof.** Refer to Section 4.1 of the full version [41]. ◀

As with the BQP lower bound for the Recurrence Time problem, the QMA lower bounds for the Recurrence Time problem requires more work than leveraging lower bounds on a related classical problem. However, using a similar technique as the BQP lower bound, we obtain the following:

► **Theorem 1.16** (QMA lower bound for the Recurrence Time problem). *Let  $\epsilon \leq \frac{1}{2\pi}$ . Suppose there is a  $T$ -query algorithm that solves the Recurrence Time problem for  $d$ -dimensional unitaries with the help of an  $m$ -qubit proof. Then either  $m \geq \Omega(d)$ , or  $T \geq \Omega(\max(\sqrt{\frac{d}{m}}, \frac{t}{m}, \frac{1}{\epsilon}))$ .*

**Proof.** Refer to Section 4.2 of the full version [41]. ◀

Finally, we also adapt the technique for the BQP lower bound for the Entanglement Entropy problem, to prove a QMA lower bound for the same problem.

► **Theorem 1.17** (QMA lower bound for the Entanglement Entropy problem). *Assume  $a \geq 5$  and  $a < b \leq \log d$ . Suppose there is a  $T$ -query algorithm that solves the entanglement entropy problem with the help of an  $m$ -qubit witness, then  $mT \geq \Omega(\exp(a/4))$ .*

**Proof.** Refer to Section 5.2 of the full version [41]. ◀

We note that we are able to give an algorithm for the entanglement entropy problem using  $O(\exp(a))$  queries and a certificate size with  $O(\exp(a))$  qubits as long as the gap satisfies  $b \geq 2a$ . Furthermore, our best upper bound in the QMA setting for the Recurrence Time problem is identical to the BQP upper bound (Theorem 1.7), which uses  $O(\frac{t\sqrt{d}}{\epsilon})$  queries. It is open whether or not the query complexity in the QMA setting can be improved by making use of the witness, or if the lower bounds can be tightened.

## 1.4 QMA vs. QMA(2)

In addition to the QMA model of property testing with the help of a quantum proof, we can also study what happens if we place restrictions on the proof states allowed. For example, what if the proof is guaranteed to be a classical string (i.e., a QCMA proof) or is unentangled across a fixed bipartition of qubits (i.e., a QMA(2) proof)?

For example, consider the problem of testing whether or not a given unitary  $U$  was the identity  $I$  or a reflection  $I - 2|\psi\rangle\langle\psi|$  where  $|\psi\rangle$  is an  $n$ -qubit state. As we observed in the previous section, this problem can be solved using one query given access to a proof state  $|\psi\rangle$ . However, Aaronson and Kuperberg [5] showed that if the proof given was an  $m$ -bit classical string, any quantum algorithm must use  $\Omega(\sqrt{\frac{2^n}{m+1}})$  queries to distinguish between the two cases. In particular, the result was used by Aaronson and Kuperberg in [5] to give a quantum oracle separation between QMA and QCMA.

The complexity class QMA( $k$ ) is defined as the class of problems verifiable by a polynomial time quantum circuit with access to  $k \geq 2$  unentangled proofs. It was shown in [27] that for any constant  $k > 2$ , we have QMA( $k$ ) = QMA(2), as any QMA( $k$ ) verifier can be simulated by a QMA(2) verifier.

As discussed in Aaronson’s survey paper on quantum query complexity in [2], an oracle separation between QMA(2) and QMA is a notorious open problem in quantum complexity theory. There is evidence that QMA(2) could be more powerful than QMA:

- The existence of QMA(2) protocols for the verification of NP-complete problems (eg. graph 3-colouring) using a logarithmic number of qubits, as outlined in [14]. If there were a QMA protocol for these problems using a logarithmic number of qubits, then  $\text{NP} \subseteq \text{QMA}_{\log} = \text{BQP}$ , which is considered unlikely [33]. Otherwise, if there exists a QMA protocol with a sublinear number of qubits for 3-SAT or 3-colouring, then the (classical) Exponential Time Hypothesis is false [3, 19].<sup>5</sup>
- Certain problems in quantum chemistry, specifically the pure state  $N$ -representability problem, known to have QMA(2) protocols but not QMA protocols [31].
- The non-existence of a product test using local quantum operations and classical communication (LOCC) as proven in [27]. If an LOCC product test existed, then  $\text{QMA}(2) = \text{QMA}$ .

