On the power of Quantum Proofs

Ran Raz *   Amir Shpilka †

Abstract

We study the power of quantum proofs, or more precisely, the power of Quantum Merlin-Arthur (QMA) protocols, in two well studied models of quantum computation: the black box model and the communication complexity model.

Our main results are obtained for the communication complexity model. For this model, we identify a complete promise problem for QMA protocols, the *Linear Subspaces Distance* problem. The problem is of geometrical nature: Each player gets a linear subspace of $\mathbb{R}^m$ and considers the sphere of unit vectors in that subspace. Their goal is to output 1 if the distance between the two spheres is very small (say, smaller than $0.1 \cdot \sqrt{2}$) and 0 if the distance is very large (say, larger than $0.9 \cdot \sqrt{2}$). We show that:

1. The QMA communication complexity of the problem is $O(\log m)$.
2. The (classical) MA communication complexity of the problem is $\Omega(m^{\epsilon})$ (for some $\epsilon > 0$).
3. The (standard) quantum communication complexity of the problem is $\Omega(\sqrt{m})$.

In particular, this gives an exponential separation between QMA communication complexity and MA communication complexity.

For the black box model we give several observations. First, we observe that the block sensitivity method, as well as the polynomial method for proving lower bounds for the number of queries, can both be extended to QMA protocols. We use these methods to obtain lower bounds for the QMA black box complexity of functions. In particular, we obtain a tight lower bound of $\Omega(N)$ for the QMA black box complexity of a random function, and a tight lower bound of $\Omega(\sqrt{N})$ for the QMA black box query complexity of $NOR(X_1, ..., X_N)$. In particular, this shows that any attempt to give short quantum proofs for the class of languages $Co-NP$ will have to go beyond black box arguments.

We also observe that for any boolean function $G(X_1, ..., X_N)$, if for both G and $\neg G$ there are QMA black box protocols that make at most T queries to the black box, then there is a classical deterministic black box protocol for G that makes $O(T^6)$ queries to the black box. In particular, this shows that in the black box model $QMA \cap Co-QMA = P$.

On the positive side, we observe that any (total or partial) boolean function $G(X_1, ..., X_N)$ has a QMA black box protocol with proofs of length N that makes only $O(\sqrt{N})$ queries to the black box.

Finally, we observe a very simple proof for the exponential separation (for promise problems) between QMA black box complexity and (classical) MA black box complexity (first obtained by Watrous).

---

*ranraz@wisdom.weizmann.ac.il, Department of Computer Science, Weizmann Institute, Rehovot 76100, ISRAEL. Part of this work was done when the author was a member at the Institute for Advanced Study, Princeton. Research supported by The Israel Science Foundation and by a European Union grant.

†amirs@deas.harvard.edu, Department of Arts and Sciences, Harvard University, Cambridge, Ma 02138. Supported by National Security Agency (NSA) and Advanced Research and Development Activity (ARDA) under Research Office (ARO) contract no. DAAD19-01-1-0506.

1 Introduction

The notion of *proof* is a central notion in complexity theory and in mathematics in general. Progress in understanding this notion has played a crucial role in the history of mathematics and in the development of complexity theory. Many of the most exciting ideas in complexity theory were originated by studying different views and aspects of the notion of proof.

Quantum computation is a relatively young field that studies the relative power of quantum and classical computational models. In 1994, Peter Shor proved that integers can be factored by quantum algorithms in polynomial time [28]. This gives a strong indication that quantum algorithms are significantly stronger than classical ones. Is the same true for *quantum proofs* ?

A classical proof for a certain statement (such as $x \in L$, for some input x and language L) is a string of bits that convinces a verifier that the statement is correct. Analogously, a quantum proof for a statement is a string of quantum bits (qubits) that convinces the verifier that the statement is correct. Since probabilities are inherent in quantum computation, we usually consider the probabilistic case were the verifier only wishes to be convinced "with high probability" that the statement is correct.

The probabilistic case is usually presented as an interaction between two players: Merlin, the infinitely powerful prover, who is supposed to supply the proof for the correctness of the statement, and Arthur, the verifier, who is usually limited to a polynomial time machine and is supposed to probabilistically verify the given proof. It should be the case that if the statement is not correct then Arthur rejects any conceivable proof, with high probability. Intuitively, the class of languages MA (Merlin-Arthur) is defined to be the class of all languages L such that for every input x the statement $x \in L$ has a proof of length at most polynomial in the length of x.

Formally, denote by $|x|$ the length of a string x, and let $T : \mathbb{N} \to \mathbb{N}$ and $W : \mathbb{N} \to \mathbb{N}$ be any two monotone functions. The class of languages $MA(T, W)$ is defined as follows:

Definition 1 *A language L is in $MA(T, W)$ if there exists a (classical) machine V, such that:*

1. *For any $x \in L$, there exists a string y of at most $W(|x|)$ bits, s.t.,*
$$\Pr[V(x, y) = 1] > 2/3.$$
2. *For any $x \notin L$ and any string y of at most $W(|x|)$ bits,*
$$\Pr[V(x, y) = 1] < 1/3.$$
3. *For any x, y, the running time of V on (x, y) is at most $T(|x|)$.*

We say, in this case, that V is an $MA(T, W)$ algorithm for the language L. The algorithm V is referred to as the *verifier*. The string y is referred to as the *proof* or *witness* for the algorithm. The function T is referred to as the *time complexity* of the algorithm. The function W is referred to as the *length of proofs* for the algorithm. The sum $T + W$ is referred to as the *total MA complexity* of the algorithm. The class of languages MA is usually defined as $MA(poly, poly)$, that is, a language L is in MA if for some polynomial $g : \mathbb{N} \to \mathbb{N}$ the language L is in $MA(g, g)$.

In the same way, the class of languages $QMA(T, W)$ is defined as follows:

Definition 2 *A language L is in $QMA(T,W)$ if there exists a quantum machine V, such that:*

1. *For any $x \in L$, there exists a quantum state $|y\rangle$ of at most $W(|x|)$ qubits, s.t.,*
$$\Pr[V(|x\rangle|y\rangle) = 1] > 2/3.$$

2. *For any $x \notin L$ and any quantum state $|y\rangle$ of at most $W(|x|)$ qubits,*
$$\Pr[V(|x\rangle|y\rangle) = 1] < 1/3.$$

3. *For any $x, |y\rangle$, the running time of V on $|x\rangle|y\rangle$ is at most $T(|x|)$.*

We say, in this case, that V is a $QMA(T,W)$ algorithm for the language L. As before, the class of languages QMA is usually defined as $QMA(poly, poly)$.

Note that definitions similar to Definition 1 and Definition 2 can be given for (almost) any other computational model. We can hence define $MA(T,W)$ and $QMA(T,W)$ algorithms and protocols for many other computational models, and in particular for the black box model and for the communication complexity model. In these definitions, T is always the complexity measure intrinsic in the specific computational model analyzed (e.g., the number of queries in the black box model, and the length of communication in the communication complexity model), and W is the length of the proof y supplied to V. Thus, the complexity of V is limited to T, while the length of the proof y is limited to W.

The class QMA was first defined by Kitaev, who also identified a complete problem for this class. This is very important because complete problems usually play crucial role in understanding the power of a computational model. For a recent survey that describes the class QMA and Kitaev's complete problem for it, see [3]. For a recent extension of Kitaev's result see [13]. For excellent surveys on quantum computation see [2, 24, 21, 16].

1.1 The Black Box Model

In the black box model, we have input variables $X_1, ..., X_N$ from some domain D (say, $D = \{0,1\}$ or $D = \{0,1\}^k$) and we wish to compute the value of a certain function $G(X_1, ..., X_N)$. We think of the input variables as hidden in a *black box*. The black box answers *queries* about the value of input variables. In the classical case, a query to the black box is an index i and the black box answers with the value X_i. The complexity measure is the total number of queries to the black box. That is, the black box complexity of a function G is the number of queries to the black box needed to compute G (by the best protocol on the worst case input). The classical black box complexity of a function G is also known as the *decision tree complexity* of G.

If the function G is defined for every $(X_1, ..., X_N) \in D^N$, we say that G is a *total function*. In some cases, we also consider functions that are only defined on a subset $\mathrm{Dom}(G) \subset D^N$. We say in this case that G is a *partial function* or a *promise problem*. We assume in this case that we are promised that $(X_1, ..., X_N) \in \mathrm{Dom}(G)$.

In the quantum case, a query to the black box is a quantum state $\sum_{i,z} \alpha_{i,z}|i\rangle|z\rangle$ and the black box answers with $\sum_{i,z} \alpha_{i,z}|i\rangle|z \oplus X_i\rangle$. The quantum black box complexity of G is, as before, the number of queries needed to compute G by the best quantum protocol. Since the quantum model is probabilistic by its nature, we allow a small probability of error and we only require that the protocol computes $G(X_1, ..., X_N)$ with high probability (for every $(X_1, ...X_N) \in \text{Dom}(G)$). We usually compare the quantum case to the classical probabilistic case, where we also allow a small probability of error.

We define the classes of $MA(T, W)$ black box protocols and $QMA(T, W)$ black box protocols in a similar manner to Definition 1 and Definition 2. Here, T is a limit on the number of queries to the black box and W is a limit on the length of the proof y given to the protocol. We think of the proof y as given to V for free. Thus, we do not count the number of accesses to the proof y. The formal definitions are given in Section 4.

The black box model is a well studied computational model in both the classical and the quantum case, and is particularly popular in the quantum case. In particular, Shor's algorithm for factoring is based on a black box algorithm for finding periodicity [28], and Grover's database search algorithm is stated and proved directly in the black box model. Grover actually shows that the quantum black box complexity of $OR(X_1, ..., X_N)$ is $O(\sqrt{N})$ [12].

Lower bounds for the quantum black box complexity of many functions are well known. Let us mention here two of the most interesting results. It was proved by Bennett et al that the quantum black box complexity of $OR(X_1, ..., X_N)$ is $\Omega(\sqrt{N})$ [6]. This shows that Grover's algorithm is optimal, and that any attempt to give fast quantum algorithms for the class of languages NP will have to go beyond black box arguments. Beals et al presented the polynomial method for proving lower bounds for quantum black box protocols and used that method to show that for any total function $G(X_1, ..., X_N)$ the quantum black box complexity of G is polynomially related to its classical deterministic black box complexity [5]. Thus, it is not possible to give exponential gaps between the quantum and classical black box complexity of total functions. An exponential gap between the quantum black box complexity and the classical (deterministic or probabilistic) black box complexity of partial functions was first proved by Simon [29].

For an excellent survey of the quantum and classical black box model and related results and open problem, see [9].

1.2 Communication Complexity

A communication complexity problem is given by 3 (usually finite) sets X, Y, Z and a function $f : X \times Y \to Z$. As before, we sometimes consider the case where f is a partial function (or promise problem). We have two players, Player I and Player II. Player I is given an input $x \in X$. Player II is given an input $y \in Y$. In cases that f is a partial function, we assume that we are promised that f is defined on the pair (x, y). The goal of the two players is to compute $f(x, y)$. The communication complexity of the problem is the number of bits the two players have to exchange between them in order to compute $f(x, y)$. Each player has an unlimited computational power and the players cooperate with each other. We count the amount of communication needed by the best protocol on the worst case input.

In the quantum communication complexity model, each of the players has an infinitely powerful quantum computer and the two players can exchange between them quantum bits (qubits), rather than classical bits. Since the quantum model is probabilistic by its nature, we allow a small probability of error and (as before) we only require that for every x, y the answer will be the correct value of

$f(x, y)$ with high probability. The quantum communication complexity of a problem is the amount of communication qubits needed by the best such protocol. We usually compare the quantum case to the classical probabilistic case.

We define the classes of $MA(T, W)$ communication complexity protocols and $QMA(T, W)$ communication complexity protocols in a similar manner to Definition 1 and Definition 2. Here, T is a limit on the communication complexity of the protocol and W is a limit on the length of a proof w given to the players. We think of the proof w as presented to the players, so the players can access the proof for free. In the quantum case, in order to prevent the players from using the proof as a communication channel, we make a restriction that only the first player can access the proof. That player can then communicate parts of the proof to the other player as part of the communication protocol. Since the interesting cases are those that have relatively short proofs, this doesn't change the communication complexity by much. (Several variants of the model can also be considered. Our results hold for all those versions). The formal definitions are given in Section 2.

The model of communication complexity was introduced by Yao [34]. Besides being very interesting in its own right, the model was found out to be relevant to many other complexity issues and has become a central complexity model. The quantum communication complexity model was introduced by Yao in [35] and was extensively studied since then.

An exponential gap between quantum communication complexity and classical (deterministic or probabilistic) communication complexity of partial functions was proved in [25]. It is still open whether or not a similar gap can be obtained for total functions. The best known gap for total functions is a quadratic gap, proved by Buhrman Cleve and Wigderson [10] for the *Set Disjointness* function. Several lower bounds for the quantum communication complexity of total functions are known. Among the most interesting results are the tight lower bound of $\Omega(n)$ for the *Inner Product* (of two binary vectors of length n), first proved by Kremer and Yao [17, 35], and the tight lower bound of $\Omega(\sqrt{n})$ for the *Set Disjointness* function, recently proved by Razborov [26].

For an excellent survey on communication complexity see [18]. For excellent surveys on quantum communication complexity see [30, 33].

1.3 Our Results

We study the power of QMA protocols in both the black box model and the communication complexity model.

Our main results are obtained for the communication complexity model. For this model, we give a complete promise problem for QMA protocols. The problem is of geometrical nature and is a finite precision variation of the following problem.

The *Linear Space Distance* (LSD) problem:
Each player gets a linear subspace of $\mathbb{R}^m$ and considers the sphere of unit vectors in that subspace. The promise is that the distance between the two spheres is either very small (say, smaller than $0.1 \cdot \sqrt{2}$) or very large (say, larger than $0.9 \cdot \sqrt{2}$). The goal is to output 1 if the distance is very small and 0 if the distance is very large.

The LSD problem is not a standard communication complexity problem, as the set of possible inputs for each player is infinite. Nevertheless, its communication complexity as well as MA and

QMA communication complexity can still be defined in the same way.

We show that the LSD problem satisfies the following properties:

1. There is a $QMA(T, W)$ communication complexity protocol for the LSD problem, such that $T = O(\log m)$ and $W = O(\log m)$. Moreover, the protocol has only two rounds of communication (one for each player). Thus, the LSD problem has a very efficient QMA communication complexity protocol.

2. Any communication complexity problem that has a $QMA(T, W)$ communication complexity protocol (with any number of rounds of communication) can be reduced to the LSD problem, with $\log m = poly(T + W)$. Thus, LSD is a complete problem for QMA communication complexity protocols.

3. In any $MA(T, W)$ communication complexity protocol for the LSD problem, $T + W = \Omega(m^{\epsilon})$ (for some constant $\epsilon > 0$). Thus, the LSD problem is very hard for (classical) MA communication complexity protocols.

4. In any (standard) quantum communication complexity protocol for the LSD problem, the communication complexity is $\Omega(\sqrt{m})$. Thus, the LSD problem is very hard for quantum communication complexity protocols.

In particular, this gives an exponential separation between QMA communication complexity and MA communication complexity. Our results actually give a similar gap between (standard) quantum communication complexity and (classical) MA communication complexity.

As mentioned above, the sets of inputs for the LSD problem is infinite. We define the finite precision variation of LSD, in order to get a communication complexity problem with finite sets of inputs (as is required by the standard definition of communication complexity). Each input for LSD can be described by $O(m^2)$ real variables. We define the problem $\widetilde{LSD}$ to be the same as LSD, but where each of these $O(m^2)$ variables is described by $O(\log m)$ bits (i.e., with polynomially good precision). The length of the inputs for the new problem is hence $O(m^2 \log m)$. All the results, stated above for the LSD problem, hold for the problem $\widetilde{LSD}$ as well.

In the black box model, we observe that the block sensitivity method, as well as the polynomial method for proving lower bounds for the number of queries, can both be extended to QMA protocols. We use these methods to obtain the following negative results:

1. In any $QMA(T, W)$ black box protocol for the function $NOR(X_1, ..., X_N)$, we have $T = \Omega(\sqrt{N})$. That is, the number of queries is $\Omega(\sqrt{N})$, regardless of the length of the proof supplied to the protocol. This shows that for some functions quantum proofs of any length do not help at all, and that any attempt to give short quantum proofs for the class of languages $Co - NP$ will have to go beyond black box arguments.

2. In any $QMA(T, W)$ black box protocol for a random boolean function $G(X_1, ..., X_N)$, we have $T + W = \Omega(N)$ (with high probability). That is, the total QMA black box complexity of a random function is (with high probability) $\Omega(N)$. This shows that for most functions short quantum proofs do not help much.

3. For any boolean function $G(X_1, ..., X_N)$, if there are $QMA(T, W)$ black box protocols for both G and $\neg G$, then there is a classical deterministic black box protocol for G that makes $O(T^6)$ queries to the black box. This shows that in the black box model $QMA \cap Co - QMA = P$.

On the positive side:

1. We observe that any (total or partial) boolean function $G(X_1, ..., X_N)$ has a $QMA(O(\sqrt{N}), N)$ black box protocol, that is, a protocol with proofs of length N that makes only $O(\sqrt{N})$ queries to the black box.

2. We observe a simple proof for the exponential separation between QMA black box complexity and (classical) MA black box complexity, for promise problems (first obtained by Watrous [31]). The gap is proved for the complement of Simon's problem. We show that this problem doesn't have short (classical) MA black box protocols, while it is well known that it does have very short quantum black box protocols.

1.4 Related Work

The notion of QMA was first defined and studied by Kitaev (see [14] for a recent survey). To the best of our knowledge, this notion was never studied before in the context of black box protocols or communication complexity protocols. Nevertheless, two works are very related to ours. The first work by Watrous [31] studies QMA protocols in the so called *black box group model*, which is a variant of the standard black box model studied here. Watrous gives an exponential separation between QMA protocols and MA protocols in that model. It is not hard to see that this result can be translated to the standard black box model [1] and hence one obtains an exponential separation between QMA protocols and MA protocols for the black box model. As mentioned above, one of our observations is a simpler proof for that result. The other related result is a recent work by Aaronson [1] that was done independently of ours. Aaronson studies the notion of *quantum certificate* in the (standard) black box model. This notion is very similar to the notion of QMA studied here, except that it completely ignores the length of the proofs supplied to the protocol. That is, a proof of any length is allowed for free. Nevertheless, some of our observations for the black box model are very related to Aaronson's results. In particular, versions of the above stated lower bound of $\Omega(\sqrt{N})$ for the NOR function and the above stated upper bound of $(O(\sqrt{N}), N)$ for any function are implicit in Aaronson's work.

Other notions of quantum proofs were also studied in several previous works. Among the most interesting results are the results of Watrous, and Watrous and Kitaev, that showed the power of *quantum interactive proofs* [32, 15].

1.5 Discussion

This research initiates a study of quantum proofs, or more precisely, Quantum Merlin-Arthur protocols, in the black box model and in the communication complexity model. A large number of open problems that we find extremely interesting follow from our work.

[1] This was communicated to us by Dorit Aharonov.

We prove here exponential separations between QMA protocols and MA protocols in both the black box model and the communication complexity model. One can also consider the (hybrid) class of quantum protocols with classical proofs ($QCMA$ protocols), which are the same as QMA protocols except that the proofs given to the protocols are restricted to be classical (see [3]). The ultimate power of quantum proofs (e.g., in the black box model and in the communication complexity model) can only be demonstrated by proving an exponential separation between QMA protocols and $QCMA$ protocols. That is, we are interested in examples for problems that, on one hand, have very efficient quantum protocols with quantum proofs, and, on the other hand, are very hard for quantum protocols with classical proofs. This would ultimately show that (in these models) quantum proofs are more powerful than classical ones. In the communication complexity model, we believe that the complete problems LSD and $\widetilde{LSD}$ may give such a separation. At this point, however, we still don't have any lower bound for the $QCMA$ communication complexity of these problems. In the black box model, we still don't have any good candidate for a problem that may give such a separation.

We prove here lower bounds for the QMA black box complexity of several functions. For some of these functions, however, our lower bounds are not tight. It would be very interesting to give tight lower bounds for well studied functions, such as $Parity(X_1, ..., X_N)$. (We believe that the right answer for $Parity$ is $\Theta(N)$). Moreover, the best lower bound that we have for the QMA black box complexity of any explicit function is $\Omega(\sqrt{N})$. Can one prove a lower bound of $\Omega(N)$?

In the communication complexity model, we still don't have a lower bound for the QMA communication complexity of any function. Can one prove such a lower bound ? Can one prove lower bounds for some of the well studied functions, such as *Inner Product* and *Set Disjointness* ? (We believe that the right answer for *Inner Product* is $\Theta(n)$ while the right answer for *Set Disjointness* is $\Theta(\sqrt{n})$).

Can one show that the QMA black box complexity of every total function is polynomially related to its (classical) MA black box complexity ? We show here that the classical deterministic black box complexity of $G(X_1, ..., X_N)$ is small when the QMA black box complexity of both G and $\neg G$ is small. Can one prove a similar result for communication complexity ?

Finally, can one demonstrate the power of quantum proofs in other interesting computational models ?

1.6 Organization of the Paper

In Section 2, we discuss the different models of communication complexity that we consider. In particular, we describe the classical model, the quantum model, the MA model and the QMA model. In Section 3, we analyze the complexity of the LSD problem in different models of communication complexity. All our results for communication complexity are proved in that section. In Section 4, we give the formal definitions and describe our results for the black box model. In this version of the paper, all these sections appear in the appendix, and some parts of the paper are still not in their final form.

Acknowledgment

We would like to thank Avi Wigderson and Dorit Aharonov for very important conversations.

References

[1] S. Aaronson. Quantum certificate complexity. quant-ph/0210020, 2002.

[2] D. Aharonov. Quantum computation- a review. In Dietrich Stauffer, editor, *Annual Review of Computational Physics, World Scientific*, volume VI. 1998. quant-ph/9812037.

[3] D. Aharonov and T. Naveh. Quantum np - a survey. quantph/0210077, 2002.

[4] A. Ambainis. A note on quantum black box complexity for almost all boolean functions. *Information Processing Letters*, 71(1):5–7, 1999. quant-ph/9811080.

[5] R. Beals, H. Buhrman, R. Cleve, M. Mosca, and R. de Wolf. Quantum lower bounds by polynomials. *Journal of the ACM*, 48(4):778–797, 2001. quant-ph/9802049.

[6] C. H. Bennett, E. Bernstein, G. Brassard, and U. Vazirani. Strengths and weaknesses of quantum computing. *SIAM Journal on Computing*, 26(5):1510–1523, 1997.

[7] E. Bernstein and U. Vazirani. Quantum complexity theory. *SIAM Journal on Computing*, 26(5):1411–1473, 1997.

[8] M. Boyer, G. Brassard, P. Høyer, and A. Tapp. Tight bounds on quantun searching. *Fortschritee der Physic*, 46(4-5):493–505, 1998. Earlier version appeared in Physcomp '96. quant-ph/9605034.

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

[10] H. Buhrman, R.Cleve, and A. Wigderson. Quantum vs. classical communication and computation. In *30'th STOC*, pages 63–68, 1998.

[11] H. Ehlich and K. Zeller. Schwankung von polynomen zwischen gitterpunkten. *Mathematische Zeitschrift*, 86:41–44, 1964.

[12] L. K. Grover. A fast quantum mechanical algorithm for database search. In *28'th STOC*, pages 212–219, 1996. quant-ph/9605043.

[13] J. Kempe and O. Regev. 3-local hamiltonian is qma-complete. quant-ph/0302079, 2003.

[14] A. Kitaev. Quantum np. Talk at Hebrew University, 1999.

[15] A. Kitaev and J. Watrous. Parallelization, amplification, and exponential time simulation of quantum interactive proof systems. In *Proceedings of the thirty second annual ACM Symposium on Theory of Computing: Portland, Oregon, May 21–23*, pages 608–617, 2000.

[16] A. Yu Kitaev, A. Shen, and M. N. Vyalyi. *Classical and Quantum Computation.* American Mathematical Society, 2002.

[17] I. Kremer. Quantum communication. Master's thesis, Hebrew University, 1995.

[18] E. Kushilevitz and N. Nisan. *Communication Complexity.* Cambridge University Press, 1996.

[19] M. Minsky and S. Papert. *Perceptrons.* MIT Press, Cambridge, MA, 2nd edition, 1988.

[20] I. Newman. Private vs. common random bits in communication complexity. *Information Processing Letters*, 39(2):67–71, 1991.

[21] M. A. Nielsen and I. L. Chuang. *Quantum Computation and Quantum Information.* Cambridge University Press, 2000.

[22] N. Nisan. Crew prams and decision trees. *SIAM Journal on Computing*, 20(6):999–1007, 1991.

[23] N. Nisan and M. Szegedy. On the degree of boolean functions as real polynomials. *Computational Complexity*, 4(4):301–313, 1994.

[24] J. Preskill. Lecture notes, 1997. http://www.theory.caltech.edu/people/preskill/ph229/.

[25] Ran Raz. Exponential separation of quantum and classical communication complexity. In *Proceedings of the thirty-first annual ACM symposium on Theory of computing*, pages 358–367. ACM Press, 1999.

[26] A. A. Razborov. Quantum communication complexity of symmetric predicates. *Izvestiya of the Russian Academy of Science, mathematics*, 6, 2002. To appear.

[27] T. J. Rivlin and E. W. Cheney. A comparison of uniform approximations on an interval and finite subset thereof. *SIAM Journal on numerical Analysis*, 3(2):311–320, 1966.

[28] P. W. Shor. Polynomial-time algorithm for prime factorization and discrete logarithms on a quantum computer. *SIAM Journal on Computing*, 26(5):1484–1509, 1997. quant-ph/9508027.

[29] D. Simon. On the power of quantum computation. *SIAM Journal on Computing*, 26(5):1474–1483, 1997.

[30] A. Ta-Shma. Classical versus quantum communication complexity. *SIGACTN: SIGACT News (ACM Special Interest Group on Automata and Computability Theory)*, 30, 1999.

[31] J. Watrous. Succinct quantum proofs for properties of finite groups. In *41st Annual Symposium on Foundations of Computer Science, 12–14 November, 2000, Redondo Beach, California*, pages 537–546.

[32] John Watrous. PSPACE has constant-round quantum interactive proof systems. In *IEEE Symposium on Foundations of Computer Science*, pages 112–119, 1999.

[33] R. de Wolf. Quantum communication and complexity. *TCS*, 287(1):337–353, 2002.

[34] A. C. Yao. Some complexity questions related to distributive computing. In *11th Annual ACM Symposium on Theory of Computing*, pages 209–213, 1979.

[35] A. C. Yao. Quantum circuit complexity. In *34th Annual Symposium on Foundations of Computer Science*, pages 352–361, 1993.

APPENDIX

2 Communication Complexity

In this section, we give the formal definitions and present our results for the communication complexity model.

As mentioned above, a communication complexity problem is given by 3 (usually finite) sets X, Y, Z and a function $f : X \times Y \to Z$. As before, we sometimes consider the case where f is a partial function (or promise problem). Player I is given an input $x \in X$. Player II is given an input $y \in Y$. The players are promised that f is defined on (x, y). Their goal is to compute $f(x, y)$. The communication complexity of the problem is the number of bits the two players have to exchange between them in order to compute $f(x, y)$.

As mentioned above, we would also like to consider cases where the sets X, Y are infinite. Therefore, in all that comes below, we do not assume that X, Y are finite. The definitions will hold for both cases.

In all the communication complexity problems discussed in this paper, the set Z of possible answers is $\{0, 1\}$, that is, there are only two possible answers. In all that comes below, let us hence assume for simplicity that $Z = \{0, 1\}$.

2.1 Classical Probabilistic Communication Complexity

In classical probabilistic communication complexity we allow the two players to use classical probabilistic protocols. In each step of the protocol, one of the players sends one bit of information (about his input) to the other player. In the end, both players have to know an answer z which is supposed to be $f(x, y)$.

For simplicity, let us assume that it is known in advance which player speaks in each step of the protocol (e.g., Player I sends the first bit and then they alternate). In each step of the protocol, the bit sent by a player may depend on the player's input and on all the messages already exchanged between the two players. The bit may also depend on a random string s, shared by both players.

The answer z is therefore a random variable (depending on the random string s). We require that for every input pair (x, y) the answer z obtained by the protocol satisfies

$$\Pr_s[z = f(x, y)] \geq 1 - err,$$

where $err > 0$ is some small constant (the probability of error). The exact value of the constant err is of less importance (as long as that value is less than $1/2$) and it may change the communication complexity of a problem by only a multiplicative constant. This is true, because the probability of error can be efficiently reduced by repetition.

The maximum number of bits sent by the players in such a protocol is called the communication complexity of the protocol (where the maximum is taken over all the possible inputs). The probabilistic communication complexity of a problem is the communication complexity of the best such protocol for that problem. We identify the problem with the (possibly partial) function f, and we denote the probabilistic communication complexity of f by $PCC(f)$ (for simplicity we remove the constant err from the notation, as it is of less importance).

Note that the two players share a random string s, i.e., they can both read s. That model is hence called the public-coins model. An alternative model is the private-coins model, where each of the two players has his own random string that cannot be read by the other player. Obviously, the private-coins model is weaker than the public-coins model. It is well known, however, that the private-coins model is only slightly weaker. More precisely, any problem that can be solved with communication complexity k by a public-coins protocol can also be solved with communication complexity $k+O(\log n)$ by a private-coins protocol (where n is the length of the inputs) [20]. Hence, the two models have roughly the same power.

2.2 Quantum Communication Complexity

In the quantum communication complexity model, the two players exchange between them quantum bits (qubits), rather than classical bits. For simplicity, let us assume that the protocol has k steps (rounds), and that in each step one of the players sends d qubits to the other player. We will assume for simplicity that Player I sends the d qubits in the first round and then they alternate.

Mathematically, we will use the following model: Let d, l be two integers. We think of l as the size of the quantum memory of each player, and we think of d as the size of a "quantum blackboard", shared by both players. The blackboard is used as a communication channel between the two players. The size of the memories is not limited, that is, l can be arbitrarily large.

We will have the following 3 sets:

$$L_1 = \{0,1\}^l \ \ , \ \ L_2 = \{0,1\}^l \ \ , \ \ D = \{0,1\}^d.$$

We think of L_1 as the set of classical assignments to Player I's memory, and we think of L_2 as the set of classical assignments to Player II's memory. We think of D as the set of classical assignments to the blackboard. The set $L_1 \times D \times L_2$ is hence the set of classical assignments to the entire system. Denote the elements of $L_1 \times D \times L_2$ by $e_0, ..., e_{2^{2l+d}-1}$. The element e_0, for example, will be the all 0 assignment to $L_1 \times D \times L_2$.

We will work with the 2^{2l+d} dimensional vector space

$$\Lambda = \mathbb{C}^{L_1 \times D \times L_2}.$$

We can define Λ also by

$$\Lambda = \mathbb{C}^{L_1} \otimes \mathbb{C}^{D} \otimes \mathbb{C}^{L_2},$$

where $\otimes$ denotes the tensor product. We think of Λ as the set of all pure quantum states of the system. Each element $e_i \in L_1 \times D \times L_2$ corresponds to a unit vector in Λ, with 1 in the i^{th} coordinate and 0 in all the other coordinates. Each such element e_i can hence be viewed also as a vector in Λ. The set $\{e_0, ..., e_{2^{2l+d}-1}\}$ is then the standard basis for Λ.

Let $\mathcal{U}$ be the set of all unitary operators on Λ. That is, the set of all linear operators $U : \Lambda \to \Lambda$, such that $UU^{\dagger} = Id$, (where Id is the identity operator). Each operator $U \in \mathcal{U}$ can be described by its action on the elements of the standard basis $\{e_0, ..., e_{2^{2l+d}-1}\}$.

Let $\mathcal{U}_1 \subset \mathcal{U}$ be the set of all such operators that act only on $\mathbb{C}^{L_1} \otimes \mathbb{C}^D$. Formally, we can define $\mathcal{U}_1$ by: $U \in \mathcal{U}_1$ iff

$$U = U' \otimes Id,$$

where U' is a unitary operator on $\mathbb{C}^{L_1} \otimes \mathbb{C}^D$ and Id is the identity operator on $\mathbb{C}^{L_2}$. We can hence think of $\mathcal{U}_1$ also as the set of unitary operators on $\mathbb{C}^{L_1 \times D}$. In the same way, let $\mathcal{U}_2 \subset \mathcal{U}$ be the set of all operators in $\mathcal{U}$ that act only on $\mathbb{C}^D \otimes \mathbb{C}^{L_2}$. We can think of $\mathcal{U}_2$ also as the set of unitary operators on $\mathbb{C}^{D \times L_2}$.

$\mathcal{U}_1$ will be the set of allowed operators for Player I. $\mathcal{U}_2$ will be the set of allowed operators for Player II. The players start from the initial vector $e_0 \in \Lambda$ (recall that e_0 corresponds to the all 0 assignment to $L_1 \times D \times L_2$). In each step of the protocol, one of the players apply a unitary operator from $\mathcal{U}_1$ or $\mathcal{U}_2$ respectively. In each step of the protocol, the operator used by a player may depend on the player's input (x or y respectively), but cannot depend on anything else. We will assume for simplicity that Player I applies the first operator and then they alternate.

Denote by U_1 the operator used by Player I in the first step and by U_2 the operator used by Player II in the second step, and so on (recall that U_1 is chosen as a function of x and U_2 is chosen as a function of y, and so on). The final state F is then defined by

$$F = U_k U_{k-1} \cdots U_2 U_1 e_0.$$

Note that since e_0 is a unit vector and since all operators are unitary, the final vector F is a unit vector as well.