On the other hand, despite many years of study, the only complexity inclusions about QMA(2) known are  $\text{QMA} \subseteq \text{QMA}(2) \subseteq \text{NEXP}$ , a vast gap in the complexity-theoretic landscape. A first step towards showing that QMA(2) is indeed more powerful than QMA would be to identify an oracle relative to which QMA(2) is different than QMA. This would already have very interesting consequences in quantum information theory, such as ruling out the existence of disentanglers [3].

In this paper we identify a unitary property testing problem, for which if we can prove a strong QMA lower bound would immediately imply an oracle separation between QMA(2) and QMA. While we do not obtain a lower bound, we present some observations that may be helpful towards eventually obtaining the desired oracle separation.

In order to define the problem, we first have to define the notion of an  $\epsilon$ -completely entangled subspace. This is a subspace  $S \subseteq \mathbb{C}^d \otimes \mathbb{C}^d$  such that all states  $|\theta\rangle \in S$  are  $\epsilon$ -far in trace distance from any product state  $|\psi\rangle \otimes |\phi\rangle$ . It is known, via the probabilistic method, that there exist subspaces of dimension  $\Omega(d^2)$  that are  $\Omega(1)$ -completely entangled [28]. We now introduce the Entangled Subspace problem:

► **Definition 1.18** (Entangled Subspace problem). *Let  $0 \leq a < b < 1$  be constants. The  $(a, b)$ -Entangled Subspace problem is to decide, given oracle access to a unitary  $U = I - 2\Pi$  where  $\Pi$  is the projector onto a subspace  $S \subseteq \mathbb{C}^d \otimes \mathbb{C}^d$ , whether*

- (yes case)  $S$  contains a state  $|\theta\rangle$  that is  $a$ -close to a product state  $|\psi\rangle \otimes |\phi\rangle$ .
  - (no case)  $S$  is  $b$ -completely entangled
- promised that one is the case.*

First, we observe that the Entangled Subspace property is LU-invariant: applying local unitaries  $g \otimes h$  to a subspace  $S$  preserves whether it is a *yes* instance or a *no* instance of the problem. Thus one can hope to prove query lower bounds for the Entangled Subspace problem in both the BQP and QMA setting using our generalized polynomial method and tools from invariant theory.

Next, we observe that there is in fact a QMA(2) upper bound for the Entangled Subspace problem:

---

<sup>5</sup> We can prove there is an oracle relative to which NP does not have sublinear-sized QMA proofs, using an argument of Aaronson similar to that in [1] that combines the “guessing lemma” and the polynomial method.

► **Proposition 1.19** (QMA(2) upper bound for the Entangled Subspace problem). *The Entangled Subspace problem can be solved by a QMA(2) tester, meaning that the tester receives a proof state in the form  $|\psi\rangle \otimes |\varphi\rangle$  of  $\text{poly log}(d)$  qubits, makes  $\text{poly log}(d)$  queries to the unitary  $U$ , and can distinguish between yes and no cases with constant bias. We call this tester the product test verifier.*

**Proof.** Refer to Section 6.1 of the full version [41]. ◀

The QMA(2) tester from Proposition 1.19 is based on the *product test*, which is a procedure for detecting whether a state  $|\theta\rangle$  is close to a product state  $|\psi\rangle \otimes |\phi\rangle$ , given access to  $k \geq 2$  copies  $|\theta\rangle^{\otimes k}$ . We use of a generalization of the product test analysis of [43] that applies for all  $k \geq 2$ , which they showed for the case  $k = 2$ .

We conjecture the following QMA lower bound on the Entangled Subspace problem.

► **Conjecture 1.20.** *Any QMA tester for the Entangled Subspace problem that makes  $T$  queries to the oracle and receives an  $m$ -qubit witness must have either  $m$  or  $T$  be superpolynomial in  $\log d$ .*

If this conjecture is true, then this would imply the existence of a quantum oracle that separates QMA from QMA(2): the oracle would encode, for each QMA tester, an instance of the Entangled Subspace problem that the tester decides incorrectly.

As a first step towards understanding whether Conjecture 1.20 is true, we analyze the behavior of product test verifier from Proposition 1.19 in the QMA setting. This means that instead of getting a proof state that is promised to be unentangled across a bipartition, the product test verifier receives a single pure state that may be entangled everywhere. We show that the product test verifier loses its soundness in the QMA setting:

► **Theorem 1.21.** *There exists a 6-dimensional completely entangled subspace  $S$  and an entangled proof state  $|\theta\rangle$  that the product test verifier accepts with probability 1 (even though it rejects all product proof states  $|\psi\rangle \otimes |\phi\rangle$  with high probability).*

**Proof.** Refer to Section 6.3 of the full version [41]. ◀

The construction of the “fooling” subspace  $S$  and the proof state  $|\theta\rangle$  is completely explicit and combinatorial; we believe it can suggest how more general QMA verifiers (beyond the product test) can be fooled, which would give insight towards proving Conjecture 1.20. Indeed in Appendix B of [41], we characterize all states that pass the product test with probability one, which enables the construction of other “fooling” subspaces for the product test.

### Constraints on the Conjecture

We now identify constraints on the conjecture: we show that we cannot hope to prove a superpolynomial QMA lower bound on the Entangled Subspace problem when only considering one-dimensional subspaces. This is a consequence of the following general statement:

► **Lemma 1.22.** *Let  $\mathcal{P}$  denote a property where the instances are unitaries encoding a one-dimensional subspace (i.e. a pure state):  $U = I - 2|\psi\rangle\langle\psi|$  for some state  $|\psi\rangle$ . Suppose that there is a  $T$ -query QMA(2) tester that decides  $\mathcal{P}$ , with the condition that a valid proof state for yes instances is  $|\psi\rangle^{\otimes 2}$ . Then there exists a  $O(T)$ -query QMA tester that also decides  $\mathcal{P}$ .*

**Proof.** Refer to Section 6.2 of the full version [41]. ◀

## 96:12 Unitary Property Testing Lower Bounds by Polynomials

Let  $\mathcal{P}$  denote the property testing problem where in the *yes* case the unitary encodes a product state  $|\varphi\rangle \otimes |\xi\rangle$  and in the *no* case it encodes an entangled state. Proposition 1.19 yields an efficient QMA(2) tester for this property. This lemma shows that there is also a QMA tester for this property. Intuitively, this theorem is saying that property testing questions related to quantum *states* are insufficient to give an oracle separation between QMA and QMA(2). On the other hand, as far as we know, Conjecture 1.20 potentially could be proved by giving a QMA lower bound for the Entangled Subspace problem restricted to *two-dimensional* subspaces.

### Average Case Problems

Finally, we also propose two *average case* variants of the Entangled Subspace problem, in which the task is to distinguish between two distributions over unitaries  $U$ .

► **Definition 1.23** (Planted Product State Problem). *Let  $0 < s < d^2$  denote an integer parameter. Consider the following two distributions over subspaces  $S$  of  $\mathbb{C}^d \otimes \mathbb{C}^d$ :*

- **No planted state:**  $S$  is a Haar-random subspace of dimension  $s$ .
- **Has planted state:**  $S$  is an  $(s + 1)$ -dimensional subspace chosen by taking the span of a Haar-random  $s$ -dimensional subspace with a product state  $|\psi\rangle \otimes |\phi\rangle$  for Haar-random  $|\psi\rangle, |\phi\rangle$ .

*The Planted Product State problem is to distinguish, given oracle access to a unitary  $U = I - 2\Pi$  encoding a subspace  $S$ , whether  $S$  was sampled from the **No planted state** distribution (no case) or the **Has planted state** distribution (yes case), promised that one is the case.*

► **Definition 1.24** (Restricted Dimension Counting Problem). *Let  $0 < t \leq d$  and  $0 < r \leq t^2$  denote integer parameters. Consider the following distribution, parameterized by  $(t, r)$ , over subspaces  $S \subseteq \mathbb{C}^d \otimes \mathbb{C}^d$ :*

- Sample Haar-random  $t$ -dimensional subspaces  $R, Q \subseteq \mathbb{C}^d$ .
- Sample a Haar-random  $r$ -dimensional subspace of  $S \subseteq R \otimes Q$ .