The answer z is now determined by a measurement applied on F by one of the players. We assume for simplicity that the player that applies the measurement is always Player I.

Formally, the measurement is described by two orthogonal linear subspaces $M_0, M_1 \subset \Lambda$, of dimension 2^{2l+d-1} each. Since the measurement is applied by Player I, we require that the measurement is applied on $L_1 \times D$ only. Formally, this means that

$$M_0 = M'_0 \otimes \mathbb{C}^{L_2},$$

and

$$M_1 = M'_1 \otimes \mathbb{C}^{L_2},$$

where M'_0, M'_1 are two orthogonal subspaces of $\mathbb{C}^{L_1} \otimes \mathbb{C}^D$, of dimension 2^{l+d-1} each.

Thus, we can think of M_0, M_1 as subspaces of $\mathbb{C}^{L_1 \times D}$ and we can think of the measurement as applied on $L_1 \times D$.

Now denote by λ_0 the length of the projection of the final state F on M_0 and denote by λ_1 the length of the projection of F on M_1. The answer z, given by the protocol, is defined to be 0 with probability λ_0^2 and 1 with probability λ_1^2. Note, that since F is a unit vector, we have

$$\lambda_0^2 + \lambda_1^2 = 1.$$

The answer z is hence a random variable. As before, we require that for every input pair (x, y),

$$\Pr[z = f(x, y)] \geq 1 - err,$$

where $err > 0$ is some small constant. (As before, the exact value of the constant err is of less importance, as it can be reduced efficiently). The communication complexity of the protocol is defined to be $k \cdot d$. The number of steps k is also called the number of rounds in the protocol.

The quantum communication complexity of a problem is the communication complexity of the best such protocol for that problem. As before, we identify the problem with the function f, and we denote the quantum communication complexity of f by $QCC(f)$.

This defines the mathematical model of quantum communication complexity. Let us add two comments about that model:

First comment: As mentioned above, we do not limit the memory size l. It is not hard to see, however, that w.l.o.g. we can assume that l is at most $k \cdot d$. To show that Player I can use memory size of at most $k \cdot d$, we can use the following argument: Fix x (the input for Player I). Assume by induction that after i steps, a memory of size $i \cdot d$ is enough. If i is odd then Player II applies the next operator U_{i+1}, and hence the same memory size (for Player I) is enough after $i+1$ steps as well. If i is even then Player I applies the operator U_{i+1}. Obviously, we only care about the action of U_{i+1} on the set of assignments to the previous memory and to the set D. Since that set of assignments is of size $\leq 2^{(i+1)\cdot d}$, the image of U_{i+1} on that set of assignments is of dimension $\leq 2^{(i+1)\cdot d}$, and hence it can be condensed into a memory of size $(i+1) \cdot d$.

Second comment: The above definition uses complex numbers and require all operators to be unitary operators (over the complex numbers). However, we can assume w.l.o.g. that all numbers used are real numbers and that all the operators used are orthogonal operators over the real numbers. This was first observed by [7] for quantum computation in general (see also [2]) and is obviously true for quantum communication complexity as well. It can be proved simply by representing each complex number by two real numbers (one for the real part and one for the imaginary part). This requires adding only one qubit to the size of the blackboard.

2.3 MA Communication Complexity

We define MA communication complexity protocols in a similar manner to Definition 1. As before, we will have two parameters, T, W, and in order to make the discussion more general we will add another parameter, ϵ, which is a limit on the probability of error of the protocol (note that in Definition 1 the error ϵ was fixed to be the constant $1/3$). Here, T is a limit on the communication complexity of the protocol and W is a limit on the length of a proof w given to the players. We think of the proof w as presented to both players so both players can access the proof for free. One can also consider an alternative model, where the proof is only presented to Player I. Player I can then communicate the proof to the other player. Since the interesting cases are those that have relatively short proofs, this doesn't change the communication complexity by much.

The players proceed as before. In each step of the protocol, one of the players sends one bit of information to the other player. In the end, both players have to know an answer z. Thus, the answer z depends on the input (x, y) as well as on the proof w presented to the players. For a protocol P, denote by $P((x, y), w)$ the answer z given by the protocol on input (x, y) and proof w. Note that since the protocol P is probabilistic, the answer $P((x, y), w)$ is a random variable.

We can now define $MA_\epsilon(T, W)$ communication complexity protocols as follows:

Definition 3 *An $MA_\epsilon(T, W)$ communication complexity protocol for a function f is a classical probabilistic protocol P, as above (i.e., with an additional string w presented to the players), such that:*

1. *For any $(x, y) \in f^{-1}(1)$, there exists a string w, s.t.,*
$$\Pr[P((x,y),w) = 1] > 1 - \epsilon.$$
2. *For any $(x, y) \in f^{-1}(0)$ and any string w,*
$$\Pr[P((x,y),w) = 1] < \epsilon.$$
3. *The length of w is at most W, and P never communicates more than T bits (for any x, y, w, and any random string).*

We say, in this case, that f is in the class $MACC_\epsilon(T, W)$. The sum $T + W$ is sometimes referred to as the total MA complexity of the function f (for a constant error ϵ).

2.4 QMA Communication Complexity

We define QMA communication complexity protocols in a similar manner to Definition 2 and Definition 3. As before, we will have three parameters, T, W, and ϵ. As before, ϵ is a limit on the probability of error of the protocol. T is a limit on the communication complexity of the protocol and W is a limit on the length of a quantum proof $|w\rangle$ given to the players. We assume that only Player I can access the proof $|w\rangle$. This restriction makes sure that the players will not be able to use the proof as a communication channel. We think of the proof $|w\rangle$ as presented to Player I, so Player I can access the proof for free. For simplicity, we think of the proof $|w\rangle$ as the initial quantum state of some of the qubits in the memory of Player I.

Several alternative models can also be considered. For example, we can consider a model where each player gets a separate proof $|w\rangle$. Our results hold for all these models. Note also that typically we consider cases where the length of the proof is relatively small. Hence, even if we did allow the players to use the proof as a communication channel it wouldn't have changed the communication complexity by too much.

Thus, the players start from an initial vector $|e\rangle \in \Lambda$, such that $|e\rangle = |w\rangle \otimes |e'\rangle$, where $|w\rangle$ is the quantum proof given to Player I and $|e'\rangle$ is a vector that corresponds to the all 0 assignment in all the other qubits of the two players. The rest is as before. In each step of the protocol, one of the players apply a unitary operator from $\mathcal{U}_1$ or $\mathcal{U}_2$ respectively. In each step of the protocol, the operator used by a player may depend on the player's input (x or y respectively), but cannot depend on anything else. As before, we assume that Player I applies the first operator and then they alternate. The final state F is then defined by
$$F = U_k U_{k-1} \cdots U_2 U_1 |e\rangle,$$
where $U_k, U_{k-1}, \ldots, U_2, U_1$ are the operators applied by the players.

The answer z is now determined by a measurement applied on F by Player I. For a protocol P, denote by $P((x,y),|w\rangle)$ the answer given by the protocol on input (x, y) and proof $|w\rangle$. Note that the answer $P((x,y),|w\rangle)$ is a random variable.

We can now define $QMA_\epsilon(T,W)$ communication complexity protocols as follows:

Definition 4 *A $QMA_\epsilon(T,W)$ communication complexity protocol for a function f is a quantum protocol P, as above (i.e., with an additional quantum state $|w\rangle$ presented to Player I), such that:*

1. *For any $(x,y) \in f^{-1}(1)$, there exists a state $|w\rangle$, s.t.,*
$$\Pr[P((x,y),|w\rangle) = 1] > 1 - \epsilon.$$
2. *For any $(x,y) \in f^{-1}(0)$ and any state $|w\rangle$,*
$$\Pr[P((x,y),|w\rangle) = 1] < \epsilon.$$
3. *$|w\rangle$ is a quantum state of at most W qubits, and the communication complexity of P is at most T.*

We say, in this case, that f is in the class $QMACC_\epsilon(T,W)$. The sum $T+W$ is sometimes referred to as the total QMA complexity of the function f (for a constant error ϵ).

Comment: We saw before that we could assume w.l.o.g. that the memory size l is at most $k \cdot d$ (see a comment at the end of Subsection 2.2). Here, since the proof $|w\rangle$ is of length W, we can only assume that the memory size l is at most $k \cdot d + W$.

3 The Linear Space Distance (LSD) problem

In this section, we study the complexity of the LSD problem in several models of communication complexity. We will show the following.

1. LSD is a complete problem for QMA communication complexity.
2. LSD is hard for (classical) MA communication complexity protocols.
3. LSD is hard for (standard) quantum communication complexity protocols.
4. LSD is very easy for QMA communication complexity protocols.

In particular, we will show here that the QMA communication complexity model is exponentially stronger than the quantum communication model and the MA communication model. It is not hard to see that the problems LSD and $\widetilde{LSD}$ can be reduced to each other. Hence, all these results will hold for $\widetilde{LSD}$ as well. Following the second comment at the end of section 2.2, we work from now on over the real field, $\mathbb{R}$, instead of the complex field, $\mathbb{C}$.

3.1 Preliminaries

For a subspace $V \subset \mathbb{R}^m$ we denote with $S(V)$ the unit sphere in V
$$S(V) = \{ \ v \in V \mid \|v\| = 1 \ \}$$

where $\| \ \|$ is the Euclidean norm. For two subspaces $V_1, V_2 \subset \mathbb{R}^m$ we define the distance between them, $\Delta(V_1, V_2)$, as the distance between their unit spheres. In other words

$$\Delta(V_1, V_2) = \min_{v_1 \in S(V_1)} \min_{v_2 \in S(V_2)} \|v_1 - v_2\| \tag{1}$$

Notice that for any two subspaces V_1, V_2 we have that

$$0 \leq \Delta(V_1, V_2) \leq \sqrt{2}$$

and that

$$\Delta(V_1, V_2) = 0 \Leftrightarrow V_1 \cap V_2 \neq \{0\}$$
$$\Delta(V_1, V_2) = \sqrt{2} \Leftrightarrow V_1 \perp V_2$$

Thus, given two inputs V_1 and V_2, we have that $LSD(V_1, V_2)$ is 1 if the spaces are very close to each other (in a sense, almost intersecting), and 0 if the spaces are far (almost perpendicular).

For a vector v and a subspace V we denote with $\text{Proj}_V(v)$ the projection of v on V. If $\text{Proj}_V(v) \neq 0$ then we define

$$\widehat{\text{Proj}}_V(v) = \frac{\text{Proj}_V(v)}{\|\text{Proj}_V(v)\|}$$

Remember that $\widehat{\text{Proj}}_V(v)$ is the unit vector in V that is the closest to v (in case that $\text{Proj}_V(v) \neq 0$). Using the fact that

$$\|v - u\|^2 = \|v\|^2 + \|u\|^2 - 2\langle v, u\rangle$$

we get that for a unit vector v

$$(\Delta(v, V))^2 = 2 - 2\langle v, \widehat{\text{Proj}}_V(v)\rangle = 2 - 2\|\text{Proj}_V(v)\|$$

As a corollary we get

Lemma 5

$$\Delta(V_1, V_2) = \cdot \min_{v_1 \in S(V_1)} \left(\sqrt{2 - 2 \cdot \|\text{Proj}_{V_2}(v_1)\|}\right)$$

Notice that the lemma is true even if $\text{Proj}_V(v) = 0$. In the definition of LSD we require that the distance between the two space is either less than $0.1 \cdot \sqrt{2}$ or is larger than $0.9 \cdot \sqrt{2}$, however, this choice was arbitrary, and as the next lemma demonstrates, we can take any two numbers which are polynomially bounded and have a polynomial gap between them.

Lemma 6 *Given two families of subspaces in $\mathbb{R}^m$, $\{A_x\}_x$ and $\{B_y\}_y$, with the property that for every (x, y) either $\Delta(A_x, B_y) \leq \alpha$ or $\Delta(A_x, B_y) \geq \beta$, and $\beta - \alpha = \epsilon > 0$, we can construct a mapping $A_x \to \tilde{A}_x$ and $B_y \to \tilde{B}_y$ such that $\tilde{A}_x, \tilde{B}_y \subset \mathbb{R}^{m^{\text{poly}(\frac{1}{\epsilon}, \frac{1}{\alpha})}}$, and such that for every (x, y) either $\Delta(\tilde{A}_x, \tilde{B}_y) \leq 0.1 \cdot \sqrt{2}$ or $\Delta(A_x, B_y) \geq 0.9 \cdot \sqrt{2}$.*

This construction is similar in nature to error amplification. When we wish to reduce the error in a probabilistic process, we just repeat it several times and use the Chernoff inequality. We use the same intuition here.

Proof Denote

$$a = (1 - \frac{\alpha^2}{2})^2 \ , \ b = (1 - \frac{\beta^2}{2})^2$$

Using lemma 5 we see that for any (x, y)

$$\Delta(A_x, B_y) \geq \beta \ \Rightarrow \ \forall|\psi\rangle \in S(A_x) \quad \|\mathrm{Proj}_{B_y}(|\psi\rangle)\|^2 \leq b$$

$$\Delta(A_x, B_y) \leq \alpha \ \Rightarrow \ \exists|\psi\rangle \in S(A_x) \ \text{ such that } \ \|\mathrm{Proj}_{B_y}(|\psi\rangle)\|^2 \geq a$$

Let

$$\tilde{A}_x = A_x \otimes A_x \ldots \otimes A_x \subset \mathbb{R}^{m^k}$$

for some parameter k that we later determine. Denote $B_y^1 = B_y$ and $B_y^0 = B_y^{\perp}$. For any vector $v = (v_1, ..., v_k) \in \{0,1\}^k$ we define

$$B_y^v = B_y^{v_1} \otimes B_y^{v_2} \ldots \otimes B_y^{v_k} \subset \mathbb{R}^{m^k}$$

Notice that for every two different vectors $v, u \in \{0,1\}^k$ we have that

$$B_y^v \perp B_y^u$$

Let $\gamma = \frac{a+b}{2}$, define

$$\tilde{B}_y = \mathrm{span}\left\{ \ B_y^v \mid v \in \{0,1\}^k \ \text{ and } \ wt(v) > \gamma \cdot k \ \right\}$$

where $wt(v)$ is the number of nonzero coordinates of v. As $\tilde{B}_y$ is spanned by pairwise orthogonal spaces, we see that for any state $|\phi\rangle \in \mathbb{R}^{m^k}$ we have that

$$\|\mathrm{Proj}_{\tilde{B}_y}(|\phi\rangle)\|^2 = \sum_{\substack{v \in \{0,1\}^k \\ wt(v) > \gamma k}} \|\mathrm{Proj}_{B_y^v}(|\phi\rangle)\|^2$$

Case $\Delta(A_x, B_y) \leq \alpha$:

Note that if $|\phi\rangle = |\psi\rangle|\psi\rangle \ldots |\psi\rangle$, where $|\psi\rangle \in S(\mathbb{R}^m)$, then

$$\|\mathrm{Proj}_{B_y^v}(|\phi\rangle)\|^2 = \|\mathrm{Proj}_{B_y}(|\psi\rangle)\|^{2wt(v)} \cdot \|\mathrm{Proj}_{B_y^{\perp}}(|\psi\rangle)\|^{2(k-wt(v))}$$

Let $z_i \in \{0,1\}_{i=1}^k$ be i.i.d. random variables such that

$$\Pr[z_i = 1] = p \stackrel{\text{def}}{=} \|\mathrm{Proj}_{B_y}(|\psi\rangle)\|^2$$

We have that

$$\|\mathrm{Proj}_{\tilde{B}_y}(|\phi\rangle)\|^2 = \sum_{\substack{v \in \{0,1\}^k \\ wt(v) > \gamma k}} p^{wt(v)}(1-p)^{k-wt(v)} = \Pr[\sum_{i=1}^k z_i > \gamma k] \qquad (2)$$

Let $|\psi\rangle \in S(A_x)$ be such that

$$p \stackrel{\text{def}}{=} \|\mathrm{Proj}_{B_y}(|\psi\rangle)\|^2 \geq a$$

We thus get

$$|\phi\rangle \in S(\tilde{A}_X) \text{ and } \|\mathrm{Proj}_{\tilde{B}_y}(|\phi\rangle)\|^2 = \Pr[\sum_{i=1}^{k} z_i > \gamma k]$$

Notice that $\mu \stackrel{\mathrm{def}}{=} E(\sum_{i=1}^{k} z_i) = kp \geq ka$, so by the Chernoff inequality we get

$$\Pr[\sum_{i=1}^{k} z_i > \gamma k] \geq \Pr[\sum_{i=1}^{k} z_i > \mu(1 - \frac{a-b}{2a})] \geq 1 - exp(-\frac{1}{3}\cdot\mu(\frac{a-b}{2a})^2) \geq 1 - exp(-\frac{1}{3}\cdot ka(\frac{a-b}{2a})^2)$$

Thus, if $k = \mathrm{poly}(\frac{1}{\alpha}, \frac{1}{\epsilon})$ is large enough, than the right hand side is larger than, say, $1 - \frac{1}{1000}$. We now get by lemma 5 that

$$\Delta(\tilde{A}_x, \tilde{B}_y) \leq \sqrt{2 - 2\|\mathrm{Proj}_{\tilde{B}_y}(|\phi\rangle)\|} < 0.1 \cdot \sqrt{2}$$

Case $\Delta(A_x, B_y) \geq \beta$:

Let $|\phi\rangle \in S(\tilde{A}_x)$, and let

$$B_y^{(i)} = \mathbb{R}^m \otimes \ldots \otimes \mathbb{R}^m \otimes B_y \otimes \mathbb{R}^m \ldots \otimes \mathbb{R}^m$$

where there are k spaces in the product and B_y is the i'th one. As for any vector $|\psi\rangle \in S(A_x)$ we have that $\|\mathrm{Proj}_{B_y}(|\psi\rangle)\|^2 \leq b$, we get that the probability of getting a result in $B_y^{(i)}$ when measuring $|\phi\rangle$ w.r.t. $B_y^{(i)}$ and $(B_y^{(i)})^{\perp}$ is at most b. Also note that for $i \neq j$, first measuring a state w.r.t. $B_y^{(i)}$ (and its perpendicular complement), and then measuring w.r.t. $B_y^{(j)}$ (and its perpendicular complement), is the same as first measuring with respect to $B_y^{(j)}$ and then w.r.t. $B_y^{(i)}$. Thus, for any $v \in \{0,1\}^k$ we have that

$$\|\mathrm{Proj}_{B_y^v}(|\phi\rangle)\|^2 \leq b^{wt(v)}(1-b)^{k-wt(v)}$$

Therefor, as in equation 2, we get that

$$\|\mathrm{Proj}_{\tilde{B}_y}(|\phi\rangle)\|^2 \leq \sum_{\substack{v \in \{0,1\}^k \\ wt(v) > \gamma k}} b^{wt(v)}(1-b)^{k-wt(v)}$$

As $b < \frac{a+b}{2} = \gamma$ we get by the Chernoff inequality that

$$\|\mathrm{Proj}_{\tilde{B}_y}(|\phi\rangle)\|^2 \leq exp(-\frac{1}{3}kb(\frac{a-b}{2b})^2)$$

Therefor, if $k = \mathrm{poly}(\frac{1}{\alpha}, \frac{1}{\epsilon})$ is large enough, than the right hand side is smaller than, say, $\frac{1}{1000}$. We now get by lemma 5 that

$$\Delta(\tilde{A}_x, \tilde{B}_y) = \min_{|\phi\rangle \in S(\tilde{A}_x)} \left(\sqrt{2 - 2\|\mathrm{Proj}_{\tilde{B}_y}(|\phi\rangle)\|}\right) > 0.9 \cdot \sqrt{2}$$

$\square$

3.2 The Completeness of LSD

As mentioned above, LSD is a complete problem for QMA communication complexity protocols. More precisely, any (partial or total) communication complexity problem f that has a $QMA_\epsilon(T,W)$ communication complexity protocol (for constant $\epsilon < 1/2$) can be reduced to the LSD problem with $\log m = \text{poly}(T+W)$. The theorem is formally stated below, and its proof is given in Subsection 3.6.

Theorem 7 *If $f : X \times Y \to \{0,1\}$ has a $QMA(T,W)$ protocol, then there is a mapping from X and Y to subspaces of $\mathbb{R}^{2^{\text{poly}(T,W)}}$, $x \to A_x$, $y \to B_y$, such that*

$$f(x,y) = 1 \;\Rightarrow\; \Delta(A_x, B_y) \le 0.1 \cdot \sqrt{2}$$

$$f(x,y) = 0 \;\Rightarrow\; \Delta(A_x, B_y) \ge 0.9 \cdot \sqrt{2}$$

3.3 Lower Bounds for the MA Communication Complexity of LSD

In this subsection, we prove our lower bound for the (classical) MA communication complexity of the LSD problem. The lower bound will follow by the completeness of the LSD problem for QMA communication complexity protocols, and by proving a lower bound for the MA communication complexity of a different problem, the problem $\mathcal{P}(\vartheta_0, \vartheta_1)$. The problem $\mathcal{P}(\vartheta_0, \vartheta_1)$ has two parameters, positive constants $\vartheta_0, \vartheta_1 \in \mathbb{R}$, such that $\vartheta_0^2 + \vartheta_1^2 < 1$.

The problem $\mathcal{P}(\vartheta_0, \vartheta_1)$:
Player I gets as input a unit vector $v \in \mathbb{R}^n$, and two orthogonal vector-spaces $M_0, M_1 \subset \mathbb{R}^n$ of dimension $n/2$ each (we assume for simplicity that n is even). Player II gets as input an orthogonal matrix $U : \mathbb{R}^n \to \mathbb{R}^n$, (i.e., a matrix U, such that $UU^\dagger = Id$, where Id is the identity matrix). Their goal is to answer 0 if $U(v)$ is of distance $\le \vartheta_0$ from M_0 and 1 if $U(v)$ is of distance $\le \vartheta_1$ from M_1, (and any answer in any other case).

The problem $\mathcal{P}(\vartheta_0, \vartheta_1)$, for $\vartheta_0 = \vartheta_1$, was introduced in [25] as a complete problem for two rounds quantum communication complexity. It was proved in [25] that for any $0 < \vartheta < 1/\sqrt{2}$, the quantum communication complexity of $\mathcal{P}(\vartheta, \vartheta)$ is $O(\log n)$, while its classical probabilistic communication complexity is $\Omega(\sqrt{n})$. This gave an exponential separation between quantum and classical communication complexity. More precisely, the following two theorems were proved.

Theorem 8 *[25] For any $0 \le \vartheta < 1/\sqrt{2}$, the problem $\mathcal{P}(\vartheta, \vartheta)$ can be solved by a quantum communication complexity protocol with complexity $O(\log n)$ and with only two rounds of communication (one for each player).*

Theorem 9 *[25] For any $0 < \vartheta < 1/\sqrt{2}$, $PCC(\mathcal{P}(\vartheta, \vartheta)) = \Omega(\sqrt{n})$.*

Note however that the proof for Theorem 9 (given in [25]) doesn't give a lower bound for the MA communication complexity of the problem. Here, we extend the proof of Theorem 9 to the case of MA communication protocols, but we are only able to prove it for a different range of the parameters ϑ_0, and ϑ_1 (the proof doesn't work for $\vartheta_0 = \vartheta_1$).

Theorem 10 *Let $\vartheta_0, \vartheta_1 \in \mathbb{R}$ be two constants, such that $\vartheta_0 > 0$, and $\vartheta_1 \ge 1/\sqrt{2}$, and $\vartheta_0^2 + \vartheta_1^2 < 1$. Let ϵ be a constant, such that $\epsilon < 1/2$. In any $MA_\epsilon(T,W)$ communication complexity protocol for $\mathcal{P}(\vartheta_0, \vartheta_1)$, we have $T \cdot W = \Omega(\sqrt{n})$. (Hence, we also have $T + W = \Omega(n^{1/4})$.*

Proof Fix n. Denote by X the set of inputs for Player I and by Y the set of inputs for Player II. Denote $Z = X \times Y$. The sets X, Y, Z are compact manifolds (and each of them is a compact metric space with a transitive group of isometries). On each of them we have a standard notion of *the uniform measure*, which is just Haar's measure. We assume that the uniform measure is normalized to be a probability measure.

Assume that there is an $MA_\epsilon(T, W)$ communication complexity protocol for $\mathcal{P}(\vartheta_0, \vartheta_1)$. By repeating the probabilistic part of the protocol $O(W)$ times, we can reduce the error probability to be smaller than $2^{-c \cdot W}$ (for an arbitrary large constant c). Hence, there exists an $MA_{\epsilon'}(k, W)$ protocol for $\mathcal{P}(\vartheta_0, \vartheta_1)$, with $\epsilon' \leq 2^{-c \cdot W}$, and $k = O(T \cdot W)$. Let $Prot$ be such a protocol. Assume for a contradiction that $k = o(\sqrt{n})$. We will get a contradiction by showing that ϵ', the probability of error of $Prot$, is larger than $2^{-c \cdot W}$.

If we fix an assignment s to the random string of $Prot$ and an assignment w to the proof supplied to the players, $Prot$ becomes a deterministic communication complexity protocol. For any input pair $(x, y) \in Z$ and any assignment s to the random string of $Prot$ and any assignment w to the proof supplied to the players, the string of communication bits exchanged by the two players on the inputs (x, y), using the random string s and the proof w, is called the history of (x, y, s, w). For any $h \in \{0, 1\}^k$ and any assignment s to the random string of $Prot$ and any assignment w to the proof supplied to the players, we denote by $Z_{s,w,h} \subset Z$ the set of all input pairs $(x, y) \in Z$ such that the history of (x, y, s, w) is h.

It is well known (and easy to show) that for any s, w, h, the set $Z_{s,w,h}$ is a product set, that is

$$Z_{s,w,h} = X_{s,w,h} \times Y_{s,w,h},$$

where $X_{s,w,h} \subset X$ and $Y_{s,w,h} \subset Y$. Also, it is well known (and easy to show) that for any fixed s, w, the family $\{Z_{s,w,h}\}_{h \in \{0,1\}^k}$ is a partition of Z. That is,

1. For any s, w and any $h \neq h'$,
$$Z_{s,w,h} \cap Z_{s,w,h'} = \emptyset.$$

2. For any s, w,
$$\bigcup_h Z_{s,w,h} = Z.$$

The answer given by the protocol $Prot$ on (x, y, s, w) depends only on s, w, h. That is, for all input pairs in $Z_{s,w,h}$ the answer given by the protocol $Prot$ on (x, y, s, w) will be the same. Let us denote that answer by $Prot(s, w, h)$.

In general, $X_{s,w,h} \subset X$ and $Y_{s,w,h} \subset Y$ can be arbitrary sets. In particular, they are not necessarily measurable. For our analysis, we will need to assume that for any s, w, h, the sets $X_{s,w,h}, Y_{s,w,h}$ are Borel sets. A claim similar to the following one was proved in [25].

Claim 11 *W.l.o.g., we can assume that for any s, w, h, the sets $X_{s,w,h}, Y_{s,w,h}$ are Borel sets.*

Define $H_0 \subset Z$ to be the set of all input pairs $((v, M_0, M_1), U) \in Z$ such that $U(v) \in M_0$. Define $H_1 \subset Z$ to be the set of all input pairs $((v, M_0, M_1), U) \in Z$ such that $U(v)$ is of distance $\leq \vartheta_1$ from M_1. Note that H_0 is a set of measure 0 (in Z), while the measure of the set H_1 is at least $1/2$ (by the condition on ϑ_1 and by Pythagoras theorem and by an argument of symmetry).

For each of the sets H_0, H_1, we have a standard notion of *uniform measure*. (The uniform measures on H_0, H_1 are just the ones induced on H_0, H_1 from the uniform measure on Z). As before, we assume that the uniform measure is normalized to be a probability measure. Note that H_0 is a set of zeros of the problem, so for this set $Prot$ has to answer 0 with high probability for every possible w, and H_1 is a set of ones of the problem, so for this set $Prot$ has to answer 1 with high probability for at least one w.

For a Borel set $Z' \subset Z$, we denote by $\alpha(Z')$ the measure of Z' in Z. We denote by $\beta_0(Z')$ the measure of $Z' \cap H_0$ in H_0, and we denote by $\beta_1(Z')$ the measure of $Z' \cap H_1$ in H_1.

Theorem 10 will follow from the following lemma, proved in [25].

Lemma 12 *There exists a universal constant $\delta' > 0$, s.t., for any two Borel sets $X' \subset X$, $Y' \subset Y$,*

$$\beta_0(X' \times Y') \ \geq \delta' \cdot \alpha(X' \times Y') - O(2^{-\sqrt{n}}).$$

Note that since H_1 is a set of probability at least $1/2$ in Z, for every $Z' \subset Z$ we have $\beta_1(Z') \leq 2\alpha(Z')$. Therefore, we also have the following lemma.

Lemma 13 *There exists a universal constant $\delta > 0$, s.t., for any two Borel sets $X' \subset X$, $Y' \subset Y$,*

$$\beta_0(X' \times Y') \ \geq \delta \cdot \beta_1(X' \times Y') - O(2^{-\sqrt{n}}).$$

For any s, w, denote by $A_0(s, w) \subset Z$ the union of all sets $Z_{s,w,h}$, s.t., $Prot(s, w, h) = 0$ (i.e., the answer of the protocol is 0). For any s, w, denote by $A_1(s, w) \subset Z$ the union of all sets $Z_{s,w,h}$, s.t., $Prot(s, w, h) = 1$ (i.e., the answer of the protocol is 1). Then, for any s, w, the sets $A_0(s, w)$ and $A_1(s, w)$ are disjoint, and their union is Z.

Since each of $A_0(s, w), A_1(s, w)$ is a union of at most 2^k of the sets $X_{s,w,h} \times Y_{s,w,h}$, we have by Lemma 13 for any s, w,

$$\beta_0(A_1(s, w)) \geq \delta \cdot \beta_1(A_1(s, w)) - O(2^k \cdot 2^{-\sqrt{n}}).$$

Hence,

$$\beta_0(A_1(s, w)) \geq \delta \cdot \beta_1(A_1(s, w)) - o(2^{-W}).$$

Note that $\beta_1(A_1(s, w))$ is the fraction of inputs in H_1, s.t., the answer of $Prot$ on (x, y, s, w) is 1. Recall that H_1 is a set of ones of the problem. Hence, for every input (x, y) in H_1 there is at least one w such that with probability at least $(1 - \epsilon')$ (over the random string s) the input (x, y) is in $A_1(s, w)$. Since the number of possible strings w is at most 2^W, there is (at least one) w that corresponds to at least 2^{-W} fraction of inputs in H_1. Fix w to be that w. Then, for almost every s, we have that $\beta_1(A_1(s, w))$ is of size almost 2^{-W}. Formally, with probability at least $1/2$ (over the random string s), we have

$$\beta_1(A_1(s, w)) > 2^{-(W+1)}.$$

Hence (for that fixed w), with probability at least $1/2$ (over the random string s),

$$\beta_0(A_1(s, w)) \geq (\delta/2) \cdot 2^{-W} - o(2^{-W}) \geq (\delta/4) \cdot 2^{-W}.$$

But $\beta_0(A_1(s, w))$ is the fraction of inputs in H_0, s.t., the answer of $Prot$ on (x, y, s, w) is 1. Thus (for that fixed w), the probability, over all $(x, y) \in H_0$ and over the random string s, to get an answer 1 is at least $(\delta/8) \cdot 2^{-W}$. Recall that H_0 is a set of zeros of the problem, so for every $(x, y) \in H_0$ (and for every w) the protocol is supposed to answer 1 with probability at most ϵ'. Hence,

$$\epsilon' \geq (\delta/8) \cdot 2^{-W},$$

which is a contradiction. □

Theorem 14 *For any $\vartheta_0, \vartheta_1 > 0$, such that $\vartheta_0^2 + \vartheta_1^2 < 1$, the problem $\mathcal{P}(\vartheta_0, \vartheta_1)$ can be solved by a quantum communication complexity protocol with complexity $O(\log n)$ and with only two rounds of communication (one for each player).*

Proof The proof is similar to the proof of Theorem 8 (given in [25]). In short, Player I sends v as a quantum state of $\log n$ qubits. Player II applies the transformation U on v and sends back $U(v)$. Player I then measures according to M_0 and M_1. This is repeated (in parallel) a constant number of times, and the Players decide according to the number of times that $U(v)$ was measured to be in M_0 and the number of times that it was measured to be in M_1. □

We can now state our lower bound for the MA communication complexity of the LSD problem.

Theorem 15 *In any $MA_\epsilon(T, W)$ communication complexity protocol for the LSD problem, we have $T + W = \Omega(m^\delta)$, for some constant $\delta > 0$.*

Proof Immediate from Theorem 10 and from the completeness of the LSD problem for QMA communication complexity protocols. □

3.4 Fast QMA Protocols for LSD

We will now show a fast QMA protocol for the LSD problem.

Theorem 16 *There exists a two round QMA communication protocol of complexity $O(\log m)$ that solves the LSD problem.*

The idea of the proof is the following. If the spaces given as inputs are close then the prover gives as a proof a vector in V_1 (the space of player I) that is close to V_2. Player I verifies that this vector is in her space and then player II measures the vector with respect to his space.