*Let  $0 < C_1 < C_2 < 1$  denote constants. The Restricted Dimension Counting problem is to decide, given query access to a unitary  $U = I - 2\Pi$  encoding a subspace  $S$ , whether  $S$  was sampled from either the  $(t, C_1 t^2)$  distribution or  $(t, C_2 t^2)$  distribution, promised that one is the case.*

The relationship between these two average case problems and the Entangled Subspace problem is captured by the following propositions.

► **Proposition 1.25.** *If  $S$  is sampled from the **Has planted state** distribution of the Planted Product State problem, then it is a yes instance of the Entangled Subspace problem. If  $S$  is sampled from the **No planted state** distribution with  $s = Cd^2$  for some sufficiently small constant  $C > 0$ , then it is a no instance with overwhelming probability.*

**Proof.** Refer to Section 6.4 of the full version [41]. ◀

► **Proposition 1.26.** *There exist constants  $0 < C_1 < C_2 < 1$  such that if  $S$  is sampled from the  $(t, C_1 t^2)$  distribution from the Restricted Dimension Counting problem, it is a no instance of the Entangled Subspace problem with overwhelming probability. If it is sampled from the  $(t, C_2 t^2)$  distribution, then it is a yes instance with overwhelming probability.*

**Proof.** Refer to Section 6.4 of the full version [41]. ◀

These two propositions are proved using methods from random matrix theory. Proposition 1.19 in turn implies that the Planted Product State and Restricted Dimension Counting problems can be solved by a QMA(2) tester with overwhelming probability. We further conjecture that in fact these two problems cannot be solved in polynomial time by a QMA tester. This conjecture would clearly also imply an oracle separation between QMA and QMA(2).

► **Conjecture 1.27.** *Any QMA tester that solves the Planted Product State or Restricted Dimension Counting problems with constant probability either requires a proof state of super-polynomial size, or make super-polynomially many queries to the oracle.*

As evidence towards Conjecture 1.27, we show a lower bound against QCMA testers.

► **Theorem 1.28.** *Any  $T$ -query quantum algorithm solving the Planted Product State problem with the help of an  $m$ -bit classical witness must have  $m$  or  $T$  superpolynomial in  $\log d$ .*

**Proof.** Refer to Section 6.6 of the full version [41]. ◀

We believe that these two average case problems are connected to several interesting topics. The first is the QMA versus QMA(2) problem, of course. Their formulation also suggests that tools from random matrix theory can be brought to bear to study them. Finally, the Restricted Dimension Counting problem is, as the name suggests, a more structured, special case of the general Approximate Dimension problem, for which we proved a strong QMA lower bound! Perhaps the techniques used to prove lower bounds on the Approximate Dimension problem can be extended to the Restricted Dimension Counting problem.

## 1.5 Related Work

In quantum query complexity, there have been two main paradigms for proving lower bounds for query complexity, namely the polynomial method [10] and the adversary method [45]. The methods are generally incomparable, as there are problems where one method is able to prove a tight lower bound but the other cannot. For instance, the collision problem [30] is a case where the polynomial method proves a tight lower bound but the adversary method provably fails to do so. On the other hand, Ambanis [8] constructed an example where the adversary method is provably better than the polynomial method. Furthermore, for evaluation of Boolean functions, the general adversary method characterizes the quantum query complexity up to constant factors [39].

However, in their original form, both methods assume that the quantum oracle encodes a Boolean function  $f : \{0, 1\}^n \rightarrow \{0, 1\}$ . A similar generalization of the adversary method to unitary property testing problems was introduced by Belovs [11]. In particular, it was applied to the approximate counting problem [12], with a symmetrization technique based on the representation theory of the symmetric group.

Next, unitary property testing questions were also considered by Aharonov et al. in [7], who introduced a model called quantum algorithmic measurement and studied the query complexity of problems in this model. Their techniques used to prove lower bounds in their model primarily rely on the combinatorics of Weingarten functions, which discuss in Section 3.2 of the full version [41]. However, their model allows for interaction between a prover and a verifier, whereas we discuss query lower bounds in the non-interactive setting. Similar sample complexity lower bounds for quantum machine learning problems were proven using Weingarten calculus techniques in the works of [20] and [21].