Proof Let $V_1 \subset \mathbb{R}^m$ be the input of Player I and $V_2 \subset \mathbb{R}^m$ be the input of Player II. Consider the following communication protocol. The prover give as proof a unit vector $v \in V_1$ that achieves the minimum in equation 1. Note the proof is given as a super position of $\lceil \log m \rceil = \Theta(\log n)$ qubits. Player I now measures v with respect to V_1 and ${V_1}^\perp$. If v was projected to ${V_1}^\perp$ then Player I outputs 0. Otherwise she sends the projected vector to Player II. Player II now measures this vector with respect to V_2 and ${V_2}^\perp$. If the vector was projected to V_2 then he outputs 1, otherwise he outputs 0. Let us analyze the protocol. Assume that $\Delta(V_1, V_2) \leq 0.1 \cdot \sqrt{2}$. Let $v_1 \in S(V_1)$ and $v_2 \in S(V_2)$ be two vectors that achieve the minimum in equation 1. We have that

$$\langle v_1, v_2 \rangle = 1 - \frac{1}{2}\|v_1 - v_2\|_2^2 \geq 1 - 0.01 > 0.95$$

According to the algorithm, the prover gives as a proof the vector v_1. Since $v_1 \in V_1$ player I never rejects, and the outcome of his measurement is always v_1. Now player I sends v_1 to player II. Since $\langle v_1, v_2 \rangle > 0.95$ we see that the length of the projection of v_1 on V_2 is at least 0.95. Therefor with probability at least $0.95^2 > 0.9$ the protocol outputs the correct answer.

On the other hand if $\Delta(V_1, V_2) \geq 0.9 \cdot \sqrt{2}$ then for any two unit vectors $v_1 \in S(V_1)$ and $v_2 \in S(V_2)$ we have that

$$\langle v_1, v_2 \rangle = 1 - \frac{1}{2}\|v_1 - v_2\|_2^2 \leq 1 - 0.9^2 < 0.2$$

thus if player I didn't reject at the first stage, then the outcome of the measurement of player II will be 1 with probability at most 0.2^2 (as the probability of getting 1 is given by the square of the length of the projection of some unit vector from V_1 on V_2). Notice that the number of qubits that each of the players used is at most $\log m$.

We are almost done now. The only problem is that in the definition of QMA protocols we allow measurements only at the final step of the protocol. However a simple argument shows that this is not a real problem. Given a decomposition $\mathbb{R}^m = V \perp V^\perp$, consider the following unitary operation

$$U(|\Psi\rangle|a\rangle) = |\Psi_1\rangle|a \oplus 1\rangle + |\Psi_2\rangle|a\rangle$$

where $|\Psi_1\rangle$ and $|\Psi_2\rangle$ are the projections of $|\Psi\rangle$ on V and $V^\perp$ respectively. Thus, instead of measuring $|\Psi\rangle$ with respect to V and $V^\perp$ we decompose it to its projections. We need to add one more qubit to our memory in this process. Notice that if we apply this to $|\Psi\rangle|0\rangle$ and then measure the qubit in the last register (the register we denoted with $|a\rangle$) then the probability of getting 1 is equal to the probability that we get 1 when measuring $|\Psi\rangle$ with respect to V and $V^\perp$. Thus, given a protocol which uses measurements, we can replace any of the measurement performed during the protocol (except the last one) with such a transformation, and get a new protocol. At the end of the new protocol we measure all the qubits that we added. This shows that by adding to the memory a small number of qubits (the exact number is equal to the number of measurements performed in the protocol) we can postpone all the measurements to the last step of the protocol. $\square$

3.5 Lower Bounds for the Quantum Communication Complexity of LSD

We will now show a lower bound for the quantum communication complexity of the LSD problem. For simplicity, we state and prove the theorem for $\widetilde{LSD}$. Since, as mentioned before, it is not hard to see that these two problems are equivalent, the same bounds holds for LSD as well.

Theorem 17 *Any quantum communication protocol for $\widetilde{LSD}$ must have communication of size at least $\Omega(\sqrt{m})$.*

The proof of this theorem is by a reduction to the Intersection problem (the complement of the Disjointness). The inputs to the Intersection problem are two subsets $S_1 \subset [m]$ and $S_2 \subset [m]$. We have to output 1 if $S_1 \cap S_2 \neq \emptyset$, and 0 otherwise. Recently Razborov [26] managed to prove that any quantum communication protocol for Intersection must communicate at least $\Omega(\sqrt{m})$ qubits. We show how to translate any pair of inputs to the Intersection problem (that is two subsets of $[m]$), to a pair of inputs to the $\widetilde{LSD}$ problem (that is two linear subspaces of $\mathbb{R}^m$, given by a finite precision) such that if the original inputs represent intersecting sets then their corresponding subspaces are close (they are actually intersecting), and if the sets are disjoint, then the corresponding subspaces are far (they are actually perpendicular).

Proof Let $S_1, S_2 \subset [m]$ be the inputs to the Intersection problem. Let $e_i \in \mathbb{R}^m$ be the vector that all its coordinates are zero except for the i'th coordinate which is 1. Player I prepares the subspace

$$V_1 = \mathrm{span}(e_i \mid i \in S_1) .$$

Player II prepares the subspace

$$V_2 = \mathrm{span}(e_i \mid i \in S_2) .$$

We clearly have that if $S_1 \cap S_2 = \emptyset$ then $V_1 \perp V_2$, and if $S_1 \cap S_2 \neq \emptyset$ then there is a non zero vector in $V_1 \cap V_2$. It is also easy to see that V_1 and V_2 are valid inputs to $\widetilde{LSD}$. Therefor, any quantum communication protocol for $\widetilde{LSD}$ solves Intersection, with the same communication complexity. Hence according to Razborov's result any quantum communication protocol for $\widetilde{LSD}$ has complexity at least $\Omega(\sqrt{m})$ □

3.6 Proof of theorem 7

We now show that LSD is a complete problem for QMA communication protocols. We begin with a lemma showing that we can reduce the error of any QMA protocol .

Lemma 18 *If f has a $Q_\epsilon(T, W)$ communication protocol, then it also has a $QMA_{\epsilon^l}(O(l \cdot T), O(l \cdot W))$ communication protocol, with the same number of rounds.*

The idea of the proof is standard - the prover will give $O(l)$ copies of his proof, and the players will run l independent copies of the protocol, in parallel, and will output the majority result of the protocols. The only problem is to make sure that there is no way that an adversary could convince us to accept a wrong input, by giving an entangled state as proof, rather than a tensor product of l copies of the same proof. However this is a common problem when trying to reduce the error in quantum algorithms and it was handled before (see [14, 15, 3]), so we omit the details of the proof in this version of the paper.

We now move to the second stage, which is the technical part of the proof, where we show how to reduce a QMA protocol with a relatively small error to a variant of LSD where the promise is that either the distance between A_x and B_y is very small, or the distance between them is not so small (as in lemma 6). Then using lemma 6 we complete the proof.

Lemma 19 *If f has a $QMA(T, W)$ protocol that has R rounds and whose error is at moat R^{-4}, then Player I can compute for any input x, a subspace $A_x \subset \mathbb{R}^{2^{\mathrm{poly}(T,W)}}$, and Player II can compute for any input y, a subspace $B_y \subset \mathbb{R}^{2^{\mathrm{poly}(T,W)}}$, such that*

$$f(x, y) = 1 \quad \Rightarrow \quad \Delta(A_x, B_y) \leq \frac{\sqrt{2}}{R^{2.5}}$$

$$f(x, y) = 0 \quad \Rightarrow \quad \Delta(A_x, B_y) \geq \frac{1}{R^{1.5}}$$

(we assume that w.l.o.g. R is large enough, say $R > 100$).

We would like to construct the spaces A_x and B_y in such a way that if they are close, then the unit vectors achieving the minimal distance actually give us a strong evidence that the protocol accepts (x, y). Intuitively if the output of the protocol, given the input (x, y) and the proof $|\psi\rangle$, is

$$U_R U_{R-1} \dots U_1(|\psi\rangle|0\rangle)$$

then we would like the following vector to be close to both A_x and B_y:

$$|0\rangle \otimes |\psi\rangle|0\rangle + \sum_{i=1}^{R} |i\rangle \otimes U_i \dots U_1(|\psi\rangle|0\rangle)$$

Notice that in that vector, the first system serves as a "clock" that follows the protocol for f, and the second system is the state of the protocol after the i'th step. In constructing these spaces we have to remember that Player I only knows $U_1, U_3, ..., U_{2i-1}, ..., U_{R-1}$ (assume that R is even), and that Player II has access only to $U_2, U_4, ..., U_{2i}, ...U_R$. Therefor we will define A_x to be the space of all protocols that are consistent with Player I's strategy, on input x, and that end in an accepting state. Similarly, B_y will be the space of all protocols that are consistent with Player II's strategy, on input y. If $f(x, y) = 1$ then there is a protocol that ends in an almost accepting state, and that is consistent with both players strategies. On the other hand, if $f(x, y) = 0$ then there is no protocol that is consistent with both strategies and that ends in (or close to) an accepting state.

Proof of Lemma 19 Let f be computed by a QMA protocol, $\mathcal{P}$, whose error is bounded by $\frac{1}{R^4}$, as in the assumption of the lemma. We also assume w.l.o.g. that R is even. Let S be the number of qubits used by the players in the protocol. Notice that according to the comment at the end of section 2.4 we have that $S \leq 2(T + W)$. Let $U_1^{(x)}, U_3^{(x)}, \dots, U_{R-1}^{(x)}$, $U_2^{(y)}, \dots U_R^{(y)}$ be the unitary transformations applied by Player I and Player II, respectively, in $\mathcal{P}$ on input (x, y). To simplify the notation we drop the superscripts (x) and (y), and remember that U_i, for odd i, was applied by Player I, and U_i, for even i, was applied by Player II. Thus, if $|\psi\rangle$ is the proof given by the prover, then the final state of $\mathcal{P}$, before the measurement, is

$$U_R U_{R-1} \dots U_2 U_1(|\psi\rangle|0\rangle)$$

Let

$$\mathcal{H} = \mathbb{R}^{R+1} \otimes \mathbb{R}^{2^S}$$

We denote the basis vectors of $\mathbb{R}^{R+1}$ with $|i\rangle$ for $i = 0, \dots, R$. For simplicity we assume that when we have $|i\rangle|\phi\rangle$ then $|i\rangle \in \mathbb{R}^{R+1}$ and $|\phi\rangle \in \mathbb{R}^{2^S}$. Given an input (x, y) of f, we show how to define two spaces A_x and B_y, such that

$$A_x,\ B_y\ \subset \mathcal{H}$$

$$f(x, y) = 1 \ \Rightarrow\ \Delta(A_x, B_y) \leq \frac{\sqrt{2}}{R^{2.5}}$$

$$f(x, y) = 0 \ \Rightarrow\ \Delta(A_x, B_y) \geq \frac{1}{R^{1.5}}$$

Let $V_{init} \subset \mathbb{R}^{2^S}$ be the space of all possible initial states of $\mathcal{P}$. That is

$$V_{init} = \{\ |\Psi\rangle|0\rangle \in \mathbb{R}^{2^S}\ \text{ such that }\ |\Psi\rangle \in \mathbb{R}^{2^W}\ \}$$

Let $V_{acc} \subset \mathbb{R}^{2^S}$ be the space of all accepting states of $\mathcal{P}$. That is, the output of $\mathcal{P}$ is determined by the result of the measurement of its final state w.r.t. V_{acc} and $V_{acc}^{\perp}$. We now define A_x and B_y.

$$A_x = \left\{ \sum_{i=0}^{\frac{R}{2}-1} (|2i+1\rangle U_{2i+1}|\phi_i\rangle + |2i\rangle|\phi_i\rangle) + |R\rangle|\phi_R\rangle \quad \text{such that} \quad |\phi_0\rangle \in V_{init} \quad \text{and} \quad |\phi_R\rangle \in V_{acc} \right\}$$

$$B_y = \left\{ |0\rangle|\phi_0\rangle + \sum_{i=1}^{\frac{R}{2}} (|2i\rangle U_{2i}|\phi_i\rangle + |2i-1\rangle|\phi_i\rangle) \quad \text{such that} \quad |\phi_0\rangle \in V_{init} \right\}$$

It is clear that A_x, B_y are linear subspaces of $\mathcal{H}$.

Case $f(x,y) = 1$:

Let $|\psi\rangle$ be the proof given by the prover in the protocol $\mathcal{P}$ that makes the protocol accept (x,y) with high probability. Let

$$|\rho_0\rangle = |\psi\rangle|0\rangle \in \mathbb{R}^{2^S} \quad , \quad \forall 1 \le i \le R \;\; |\rho_i\rangle = U_i \cdot U_{i-1} \cdot \ldots \cdot U_1 |\rho_0\rangle$$

Notice that $|\rho_0\rangle$ is the initial state of $\mathcal{P}$, that $\forall 0 \le i \le R \;\; \||\rho_i\|| = 1$, and that $|\rho_R\rangle$ is the final state of $\mathcal{P}$ when given (x,y) as inputs and $|\psi\rangle$ as a proof. Let

$$|\rho\rangle = \mathrm{Proj}_{V_{acc}}(|\rho_R\rangle)$$

As $f(x,y) = 1$ we have that

$$\langle \rho | \rho_R \rangle = \Pr[\; \mathcal{P} \;\text{ accepts }\; |\rho_0\rangle \;] \ge 1 - \frac{1}{R^4} \tag{3}$$

Clearly $|\rho\rangle \neq 0$, so we can define

$$|\hat{\rho}\rangle = \frac{1}{\||\rho\rangle\|} \cdot |\rho\rangle$$

Let

$$|\Phi_x\rangle = \frac{1}{\sqrt{R+1}} \left(\sum_{i=0}^{R-1} |i\rangle|\rho_i\rangle \right) + \frac{1}{\sqrt{R+1}} |R\rangle|\hat{\rho}\rangle$$

Clearly $|\Phi_x\rangle$ is in $S(A_x)$. Let

$$|\Phi_y\rangle = \frac{1}{\sqrt{R+1}} \left(\sum_{i=0}^{R} |i\rangle|\rho_i\rangle \right)$$

Clearly $|\Phi_y\rangle$ is in $S(B_y)$. We have that

$$\Delta(A_x, B_y) \le \||\Phi_x\rangle - |\Phi_y\rangle\| = \frac{1}{\sqrt{R+1}} \||\hat{\rho}\rangle - |\rho_R\rangle\| \le \frac{\sqrt{2}}{R^{2.5}}$$

where the last inequality follows from equation 3, and from the fact that $\langle \rho | \rho_R \rangle = \langle \hat{\rho} | \rho_R \rangle^2$.

case $f(x, y) = 0$**:**

Let $|\Phi_x\rangle \in S(A_x)$, $|\Psi_y\rangle \in S(B_y)$ be such that

$$\Delta(A_x, B_y) = \||\Phi_x\rangle - |\Psi_y\rangle\|$$

(from compactness argument we know that such $|\Phi_x\rangle$ and $|\Psi_y\rangle$ exist). Let

$$|\Phi_x\rangle = \sum_{i=0}^{\frac{R}{2}-1} (|2i+1\rangle U_{2i+1}|\phi_i\rangle + |2i\rangle|\phi_i\rangle) + |R\rangle|\phi_R\rangle \quad \text{and}$$

$$|\Psi_y\rangle = |0\rangle|\psi_0\rangle + \sum_{i=1}^{\frac{R}{2}} (|2i\rangle U_{2i}|\psi_i\rangle + |2i-1\rangle|\psi_i\rangle)$$

where $|\phi_0\rangle, |\psi_0\rangle \in V_{init}$ and $|\phi_R\rangle \in V_{acc}$. We now show that if $\Delta(A_x, B_y)$ is small, then $\mathcal{P}$ accepts $|\phi_0\rangle$ with probability at least $1 - \frac{1}{\text{poly}(R)}$, which is a contradiction. So assume for a contradiction that

$$\||\Phi_x\rangle - |\Psi_y\rangle\| = \Delta(A_x, B_y) \leq \frac{1}{R^{1.5}}$$

We have

$$\frac{1}{R^{1.5}} \geq \||\Phi_x\rangle - |\Psi_y\rangle\| \geq \||0\rangle|\phi_0\rangle - |0\rangle|\psi_0\rangle\| + \sum_{i=0}^{\frac{R}{2}-1} \||2i+1\rangle U_{2i+1}|\phi_i\rangle - |2i+1\rangle|\psi_{i+1}\rangle\| +$$

$$+ \sum_{i=1}^{\frac{R}{2}-1} \||2i\rangle|\phi_i\rangle - |2i\rangle U_{2i}|\psi_i\rangle\| + \||R\rangle|\phi_R\rangle - |R\rangle U_R|\psi_{\frac{R}{2}}\rangle\|$$

Thus

$$\||\phi_0\rangle - |\psi_0\rangle\| + \sum_{i=0}^{\frac{R}{2}-1} \|U_{2i+1}|\phi_i\rangle - |\psi_{i+1}\rangle\| + \sum_{i=1}^{\frac{R}{2}-1} \||\phi_i\rangle - U_{2i}|\psi_i\rangle\| + \||\phi_R\rangle - U_R|\psi_{\frac{R}{2}}\rangle\| \leq \frac{1}{R^{1.5}} \qquad (4)$$

Notice that

$$\||\psi_1\rangle - U_1|\psi_0\rangle\| \leq \||\psi_1\rangle - U_1|\phi_0\rangle\| + \|U_1|\phi_0\rangle - U_1|\psi_0\rangle\| = \||\psi_1\rangle - U_1|\phi_0\rangle\| + \||\phi_0\rangle - |\psi_0\rangle\|$$

More generally:

$$\||\psi_{i+1}\rangle - U_{2i+1}\ldots U_1|\psi_0\rangle\| \leq \||\psi_{i+1}\rangle - U_{2i+1}|\phi_i\rangle\| + \|U_{2i+1}|\phi_i\rangle - U_{2i+1}\ldots U_1|\psi_0\rangle\| =$$

$$\||\psi_{i+1}\rangle - U_{2i+1}|\phi_i\rangle\| + \||\phi_i\rangle - U_{2i}\ldots U_1|\psi_0\rangle\| \leq$$

$$\||\psi_{i+1}\rangle - U_{2i+1}|\phi_i\rangle\| + \||\phi_i\rangle - U_{2i}|\psi_i\rangle\| + \|U_{2i}|\psi_i\rangle - U_{2i}\ldots U_1|\psi_0\rangle\| =$$

$$\||\psi_{i+1}\rangle - U_{2i+1}|\phi_i\rangle\| + \||\phi_i\rangle - U_{2i}|\psi_i\rangle\| + \||\psi_i\rangle - U_{2i-1}\ldots U_1|\psi_0\rangle\|$$

Thus, by induction we get that

$$\||\psi_j\rangle - U_{2j-1}\dots U_1|\psi_0\rangle\| \leq \sum_{i=1}^{j} \||\psi_i\rangle - U_{2i-1}|\phi_{i-1}\rangle\| + \sum_{i=1}^{j-1} \||\phi_i\rangle - U_{2i}|\psi_i\rangle\| + \||\phi_0\rangle - |\psi_0\rangle\| \tag{5}$$

Notice that according to equation 4 the RHS is $\leq \frac{1}{R^{1.5}}$. Therefor

$$|\ \||\psi_j\rangle\| - \||\psi_0\rangle\|| \leq \||\psi_j\rangle - U_{2j-1}\dots U_1|\psi_0\rangle\| \leq \frac{1}{R^{1.5}}$$

Since $\||\psi_0\rangle\|^2 + 2\sum_{i=1}^{\frac{R}{2}} \||\psi_i\rangle\|^2 = 1$, we see that there exist $0 \leq i,j \leq \frac{R}{2}$ such that

$$\||\psi_i\rangle\| \leq \frac{1}{\sqrt{R+1}} \ \ , \ \ \||\psi_j\rangle\| \geq \frac{1}{\sqrt{R+1}}$$

Therefor

$$\frac{1}{\sqrt{R+1}} - \frac{1}{R^{1.5}} \leq \||\psi_0\rangle\| \leq \frac{1}{\sqrt{R+1}} + \frac{1}{R^{1.5}}$$

We now get that

$$\begin{aligned} |\||\phi_R\rangle\| - \||\psi_0\rangle\|| \leq \||\phi_R\rangle - U_R\dots U_1|\psi_0\rangle\| \leq \||\phi_R\rangle - U_R|\psi_{\frac{R}{2}}\rangle\| + \|U_R|\psi_{\frac{R}{2}}\rangle - U_R\dots U_1|\psi_0\rangle\| = \\ \||\phi_R\rangle - U_R|\psi_{\frac{R}{2}}\rangle\| + \||\psi_{\frac{R}{2}}\rangle - U_{R-1}\dots U_1|\psi_0\rangle\| \leq \\ \||\phi_R\rangle - U_R|\psi_{\frac{R}{2}}\rangle\| + \textstyle\sum_{i=1}^{\frac{R}{2}} \||\psi_i\rangle - U_{2i-1}|\phi_{i-1}\rangle\| + \sum_{i=1}^{\frac{R}{2}-1} \||\phi_i\rangle - U_{2i}|\psi_{2i}\rangle\| + \||\phi_0\rangle - |\psi_0\rangle\| \leq \frac{1}{R^{1.5}} \end{aligned} \tag{6}$$

where the last two inequalities follow from equations 5, 4 respectively. We thus have

$$\|\frac{1}{\||\phi_R\rangle\|} \cdot |\phi_R\rangle - \frac{1}{\||\psi_0\rangle\|} U_R \cdot U_{R-1} \cdot \ldots \cdot U_1|\psi_0\rangle\| \leq$$

$$\frac{1}{\||\psi_0\rangle\|} \cdot \left[\| \left(\frac{\||\psi_0\rangle\|}{\||\phi_R\rangle\|} - 1\right) |\phi_R\rangle\| + \||\phi_R\rangle - U_R \cdot U_{R-1} \cdot \ldots \cdot U_1|\psi_0\rangle\|\right] \leq^{(*)} \frac{1}{\||\phi_0\rangle\|} \cdot \frac{2}{R^{1.5}} \leq \frac{3}{R}$$

where inequality $(*)$ follows from equation 6.

As $\frac{1}{\||\psi_0\rangle\|}|\psi_0\rangle$ is in $S(V_{init})$ and $\frac{1}{\||\phi_R\rangle\|}|\phi_R\rangle$ is in $S(V_{acc})$ we get that $\mathcal{P}$ accepts $|\psi_0\rangle$ with probability at least $1 - \frac{1}{\text{poly}(R)}$, contradicting the fact that $f(x,y) = 0$. $\square$

We thus have that for any function f that has a $QMA(T,W)$ protocol with R rounds and whose error is at moat R^{-4}, we have a mapping

$$x \to A_x \ , \ y \to B_y \quad \subset \mathbb{R}^{2^{\text{poly}(T,W)}}$$

such that

$$f(x,y) = 1 \quad \Rightarrow \quad \Delta(A_x, B_y) \leq \frac{\sqrt{2}}{R^{2.5}}$$

$$f(x,y) = 0 \quad \Rightarrow \quad \Delta(A_x, B_y) \geq \frac{1}{R^{1.5}}$$

By applying lemma lemma 6 we get that we can actually reduce f to LSD where the inputs are subspaces of

$$\mathbb{R}^{2^{\mathrm{poly}\,(T,W)\cdot \mathrm{poly}(R)}}$$

As $R \leq T$ we get that

$$\mathbb{R}^{2^{\mathrm{poly}\,(T,W)\cdot \mathrm{poly}(R)}} = \mathbb{R}^{2^{\mathrm{poly}\,(T,W)}}$$

as we wanted. This completes the proof of theorem 7. □

4 Black Box Complexity

For a natural number N we let $[N] = \{1, ..., N\}$. We also define $\mathcal{F}_{n,k}$ to be the set of all functions from $\{0,1\}^n$ to $\{0,1\}^k$.

A partial function G from a set S to $\{0,1\}$ is a function from some subset of S to $\{0,1\}$. We use the notations $\mathrm{Dom}(G)$ to denote the domain of G. Such a G is also called a relation on S, or a promise problem on S. (The promise is that the inputs to G come from the domain of G). In case that the domain of G is the whole set S, we say that G is total. We will usually have that S is either $\mathcal{F}_{n,k}$ or $\{0,1\}^N$.

An m qubits quantum state is a unit vector in the space $\mathbb{C}^{2^m}$, and is usually represented as

$$\sum_{x\in\{0,1\}^m} \alpha_x |x\rangle \text{ , where } \sum_{x\in\{0,1\}^m} |\alpha_x|^2 = 1 \; .$$

The focus of this section is black box models of computation. Before we describe the different models we have some remarks about the notations we use. The input to a black box will be a vector of indeterminates $X_1, ..., X_N$, where we will usually have that $N = 2^n$. The variables X_i will take values from $\{0,1\}$ or in some cases from $\{0,1\}^k$ for some large k (typically $k = n$). A classical query to the black box is an index $i \in N$, and the answer is X_i. We will sometimes think of $X_1, ..., X_N$ as representing the truth table of some function $f : \{0,1\}^n \to \{0,1\}^k$ (or to $\{0,1\}$). In these cases we say that the input to the black box comes from $\mathcal{F}_{n,k}$ and that on a query i the answer is $f(i)$.

4.1 Classical Black Box Models

We now describe the different models of classical black box algorithms. We assume that the reader is familiar with these models so we only give a brief description here. For a more detailed presentation see [9].

Let G be a promise problem on $\mathcal{F}_{n,k}$. In the black box model we assume that the input, $f \in \mathcal{F}_{n,k}$, is given as a black box, meaning that given a query i to f, it returns the value $f(i)$. The deterministic black box complexity of G, denoted by $D(G)$, is the minimal query complexity of a deterministic black box algorithm that computes G.

Note that since G is a partial function we don't care what the algorithm does on inputs not from the domain of G. This will be the case for all the models that we define. The deterministic black box complexity of G is also known as the decision tree complexity of G, as it is quite easy to see that any deterministic black box algorithm for computing G that makes T queries can be viewed as a decision tree of depth T that computes G and vice versa.

Similarly we define the probabilistic black box complexity of G, $R_\epsilon(G)$, to be the minimal query complexity of a probabilistic black box algorithm that for every input $f \in \mathrm{Dom}(G)$ computes $G(f)$ with probability $\geq 1-\epsilon$ (where the probability is taken over the coin tosses of the algorithm). Let $R(G) = R_{\frac{1}{3}}(G)$.

It is obvious that probabilistic black box algorithms are stronger than deterministic black box algorithms. In particular one can easily construct a relation, G, for which $R(G) = O(1)$, but $D(G) = \Omega(2^n)$. In contrast, Nisan [22] showed that for a total G,

$$R(G) \leq D(G) \leq 27R(G)^3 .$$

We now give the definition of the black box MA complexity of G.

Definition 20 *An $MA(T,W)$ black box algorithm that ϵ-computes G is a non deterministic probabilistic black box algorithm, A, with the following properties:*

1. *For any 1-input, f, there is a witness w_f, such that the length of w_f is W and*

$$\Pr[A(w_f)^f = 1] \geq 1-\epsilon .$$

In other words, the probability, over the coin tosses of A, that A outputs 1 when given the witness w_f and a black box access to f, is at least $1-\epsilon$.

2. *For any 0-input, f, and any witness w_f of length W,*

$$\Pr[A(w_f)^f = 1] \leq \epsilon .$$

In other words, the probability, over the coin tosses of A, that A outputs 1 when given any witness w_f and a black box access to f, is at most ϵ.

3. *For any input f the algorithm makes at most T queries.*

The complexity of the algorithm is defined to be $T+W$. The $\epsilon - MA$ complexity of G, $MA_\epsilon(G)$, is defined to be the minimal complexity of an MA algorithm that ϵ-computes G. We let $MA(G) = MA_{\frac{1}{3}}(G)$. Notice that an efficient algorithm must use short witnesses and make a small number of queries.

It is quite easy to see that there are total functions G, for which $MA(G) = O(n)$, but $R(G) = \Omega(2^n)$ (for example we can take G to be 1 for every $f \neq 0$, and 0 on the zero function). It is also easy to see that there are relations G for which $MA(G) = O(1)$ but $MA_0(G) = \Omega(2^n)$ (that is, nondeterministic algorithms are not as strong as MA algorithms).

4.2 Quantum Black Box Models

We now define the quantum analogs of the probabilistic and MA black box algorithms. We first describe the way that quantum queries are made to the black box. We assume, as before, that the black box contains a function $f \in \mathcal{F}_{n,k}$.

As in the classical case a quantum query asks for the value of f at a certain point i, that is written in a quantum register. The main difference is in the way that the answer to the query is given. The answer is written to some other quantum register that is correlated with the register that held i. More formally, if prior to the query our quantum state was given by the m bit quantum vector (where $m \geq n+k$)

$$\sum_{x \in \{0,1\}^m} \alpha_x |x\rangle = \sum_{i \in \{0,1\}^n} \sum_{y \in \{0,1\}^k} \sum_{z \in \{0,1\}^{m-n-k}} \alpha_{i,y,z} |i\rangle |y\rangle |z\rangle \ ,$$

then the result of the query is the state

$$\sum_{i \in \{0,1\}^n} \sum_{y \in \{0,1\}^k} \sum_{z \in \{0,1\}^{m-n-k}} \alpha_{i,y,z} |i\rangle |y \oplus f(i)\rangle |z\rangle \ .$$

That is, the answer to the query was written to the register that held y. (We actually receive the xor of the content of the register and the answer). We use the notation

$$\mathbb{C}^{2^m} = \mathbb{C}^{2^n} \otimes \mathbb{C}^{2^k} \otimes \mathbb{C}^{2^{m-n-k}}$$

(rather then just working with $\mathbb{C}^{2^{n+k}}$) as we want to stress that besides the n bit query and the k bit answer we have additional $m-n-k$ bits that can be viewed as additional memory registers.

Another way of looking at quantum queries is by viewing them as a special kind of unitary transformations: a query to a black box that contains f is equivalent to applying the following unitary transformation $O^{(f)}$, which operates on basis elements in the following way

$$O^{(f)}(|i\rangle |y\rangle |z\rangle) = |i\rangle |y \oplus f(i)\rangle |z\rangle \ .$$

When it will be clear from the context that the queries are made to a black box holding f we will sometime write O instead of $O^{(f)}$.

The fact that we can query at the same time the value of f at many points (although we get the answers in superposition) is the advantage that quantum black box algorithms have over classical black box algorithms.

We now give the definition of a quantum black box algorithm.

Definition 21 *A quantum black box algorithm which makes T quantum queries is a sequence of unitary transformation $U_0, O_1, U_1, O_2, ..., O_T, U_T$ from $\mathbb{C}^{2^m}$ to itself (for some $m \geq n+k$), such that the operators U_i are arbitrary unitary transformations and the operators O_i are unitary transformation representing the queries made by the algorithm (that is $\forall i \ O_i = O^{(f)}$ when the input to the black box is f). We note that the operators O_i depend on the input f (as explained above) whereas the operators U_i do not. The value of the algorithm on an input f is the outcome of the measurements of the right most bit of the quantum state*

$$U_T \cdot O_T \cdot U_{T-1} \cdots O_1 \cdot U_0 |0\rangle$$

where the queries are made to a black box holding f. Notice that the outcome of this measurement is a random variable.

We say that the algorithm ϵ-computes G if for every input $f \in \mathrm{Dom}(G)$, we get that the probability that the algorithm outputs $G(f)$ is at least $1-\epsilon$. The ϵ-quantum black box complexity of G, denoted by $Q_\epsilon(G)$, is the minimal number of queries made by any quantum black box algorithm that ϵ-computes G. We let $Q(G) = Q_{\frac{1}{3}}(G)$.

This model was extensively studied. Peter Shor's algorithm for finding periodicity [28] was done in the black box model. In this model the first super-polynomial separation between classical computation and quantum computation were done [29, 7]. So for partial functions we know that the quantum black box model is stronger than probabilistic black box algorithms. At the other hand, Beals et al. [5] proved that for all total G,

$$Q(G) \leq R(G) \leq O(Q(G)^6) .$$

That is, a separation exist only for relations and not for total functions. The method used by Beals et al. to prove this polynomial equivalence is a generalization of the method developed by Nisan in his paper [22] for showing the polynomial equivalence of probabilistic and deterministic black box algorithms for total functions. We shall also use the same technique to prove some of our results. A description of the technique will be given in the following sections.

We now introduce our main model, quantum MA black box algorithms (QMA). This is the model that we mainly focus on, in this section.

Definition 22 *A black box $QMA(T,W)$ algorithm that ϵ-computes G is, as before, a sequence of unitary transformation $U_0, O_1, U_1, O_2, ..., O_T, U_T$ from $\mathbb{C}^{2^{(m+W)}}$ to itself (for some $m \geq n+k$), such that the operators U_i are arbitrary unitary transformations and the operators O_i are unitary transformation representing the queries made by the algorithm with the following properties:*

1. *For every 1-input, f, there is a quantum witness $\Psi_f \in \mathbb{C}^{2^W}$ such that the outcome of the measurement of the right most bit of the state*

   $$U_T \cdot O_T \cdot U_{T-1} \cdots O_1 \cdot U_0(|0\rangle|\Psi_f\rangle)$$

   is 1 with probability at least $1-\epsilon$ (where $|0\rangle \in \mathbb{C}^{2^m}$).

2. *For any 0-input and any $\Psi \in \mathbb{C}^{2^W}$ the outcome of the measurement of the right most bit of*

   $$U_T \cdot O_T \cdot U_{T-1} \cdots O_1 \cdot U_0(|0\rangle|\Psi\rangle)$$

   is 1 with probability at most ϵ.