## 96:14 Unitary Property Testing Lower Bounds by Polynomials

Next, Kretschmer [29] as well as Copeland and Pommersheim [23] also studied query complexity problems where the oracle does not necessarily encode a Boolean function. In particular, [29] studied the quantum query complexity of the heavy output generation problem and is motivated by recently quantum supremacy experiments, and [23] studied problems where the set of possible oracles form a representation of a group. However, there is a substantial difference between lower bound techniques in our works, which relies on linear programming duality in [29] and group character theory in [23].

Finally, we also note that Gur et al. [26] consider the query complexity of estimating the von Neumann entropy to a multiplicative factor of  $\alpha > 1$ , promised that the entropy of the given quantum state is not too small. They established a  $\Omega(d^{\frac{1}{3\alpha^2}})$  lower bound on the query complexity by reduction to the collision problem lower bound of [30]. On the other hand, the strongest lower bound we are able to prove is a  $\Omega(d^{\frac{1}{4\alpha}})$  lower bound by setting parameters  $a = \frac{1}{\alpha} \log d$  and  $b = \log d$  in the definition of our problem. However, our results are incomparable with their results since the oracle access models we consider are different.

### 2 Open Questions

We end by describing some open problems and future directions.

#### Strong QMA Lower Bounds for the Entangled Subspace Problem

Can one show that any QMA tester for the Entangled Subspace problem requires either a superpolynomial number of queries, or a superpolynomial sized witness? This would yield a (quantum) oracle separation between QMA and QMA(2), and in particular would rule out the existence of so-called “disentangler” [3].

#### Better Query Upper Bounds

Are the bounds proven using the generalized polynomial method tight? In particular, the following gaps remain:

- We have shown that there is a  $O(\frac{t\sqrt{d}}{\epsilon})$  upper bound and a  $\Omega(\max(\frac{t}{\epsilon}, \sqrt{d}))$  lower bound in the BQP setting for the recurrence problem and used this bound to prove a similar lower bound in the QMA setting. Is there a better lower or upper bound in either the BQP or QMA settings? However, a more sophisticated symmetrization technique may be required to improve the lower bound.
- We expect the BQP lower bound in Theorem 1.14 for the entanglement entropy can be improved by using a more creative application of the polynomial method.

#### Improving Theorem 1.21

Is the counterexample of Theorem 1.21 tight, in the sense that there are no examples that fool the verifier in dimensions 2, 3, 4, or 5? Otherwise, if there was an example that fools the verifier in dimension 2, this would give additional evidence that the Entangled Subspace problem in low dimensions is already hard for QMA.

#### Other Applications of the Generalized Polynomial Method

What are other applications of the generalized polynomial method? For instance, Procesi [38] has characterized the invariants of matrix tuples under conjugation by the general linear, unitary, orthogonal, and symplectic groups. Are there natural problems in quantum query complexity that display other, non-unitary symmetries?

## Applications of Weingarten Calculus

Can the Weingarten calculus techniques introduced in Section 3.2 of the full version [41] be used to prove lower bounds on unitary property testing problems? In particular, could it be used to prove degree lower bounds for acceptance probability polynomials?

## A Generalized Dual Polynomial Method?

A line of works established tight quantum query lower bounds on classical problems by employing a method of *dual polynomials* [42, 44, 18]. The goal of this method is to prove degree lower bounds of acceptance probability polynomials, but instead of symmetrizing the polynomials to obtain a polynomial of one or two variables, one instead takes advantage of *linear programming duality* to prove the degree lower bounds; this involves constructing objects known as dual polynomials. A natural question would be to investigate whether the method of dual polynomials can be extended to prove query lower bounds for unitary property testing.