As before, the operators O_i depend on the input f whereas the operators U_i do not. The complexity of the algorithm is, as its classical analog, $T+W$. We denote by $QMA_\epsilon(G)$ the minimal complexity of a QMA black box algorithm that ϵ-computes G, and let $QMA(G) = QMA_{\frac{1}{3}}(G)$.

The aim of this section is to better understand the power of QMA black box algorithms. In particular we study their power with respect to the models of classical black box MA algorithms and black box quantum algorithm. We will show some separations, thus proving that this model is indeed stronger, and we will also give some lower bounds for QMA black box algorithms. We state our results in the next subsection.

4.3 Results

We have two kinds of results. First we prove some lower bounds for QMA black box algorithms. Then we show some separations between QMA black box complexity and the other models.

Let NOR, $Parity$ be the following total functions from $\{0,1\}^N$ to $\{0,1\}$.

$$NOR(X_1, ..., X_N) = 1 \Leftrightarrow \forall i \;\; X_i = 0 \; ,$$

$$Parity(X_1, ..., X_n) = \bigoplus_{i=1}^{N} X_i \; .$$

Theorem 23 *Any QMA black box algorithm that ϵ-computes the function NOR or the function $Parity$ makes at least $(1-\epsilon')\frac{\pi}{4}\sqrt{N}$ queries, (where ϵ' depends on ϵ and goes to 0 when ϵ goes to 0).*

Note that in both lower bounds we ignore the length of the witnesses. This kind of a result is tight (up to a constant) because of our next theorem.

Theorem 24 *Any function (partial or total) G can be computed by a $QMA(\frac{\pi}{4}\sqrt{N}, N)$ black box algorithms.*

Since in our lower bounds we ignored the length of the witnesses, we couldn't have proved a lower bounds better than $\frac{\pi}{4}\sqrt{N}$.

On the other hand, our next theorem shows that for almost all total functions G, if we take into account the length of the witnesses as well, then their QMA complexity is $\Omega(N)$.

Theorem 25 *For almost all (total) functions $G : \{0,1\}^N \to \{0,1\}$ we have that $QMA(G) \geq \Omega(N)$.*

One of our proofs for Theorem 23 uses the polynomial method, a general technique that was developed by Nisan for proving lower bounds on decision trees [22] and then generalized by Beals et al. [5] to the case of quantum black box algorithms. In the next section we will introduce the technique and will show how to generalize it to the case of QMA algorithms.

The proof of Theorem 24 makes use of Grover's database search algorithm. The proof of the lower bound for random functions (Theorem 25) uses a (non standard) counting argument. We give the proofs of Theorems 23, 24, 25 in Section 4.5.

Let us now describe the separation results. First we observe that QMA black box algorithms are stronger than quantum black box algorithms.

Theorem 26 *Let G be the OR function. Then there is a (trivial) QMA algorithm of complexity $O(n)$ that computes G with probability 1, whereas any quantum black box algorithm that ϵ-computes G (for, say, $\epsilon = 1/3$) must make at least $\Omega(\sqrt{N}) = \Omega(2^{\frac{n}{2}})$ queries. (The trivial algorithm is actually a classical nondeterministic algorithm).*

Next, we show a promise problem that separates quantum black box algorithms from MA black box algorithms. Thus, by combining these results we get that QMA black box algorithms are stronger than both MA black box algorithms and quantum black box algorithm. Note that (as mentioned above) a promise problem that separates quantum black box algorithms from MA black box algorithms follows as a consequence of Watrous result [31]. Here we give a simpler proof of that separation result.

Theorem 27 *There is a promise problem G (the negation of Simon's problem), defined on $\mathcal{F}_{n,n}$, such that G is computable by a quantum black box algorithm that makes a polynomial (in n) number of queries, and such that any (classical) MA black box algorithm for G has complexity $\geq \Omega(2^{\frac{n}{4}})$.*

On the other hand, we show that for all total G if both G and $\neg G$ have a small QMA complexity, then they actually have a small deterministic complexity as well.

Theorem 28 *For every total G,*

$$D(G) = O(QMA(G)^6 + QMA(\neg G)^6).$$

Our bound actually holds even if we ignore the length of the witnesses. That is, if G and $\neg G$ both have QMA algorithms that make at most T queries (regardless of the length of the witnesses) then there is a (classical) deterministic algorithm for G that makes $O(T^6)$ queries.

We give the proofs of Theorems 26, 27, 28 in Subsection 4.6.

4.4 Techniques

We now give some definitions that originally appeared in Nisan's paper [22]. Let G be a partial function from $\{0,1\}^N$ to $\{0,1\}$. Let $\mathrm{Dom}(G) \subset \{0,1\}^N$ be the domain of G.

Definition 29 *The* block sensitivity *of G at an input $v \in \mathrm{Dom(G)}$, which we denote by $bs(G,v)$, is the maximal number b of pairwise disjoint sets $B_1, ..., B_b \subset [N]$ such that*

$$\forall 1 \leq i \leq b,\ G(v) \neq G(v^{B_i}) \ \text{ and } \ v^{B_i} \in \mathrm{Dom(G)}\ ,$$

where for a set $B \subset [N]$, the vector v^B is defined by

$$(v^B)_i = \begin{cases} v_i & i \notin B \\ 1 - v_i & i \in B \end{cases}$$

The block sensitivity of G is

$$bs(G) = \max_{v \in \mathrm{Dom(G)}} bs(G,v)\ .$$

The 0 block sensitivity of G is defined by

$$bs^0(G) = \max_{v : G(v)=0} bs(G,v)\ ,$$

and similarly the 1 block sensitivity is defined by

$$bs^1(G) = \max_{v : G(v)=1} bs(G,v)\ .$$

We say that a polynomial p approximates G if for every v we have that $0 \leq |p(v)| \leq 1$, and for every $v \in \mathrm{Dom(G)}$ we have that $|p(v) - G(v)| < \frac{1}{3}$. We define $\widetilde{\deg}(G)$ to be the minimal degree of a polynomial that approximates G. Note that we ignore the values of the polynomial outside $\mathrm{Dom}(G)$.

In [23] Nisan and Szegedy showed that for any total function G the parameters $bs(G)$ and $\widetilde{\deg}(G)$ are polynomially related. In particular, they proved that for every G (even partial functions), we have

$$bs(G) \leq 4\widetilde{\deg}(G)^2 .$$

For a survey of results concerning the relations between this parameters and other parameters we refer to [9].

In [5] a connection between the quantum black box complexity of a function G and $\widetilde{\deg}(G)$ was shown. In particular they proved that if G can be computed by a quantum black box algorithm that makes T queries, then there is a polynomial p of degree at most $2T$ that approximates G. Then, using the techniques of Nisan and Szegedy they concluded that for every total G,

$$\frac{1}{4}\sqrt{bs(G)} \leq \frac{1}{2}\widetilde{\deg}(G) \leq Q(G) .$$

We now generalizes the technique of [5] to the case of QMA algorithms. First we take another look at the way that quantum queries are represented.

Assume that the black box contains a binary vector $X_1, ..., X_N$. Let O be the unitary transformation that corresponds to a quantum query. That is

$$O(|i\rangle|y\rangle|z\rangle) = |i\rangle|y \oplus X_i\rangle|z\rangle.$$

Thus, we can write the action of O as

$$O(|i\rangle|y\rangle|z\rangle) = (1 - X_i) \cdot |i\rangle|y\rangle|z\rangle + X_i \cdot |i\rangle|1 - y\rangle|z\rangle.$$

And in general:

$$O\left(\sum_{i,y,z} \alpha_{i,y,z}|i\rangle|y\rangle|z\rangle\right) = \sum_{i,y,z}(\alpha_{i,y,z}(1 - X_i) + \alpha_{i,1-y,z}X_i) \cdot |i\rangle|y\rangle|z\rangle.$$

That is, we can think of the entries of the matrix that represents the unitary transformation O as linear functions in the variables X_i.

Lemma 30 *Let G be a partial function from $\{0,1\}^N$ to $\{0,1\}$. Then the acceptance probabilities of any $QMA(T,W)$ black box algorithm that ϵ-computes G can be written as a polynomial, $q(\overline{X},\overline{Y})$, in the variables X_i and a set of 2^{W+1} auxiliary variables (that take real values), $\{Y_j\}_{j\in\{0,1\}^{W+1}}$, such that*

1. $\deg_{\overline{X}}(q) \leq 2T$. *That is, the degree of q with respect to the $\overline{X}$ variables is at most $2T$.*
2. *The degree of any monomial of q with respect to the $\overline{Y}$ variables is exactly* 2.
3. *For any 1-input X, there exists some Y_X, such that $\|Y_X\|_2 = 1$ and such that*

$$q(X, Y_X) \geq 1 - \epsilon .$$

4. *For any 0-input X, and any Y such that $\|Y\|_2 = 1$, we have that*

$$q(X,Y) \leq \epsilon \ .$$

Our proof is only a slight modification of the proof of Lemma 4.2 from [5].

Proof According to the definition, any $QMA(T,W)$ black box algorithm that computes G can be viewed as a sequence of unitary transformations $U_0, O_1, ..., O_T, U_T$. By the discussion above, each entry of the matrix $A = U_T \cdot O_T \cdots U_1 \cdot O_1 \cdot U_0$ is a polynomial of degree at most T in the variables X_i. Notice that the initial state of the algorithm is always of the form $|0\rangle|\Psi\rangle$, so it can be written as

$$|0\rangle|\Psi\rangle = \sum_{j\in\{0,1\}^W} \alpha_j |0\rangle|j\rangle$$

where $\sum |\alpha_j|^2 = 1$. Denote $\overline{X} = \{X_i\}_{i=1}^N$, $\overline{\alpha} = \{\alpha_j\}_{j\in\{0,1\}^W}$. We see that the final state of the algorithm, $A(|0\rangle|\Psi\rangle)$, can be written in the following form:

$$A(|0\rangle|\Psi\rangle) = \sum_{v\in\{0,1\}^{(m+W)}} q_v(\overline{X},\overline{\alpha})|v\rangle$$

where each q_v is a polynomial in $\overline{X}$ and $\overline{\alpha}$, such that as a polynomial in $\overline{X}$ its degree is at most T, and it is a linear form in $\overline{\alpha}$. We thus get that the result of the measurement is

$$q_{\text{final}}(\overline{X},\overline{\alpha}) = \sum_{v\in\{0,1\}^{m+W}:v_{m+W}=1} |q_v(\overline{X},\overline{\alpha})|^2 \ .$$

For every $j \in \{0,1\}^W$, let $Y_{j0} = \mathrm{Re}(\alpha_j)$ and $Y_{j1} = \mathrm{Im}(\alpha_j)$ be the real part and the imaginary part of α_j respectively. Let $\overline{Y} = \{Y_j\}_{j\in\{0,1\}^{W+1}}$. Then q_{final} can be viewed as a polynomial in $\overline{X}$ and $\overline{Y}$ of degree at most $2T$ in $\overline{X}$, such that the degree of any of its monomials in $\overline{Y}$ is exactly 2.

In addition, for every 1-input X there is a witness Ψ_X for which the algorithm outputs 1 with probability at least $1-\epsilon$. Therefore if we let α be the vector of coefficients of Ψ_X and Y as defined above then we see that indeed $\|Y\|_2 = 1$ and

$$q_{\text{final}}(X,Y) \geq 1-\epsilon \ .$$

Similarly for every 0-input X and any Y with $\|Y\|_2 = 1$, the value of $q_{\text{final}}(X,Y)$ is equal to the result of A on X and some witness that is implied from Y and therefore it is at most ϵ. This completes the proof of the lemma. □

Using the methods of Nisan and Szegedy we can now get the following bound.

Theorem 31 *Let G be a function from $S \subset \{0,1\}^N$ to $\{0,1\}$. Let $q(\overline{X},\overline{Y}) : \{0,1\}^N \times \mathbb{R}^k \to \mathbb{R}$ be a polynomial such that:*

1. *$\forall X$ such that $G(X) = 1$ there exists $Y_X \in \mathbb{R}^k$ such that*

   (a) $\|Y_X\|_2 = 1$.

(b) $q(X, Y_X) \geq 1 - \epsilon$.

2. $\forall X$ *such that* $G(X) = 0$ *and* $\forall Y \in \mathbb{R}^k$ *such that* $\|Y\|_2 = 1$ *we have that*

$$q(X, Y) \leq \epsilon \ .$$

Then,

$$\deg(q) \geq \sqrt{\frac{(1-2\epsilon) bs^1(G)}{2-2\epsilon}} \ .$$

The proof uses the following theorem from [11, 27].

Theorem 32 (Ehlich and Zeller; Rivlin and Cheney) *Let* $p : \mathbb{R} \to \mathbb{R}$ *be a polynomial such that for every integer* $0 \leq i \leq K$, $p(i) \in [0,1]$, *and such that its derivative satisfy* $p'(x) \geq c$ *for some* $c > 0$ *and some* $x \in [0, K]$. *Then* $\deg(p) \geq \sqrt{\frac{cK}{c+1}}$.

We now give the proof of Theorem 31. The proof is in two stages. First, given such q and G we construct a new polynomial, $\hat{q}$ from $\{0,1\}^{bs^1(G)}$ to $\mathbb{R}$, such that $\hat{q}$ has the same degree of q, and such that $\hat{q}$ is large on the all zero input, but is small on every input of weight 1. Then the second step is to consider the symmetrization of $\hat{q}$, which is a real polynomial of the same degree as $\hat{q}$, for which we can apply Theorem 32 and get the desired result. This technique was first used by Nisan and Szegedy [23].

Proof [of Theorem 31] We begin with the first step. Let $b = bs^1(G)$, and let $X_0 \in \mathrm{Dom}(G)$ be the input that achieves the block sensitivity. That is, $G(X_0) = 1$ and $b = bs(G, X_0)$. Let $B_1, ..., B_b$ be the pairwise disjoint blocks on which X_0 is sensitive. Let Y_{X_0} be such that $q(X_0, Y_{X_0}) \geq 1 - \epsilon$. We now construct a new polynomial $\hat{q} : \{0,1\}^b \to [0,1]$. Let $X(Z) \in \{0,1\}^N$ be defined as

$$(X(Z))_j = \begin{cases} (X_0)_j & j \notin \cup_{i=1}^b B_i \\ (1-(X_0)_j)Z_i + (X_0)_j(1-Z_i) & j \in B_i \end{cases}$$

We define $\hat{q}(Z_1, ..., Z_b) := q(X(Z), Y_{X_0})$. Clearly $\hat{q} : \{0,1\}^b \to [0,1]$ is a polynomial and $\deg(\hat{q}) \leq \deg_{\overline{X}}(q)$. Let $e_i \in \{0,1\}^b$ be the vector that is zero everywhere except for the i'th coordinate. We have that

$$\hat{q}(\overline{0}) = q(X_0, Y_{X_0}) \geq 1 - \epsilon \ ,$$
$$\hat{q}(e_i) = q({X_0}^{B_i}, Y_{X_0}) \leq \epsilon$$

(since $G({X_0}^{B_i}) = 0$). This completes the first step of the proof. Now let $\hat{q}_{sym} : \{0,1\}^b \to \mathbb{R}$ be the symmetrization of $\hat{q}$. That is

$$\hat{q}_{sym}(Z_1, ..., Z_b) = \frac{1}{b!} \sum_{\sigma \in S_b} \hat{q}(Z_{\sigma(1)}, ..., Z_{\sigma(k)})$$

where S_b is the group of all permutations of $\{1, ..., b\}$. It is easy to verify the following properties of $\hat{q}_{sym}$.

- $\hat{q}_{sym}$ is a symmetric polynomial.

- $\deg(\hat{q}_{sym}) \leq \deg(B)$.
- If for every $Z \in \{0,1\}^b$ we have that $0 \leq \hat{q}(Z) \leq 1$, then we also have that $\forall Z \in \{0,1\}^b \;\; 0 \leq \hat{q}_{sym}(Z) \leq 1$.

The following lemma of Minski and Papert [19] shows that we can represent $\hat{q}_{sym}$ by a univariate polynomial.

Lemma 33 (Minski and Papert) *Let $\hat{q}_{sym} : \{0,1\}^b \to \mathbb{R}$ be a symmetric polynomial. Then there exists a univariate polynomial $p : \mathbb{R} \to \mathbb{R}$ such that for every $Z \in \{0,1\}^b$ we have that*

$$\hat{q}_{sym}(Z) = p(|Z|)$$

where $|Z|$ is the weight of $Z =$ the number of nonzero coordinates of Z.

Let p be the univariate polynomial representing $\hat{q}_{sym}$ that is guaranteed by Lemma 33. We have that

1. $\deg(p) = \deg(\hat{q}_{sym})$.
2. For every integer $i \in [0,b]$ we have that $0 \leq p(i) \leq 1$.
3. $p(0) \geq 1 - \epsilon$.
4. $p(1) \leq \epsilon$.