---

### References

- 1 Scott Aaronson. Impossibility of succinct quantum proofs for collision-freeness. *arXiv preprint*, 2011. [arXiv:1101.0403](https://arxiv.org/abs/1101.0403).
- 2 Scott Aaronson. Open problems related to quantum query complexity. *ACM Transactions on Quantum Computing*, 2(4):1–9, 2021.
- 3 Scott Aaronson, Salman Beigi, Andrew Drucker, Bill Fefferman, and Peter Shor. The power of unentanglement. In *2008 23rd Annual IEEE Conference on Computational Complexity*, pages 223–236. IEEE, 2008.
- 4 Scott Aaronson, Robin Kothari, William Kretschmer, and Justin Thaler. Quantum lower bounds for approximate counting via Laurent polynomials. In *35th Computational Complexity Conference (CCC 2020)*. Schloss Dagstuhl-Leibniz-Zentrum für Informatik, 2020.
- 5 Scott Aaronson and Greg Kuperberg. Quantum versus classical proofs and advice. *Theory of Computing*, 3:129–157, 2007.
- 6 Scott Aaronson and Yaoyun Shi. Quantum lower bounds for the collision and the element distinctness problems. *Journal of the ACM (JACM)*, 51(4):595–605, 2004.
- 7 Dorit Aharonov, Jordan Cotler, and Xiao-Liang Qi. Quantum algorithmic measurement. *Nature communications*, 13(1):1–9, 2022.
- 8 Andris Ambainis. Polynomial degree vs. quantum query complexity. *Journal of Computer and System Sciences*, 72(2):220–238, 2006.
- 9 Andris Ambainis. Understanding quantum algorithms via query complexity. In *Proceedings of the International Congress of Mathematicians: Rio de Janeiro 2018*, pages 3265–3285. World Scientific, 2018.
- 10 Robert Beals, Harry Buhrman, Richard Cleve, Michele Mosca, and Ronald de Wolf. Quantum lower bounds by polynomials. *Journal of the ACM (JACM)*, 48(4):778–797, 2001.
- 11 Aleksandrs Belovs. Variations on quantum adversary. *arXiv preprint*, 2015. [arXiv:1504.06943](https://arxiv.org/abs/1504.06943).
- 12 Aleksandrs Belovs and Ansis Rosmanis. Tight quantum lower bound for approximate with quantum states. *arXiv preprint*, 2020. [arXiv:2002.06879](https://arxiv.org/abs/2002.06879).
- 13 Charles H Bennett, Ethan Bernstein, Gilles Brassard, and Umesh Vazirani. Strengths and weaknesses of quantum computing. *SIAM journal on Computing*, 26(5):1510–1523, 1997.
- 14 Hugue Blier and Alain Tapp. All languages in NP have very short quantum proofs. In *2009 Third International Conference on Quantum, Nano and Micro Technologies*, pages 34–37, 2009. [doi:10.1109/ICQNM.2009.21](https://doi.org/10.1109/ICQNM.2009.21).
- 15 P Bocchieri and A Loinger. Quantum recurrence theorem. *Physical Review*, 107(2):337, 1957.

- 16 Zvika Brakerski, Devika Sharma, and Guy Weissenberg. Unitary subgroup testing. *arXiv preprint*, 2021. [arXiv:2104.03591](https://arxiv.org/abs/2104.03591).
- 17 Richard Brauer. On algebras which are connected with the semisimple continuous groups. *Annals of Mathematics*, pages 857–872, 1937.
- 18 Mark Bun, Robin Kothari, and Justin Thaler. The polynomial method strikes back: Tight quantum query bounds via dual polynomials. In *Proceedings of the 50th Annual ACM SIGACT Symposium on Theory of Computing*, pages 297–310, 2018.
- 19 Jing Chen and Andrew Drucker. Short multi-prover quantum proofs for SAT without entangled measurements. *arXiv preprint*, 2010. [arXiv:1011.0716](https://arxiv.org/abs/1011.0716).
- 20 Sitan Chen, Jordan Cotler, Hsin-Yuan Huang, and Jerry Li. A hierarchy for replica quantum advantage. *arXiv preprint*, 2021. [arXiv:2111.05874](https://arxiv.org/abs/2111.05874).
- 21 Sitan Chen, Jordan Cotler, Hsin-Yuan Huang, and Jerry Li. Exponential separations between learning with and without quantum memory. In *2021 IEEE 62nd Annual Symposium on Foundations of Computer Science (FOCS)*, pages 574–585. IEEE, 2022.
- 22 Thomas Chen, Shivam Nadimpalli, and Henry Yuen. Testing and learning quantum juntas nearly optimally. *arXiv preprint*, 2022. [arXiv:2207.05898](https://arxiv.org/abs/2207.05898).
- 23 Daniel Copeland and Jamie Pommersheim. Quantum query complexity of symmetric oracle problems. *Quantum*, 5:403, 2021.
- 24 Marcel Dall’Agnol, Tom Gur, Subhayan Roy Moulik, and Justin Thaler. Quantum proofs of proximity. *arXiv preprint*, 2021. [arXiv:2105.03697](https://arxiv.org/abs/2105.03697).
- 25 Ronald de Wolf. A note on quantum algorithms and the minimal degree of epsilon-error polynomials for symmetric functions. *arXiv preprint*, 2008. [arXiv:0802.1816](https://arxiv.org/abs/0802.1816).
- 26 Tom Gur, Min-Hsiu Hsieh, and Sathyawageeswar Subramanian. Sublinear quantum algorithms for estimating von Neumann entropy. *arXiv preprint*, 2021. [arXiv:2111.11139](https://arxiv.org/abs/2111.11139).
- 27 Aram W Harrow and Ashley Montanaro. Testing product states, quantum Merlin-Arthur games and tensor optimization. *Journal of the ACM (JACM)*, 60(1):3, 2013.
- 28 Patrick Hayden, Debbie W. Leung, and Andreas Winter. Aspects of generic entanglement. *Communications in Mathematical Physics*, 265(1):95–117, March 2006. doi:10.1007/s00220-006-1535-6.
- 29 William Kretschmer. The Quantum Supremacy Tsirelson Inequality. *Quantum*, 5:560, October 2021. doi:10.22331/q-2021-10-07-560.
- 30 Samuel Kutin. A quantum lower bound for the collision problem. *arXiv preprint*, 2003. [arXiv:quant-ph/0304162](https://arxiv.org/abs/quant-ph/0304162).
- 31 Yi-Kai Liu, Matthias Christandl, and Frank Verstraete. Quantum computational complexity of the  $n$ -Representability Problem: QMA complete. *Physical Review Letters*, 98(11), March 2007. doi:10.1103/physrevlett.98.110503.
- 32 Olivier Marchal. Matrix models, Toeplitz determinants and recurrence times for powers of random unitary matrices, 2014. doi:10.48550/arXiv.1412.3085.
- 33 Chris Marriott and John Watrous. Quantum Arthur–Merlin games. *Computational Complexity*, 14(2):122–152, 2005.
- 34 Marvin Minsky and Seymour Papert. *Perceptrons*. MIT press, 2017.
- 35 Ashley Montanaro and Ronald de Wolf. A survey of quantum property testing. *arXiv preprint*, 2013. [arXiv:1310.2035](https://arxiv.org/abs/1310.2035).
- 36 Noam Nisan and Mario Szegedy. On the degree of Boolean functions as real polynomials. *Computational complexity*, 4(4):301–313, 1994.
- 37 Ramamohan Paturi. On the degree of polynomials that approximate symmetric Boolean functions (preliminary version). In *Proceedings of the Twenty-Fourth Annual ACM Symposium on Theory of Computing*, STOC ’92, pages 468–474, New York, NY, USA, 1992. Association for Computing Machinery. doi:10.1145/129712.129758.
- 38 C Procesi. The invariant theory of  $n \times n$  matrices. *Advances in Mathematics*, 19(3):306–381, 1976. doi:10.1016/0001-8708(76)90027-X.

- 39 Ben W Reichardt. Reflections for quantum query algorithms. In *Proceedings of the Twenty-Second Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 560–569. SIAM, 2011.
- 40 Benoit Saussol. An introduction to quantitative Poincaré recurrence in dynamical systems. *Reviews in Mathematical Physics*, 21(08):949–979, 2009.
- 41 Adrian She and Henry Yuen. Unitary property testing lower bounds by polynomials. *arXiv preprint*, 2022. [arXiv:2210.05885](https://arxiv.org/abs/2210.05885).
- 42 Alexander A Sherstov. The intersection of two halfspaces has high threshold degree. *SIAM Journal on Computing*, 42(6):2329–2374, 2013.
- 43 Mehdi Soleimanifar and John Wright. Testing matrix product states. In *Proceedings of the 2022 Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 1679–1701. SIAM, 2022.
- 44 Robert Spalek. A dual polynomial for OR. *arXiv preprint*, 2008. [arXiv:0803.4516](https://arxiv.org/abs/0803.4516).
- 45 Robert Špalek and Mario Szegedy. All quantum adversary methods are equivalent. In *International Colloquium on Automata, Languages, and Programming*, pages 1299–1311. Springer, 2005.
- 46 Guoming Wang. Property testing of unitary operators. *Physical Review A*, 84(5):052328, 2011.