From the last two facts we get that there is some point $c \in [0,1]$ for which $p' \leq 2\epsilon - 1$ (p' is the derivative of p). So according to Theorem 32 (applied to the polynomial $1-p$) we get that

$$\deg(p) = \deg(1-p) \geq \sqrt{\frac{(1-2\epsilon)b}{2-2\epsilon}}\;.$$

So we have that

$$\deg_{\overline{X}}(q) \geq \deg(\hat{q}) \geq \deg(\hat{q}_{sym}) = \deg(p) \geq \sqrt{\frac{(1-2\epsilon)b}{2-2\epsilon}} = \sqrt{\frac{(1-2\epsilon)bs^1(G)}{2-2\epsilon}}\;.$$

□

Putting everything together we get the following corollary that gives a lower bound on $QMA(G)$ in terms of $bs^1(G)$.

Corollary 34 $QMA(G) \geq \frac{1}{4}\sqrt{bs^1(G)}$.

Proof By combining Lemma 30 and Theorem 31 we get that

$$2T \geq \deg(q) \geq \frac{\sqrt{bs^1(G)}}{2}\;.$$

Hence

$$T \geq \frac{\sqrt{bs^1(G)}}{4}$$

$\square$

We will now give a much simpler proof for the last corollary, that gives a better numerical constant, by a reduction to the well known lower bounds for Grover search algorithm.

Theorem 35 *For any G, and any $QMA_\epsilon(T,W)$ black box algorithm that ϵ-computes G, we have $T \geq (1-\epsilon')\frac{\pi}{4}\sqrt{bs^1(G)}$, (where ϵ' depends on ϵ and goes to 0 when ϵ goes to 0).*

The idea of the reduction is the following. Let X_0 be a 1-input that is sensitive on the disjoint blocks $B_1, ..., B_b$. Then any QMA algorithm that solves G must distinguish between the case where the black box holds X_0 to the cases where it holds ${X_0}^{B_i}$ for $i = 1, ..., b$. This is, in some sense, what a search algorithm does. In Grover's algorithm we also have to distinguish between the situation that the black box holds $\overline{0}$ and the situation that it holds a vector of weight 1. So all that we have to do is transform the vector X_0 to $\overline{0}$ and transform ${X_0}^{B_i}$ to e_i in such a way that a query to the black box holding the input to the search problem will be equivalent to a query to a black box holding the input for G. With this in mind we prove the theorem.

Proof Let $b = bs^1(G)$. Let $X_0 \in \mathrm{Dom}(G)$ be the input that achieves the block sensitivity. That is $G(X_0) = 1$ and $b = bs(G, X_0)$. We assume for simplicity that $X_0 = \overline{0}$. Let $B_1, ..., B_b$ be the pairwise disjoint blocks on which X_0 is sensitive. Let Y_0 be such that $q(X_0, Y_0) \geq 1-\epsilon$. We will now show how to deduce an algorithm for Grover's search problem from any algorithm that outputs 1 with high probability on X_0 and 0 with high probability on ${X_0}^{B_i}$. Let $\overline{Z} = (Z_1, ..., Z_b)$ be a vector of indetermined. For any $\overline{Z} \in \{0,1\}^b$ we define an input $X(Z) \in \{0,1\}^N$ (as in the proof of Theorem 31).

$$(X(Z))_j = \begin{cases} 0 & j \notin \cup_{i=1}^{b} B_i \\ Z_i & j \in B_i \end{cases}$$

Let $e_i \in \{0,1\}^b$ be as before the vector whose entries are all zero except for the i'th coordinate. It is easy to see that there exist a pair of unitary transformations U_1 and U_2 such that U_1 transforms any query $|\Psi\rangle$, to a black box holding $X(Z)$, to a query to a black box holding Z, and U_2 transforms the answer to query $U_1|\Psi\rangle$ to the answer to the query $|\Psi\rangle$. In other words

$$O^{X(Z)}|\Psi\rangle = U_2 \cdot O^Z \cdot U_1|\Psi\rangle$$

for any $|\Psi\rangle$, where O^V is the unitary transformation that corresponds to a query to a black box holding V. Now given an input Z to the search problem (and remember that we only consider $\overline{0}, e_1, ..., e_b$ as inputs) we start running the algorithm for G given the witness Y_{X_0} (our algorithm will be fooled to think that it is running on a the input $X(Z)$). Any time that the algorithm needs to query its input we instead query the black box holding Z by the above mentioned U_1 and U_2. We do so until the algorithm outputs his result. Since the algorithm for G distinguishes between $\overline{0}$ and $\overline{0}^{B_i}$ then our algorithm distinguishes between $\overline{0} \in \{0,1\}^b$ and e_i. Therefore the query complexity of every QMA algorithm for G is at least the query complexity of every QMA algorithm for the search problem in $\{0,1\}^b$ (with the same success probability). Our result follows from the following theorem of Bennett et al. [6].

Theorem 36 ([6]) *Any quantum black box algorithm that solves the Grover search problem with error at most ϵ makes at least $(1-\epsilon')\frac{\pi}{4}\sqrt{N}$ queries (for inputs of length N), where ϵ' depends on ϵ and goes to 0 when ϵ goes to 0.*

$\square$

Note that the last lower bound on QMA black box complexity actually bounds the number of queries, and ignores the length of the proof. We shall later see that using this kind of argument we will never be able to prove a lower bound better than $\Omega(\sqrt{N})$.

4.5 Lower and Upper Bounds

We begin with the proof of Theorem 23.

Theorem 23 *Any QMA black box algorithm that ϵ-computes NOR makes at least $(1-\epsilon')\frac{\pi}{4}\sqrt{N}$ queries, Any QMA algorithm that ϵ-computes Parity makes at least $(1-\epsilon')\frac{\pi}{4}\sqrt{N}$ queries. (Where ϵ' depends on ϵ and goes to 0 when ϵ goes to 0).*

Proof It is quite easy to see that $bs^1(NOR) = bs^1(Parity) = N$. Hence the proof follows by Theorem 35. In the case of NOR we can represent the proof very simply as follows: Observe that NOR accepts only one input (the all zero input). Therefore if $|\Psi\rangle$ is the witness for this input for the best QMA algorithm then a quantum black box algorithm can simply begin by creating the state $|0\rangle|\Psi\rangle$ and then running the rest of the QMA algorithm. This assures us that the query complexity of quantum black box algorithms for NOR is **exactly** equal to the QMA complexity of NOR. And we know that any quantum black box algorithm that ϵ-computes NOR makes at least $(1-\epsilon')\frac{\pi}{4}\sqrt{N}$ queries ([8, 6]). (Notice that the only reason that we could simulate QMA black box algorithms for NOR by quantum black box algorithms was that NOR has only one accepting input.) $\square$

We now show that if we ignore the length of the witnesses then any function G has a QMA black box algorithm that makes at most $\frac{\pi}{4}\sqrt{N}$ queries.

Theorem 24 *Any function (partial or total) G can be computed by a $QMA(\frac{\pi}{4}\sqrt{N}, N)$ black box algorithm.*

The idea of the proof is the following. For every input X to the black box we give X itself as a witness (a classical witness). Now, classically we would need N queries to the black box to make sure that the witness is indeed equal to the input of the black box. However, in the quantum setting we can use Grover's algorithm [12] to verify that the witness is equal to the black box, with only $\frac{\pi}{4}\sqrt{N}$ queries.

Proof Let X be the input to the black box. We analyze the following algorithm: First the algorithm gets X itself as a witness, then the algorithm verifies that the witness is indeed equal to the input to the black box (if it is not the case then the algorithm rejects). Then the algorithm outputs $G(X)$. Clearly for every X such that $G(X) = 1$ there is a witness for which the algorithm accepts X. On the other hand for every X such that $G(X) = 0$ then either we will get as a witness $X' \neq X$, and then the verifying stage rejects, or the algorithm outputs 0. So we only have to show how to verify that the black box is holding a given witness X, with a small number of queries. But this is exactly what the Grover's database search algorithm ([12]) does:

Theorem 37 (Grover) *Given $X \in \{0,1\}^N$ and a black box that holds some $Y \in \{0,1\}^N$ there is a quantum black box algorithm that makes $\frac{\pi}{4}\sqrt{N}$ queries and decides correctly, with probability $> \frac{2}{3}$, whether $X = Y$ or not.*

This completes the proof of the theorem. $\square$

Next we prove Theorem 25 that shows that for most boolean functions G, if we take into account the length of the witnesses as well as the number of queries, then their QMA complexity is $\Omega(N)$.

Theorem 25 *For almost all (total) functions $G : \{0,1\}^N \to \{0,1\}$ we have that $QMA(G) \geq \Omega(N)$.*

The idea of the proof is to bound (from above) the number of QMA algorithms of small complexity that compute different functions (and to show that this number is much smaller than the number of all functions). This seems like a difficult task, however according to Lemma 30 this number is at most the number of polynomials of both low degree in $\overline{X}$ and a small number of $\overline{Y}$ variables, that represent *different* functions. So we bound the number of such polynomials to get the result. A similar idea can be found in [4].

Proof According to Lemma 30, any function G that has a $QMA(T,W)$ algorithm can be represented by a polynomial $q(\overline{X},\overline{Y})$ such that $\deg_{\overline{X}}(q) \leq 2T$, any monomial in q is of degree 2 in $\overline{Y}$, and such that the number of Y variables is at most 2^{W+1}. We now show that if q is such a polynomial then all its coefficients can be bounded by some function of T.

Lemma 38 *Let $q(\overline{X},\overline{Y})$ be the polynomial representing a computation of a $QMA(T,W)$ black box algorithm. Then the absolute value of the coefficient of each monomial, of degree at most d in $\overline{X}$, that appears in q, is at most $2^{\binom{d+3}{2}+1}$. Hence the absolute value of any coefficient in q is at most $2^{\binom{2T+3}{2}+1}$.*

Proof Let M be a monomial appearing in q and let c_M be its coefficient. We bound $|c_M|$ by induction on $\deg_{\overline{X}}(M)$. If $\deg_{\overline{X}}(M) = 0$ then let $M = Y_i \cdot Y_j$. Consider the assignment $Y_i = Y_j = \frac{1}{\sqrt{2}}$ (or in case that $i = j$ take $Y_i = 1$) and set any other Y_t or X_t to zero. If $i \neq j$ then we get that $\frac{C_M}{2} = c_M \cdot Y_i \cdot Y_j = q(\overline{X},\overline{Y}) \in [0,1]$. If $i = j$ then we get in a similar way that $c_M \in [0,1]$. So assume now that we proved the claim for every monomial of degree at most d in $\overline{X}$. Assume w.l.o.g. that $M = Y_1 \cdot Y_2 \cdot \prod_{i=1}^{d+1} X_i$. Now consider the value of q on the following assignment: $Y_1 = Y_2 = \frac{1}{\sqrt{2}}$, $X_1 = ... = X_{d+1} = 1$, where the rest of the variables are assigned the zero value. The value of q on this assignment is determined by the value of M and the values of the other monomials that involve only $Y_1, Y_2, X_1, ..., X_{d+1}$. Thus the value of q at this assignment is $\frac{c_M}{2}$ plus a sum of some other coefficients (that were multiplied by $\frac{1}{2}$). Since this sum must be between 0 and 1 and there are at most $3 \cdot 2^{d+1} - 1$ monomials other than M in $X_1, ..., X_{d+1}$ we get using our induction hypothesis that $|c_M| \leq 2 + (3 \cdot 2^{d+1} - 1) \cdot 2^{\binom{d+3}{2}+1} \leq 2^{\binom{d+4}{2}+1}$. This completes the proof. $\square$

So we proved that the absolute value of any coefficients in q is at most $2^{\binom{2T+3}{2}+1}$.

We now observe that if q, p are two polynomials that represents *different* functions (that were computed by $QMA(T,W)$ black box algorithms) then there is some monomial M, such that

$$|c_M(q) - c_M(p)| \geq 2^{-N+2W+4}$$

(where $c_M(q), c_M(p)$ are the coefficients of M in q and in p respectively). That is, the coefficient of M in q is "far" from its coefficient in p. The reason is that there is some input $\overline{X}, \overline{Y}$ for which $|q(\overline{X}, \overline{Y}) - p(\overline{X}, \overline{Y})| \geq \frac{1}{3}$ (otherwise q and p compute the same function). Now, if all the coefficients of $q-p$ were less than $2^{-(N+2W+4)}$ then since there are at most $2^{2W+2} \cdot \binom{N}{2T} \leq 2^{N+2W+2}$ different monomials in $q-p$ we would get that on any input the value of $q-p$ is at most $2^{N+2W+2} \cdot 2^{-(N+2W+4)} < \frac{1}{3}$ in contradiction.

So now we have that the coefficients of any such q are at most $2^{\binom{2T+3}{2}+1}$ in their absolute value, that any such q has at most $2^{2W+2} \cdot \binom{N}{2T}$ different monomials, and that for any q, p that represent different functions there is some monomial for which their coefficients are "far" from each other.

This actually mean that if we partition the segment $[-2^{\binom{2T+3}{2}+1}, 2^{\binom{2T+3}{2}+1}]$ to shorter segments of length $2^{-(N+2W+4)}$ (there are $2^{\binom{2T+3}{2}+N+2W+6}$ such segments) and consider the mapping that assigns for every q the list of segments in which its coefficients fell (according to some ordering of the monomials) then this mapping is one to one (because for any p, q representing different functions there will be a monomial that didn't fell into the same segments in p and in q). Therefore the number of q's that compute different functions is at most

$$\left(2^{\binom{2T+3}{2}+N+2W+6}\right)^{2^{2W+2} \cdot \binom{N}{2T}} .$$

Now if $W \leq \frac{N}{4}$ and $T \leq \frac{N}{8}$ then this number is exponentially small compared to 2^{2^N}. On the other hand the number of total functions from $\{0,1\}^N$ to $\{0,1\}$ is 2^{2^N}. This mean that for most functions either $W \geq \frac{N}{4}$ or $T \geq \frac{N}{8}$. $\square$

4.6 Separations

In this section we show that there are relations for which QMA black box algorithms are exponentially stronger than MA black box algorithms and quantum black box algorithms. As mentioned above, this follows from Watrous result [31] and our main contribution here is a new (and simpler) proof.

We first give a (trivial) function that separates Q from QMA (actually from MA).

Let G be the OR function. That is $G(X_1, ..., X_N) = 1$ if and only if not all the X_i's are zero.

Clearly there is a classical MA algorithm that gets a witness of length $n = \log N$ and makes 1 query, that computes G correctly.

On the other hand, as we already mentioned, the lower bound on the quantum black box complexity for the Grover search algorithm (see for example [8, 6]) shows that $Q(G) = \Omega(\sqrt{N})$.

This proves Theorem 26.

We now show a relation that has a quantum black box algorithm with a polynomial (in $n = \log N$) number of queries, but any MA black box algorithm for it has complexity $\Omega(\sqrt[4]{N})$. This relation is a modification of the famous problem of Simon [29].

Proof We define a promise problem on $\mathcal{F}_{n,n}$. $f \in \mathrm{Dom}(G)$ either if f is one to one, or if f is two to one in the following sense:

$$\exists \alpha_f \in \{0,1\}^n \setminus \overline{0} \text{ such that } \forall x \in \{0,1\}^n \;\; f(x) = f(x \oplus \alpha_f) .$$

We defined $G(f)$ to be 1 if f is one to one and zero otherwise (remember that G is only defined on $\mathrm{Dom}(G)$). This G can be viewed as the negation of Simon's problem. Simon's original algorithm output α_f with high probability when the input $f \in \mathrm{Dom}(G)$ is a two to one function, and it is easy to see that by a slight modification to the algorithm we can get an algorithm that outputs 1 w.h.p on two to one functions and zero on one to one functions (from the domain of G of course). By negating the answer given by this algorithm we get an algorithm to our problem, of the same complexity as Simon's algorithm. That is, there is an algorithm for G that make at most n^3 queries and outputs the correct result with high probability.

Next we show that any MA algorithm for G have complexity at least $\Omega(\sqrt[4]{N}) = \Omega(2^{\frac{n}{4}})$. So assume that there is a $MA(T, W)$ algorithm that $\frac{1}{3}$-computes G. By repeating the probabilistic part of the algorithm $O(W)$ times we can make the error at most $\frac{1}{4^W}$. This increases the complexity by a multiplicative factor of W.

Now, since the witnesses are of length W, there is a set of 1-inputs whose size is at least $\frac{1}{2^W}$ fraction of all the 1-inputs, such that all the 1-inputs in the set have the same witness. Denote this set of 1-inputs by S. Now fix this witness. We get that the MA algorithm is now (after fixing the witness) a probabilistic algorithm that with probability at least $1 - \frac{1}{4^W}$ outputs the correct answer for every 1-input in S and every 0-input. Therefore we can fix the random bits of the algorithm in such a way that the resulting deterministic algorithm outputs the correct answer on at least $1 - \frac{2}{4^W}$ of the inputs in S and on at least $1 - \frac{2}{4^W}$ of the 0-inputs.

We now show that any deterministic black box algorithm (i.e. a deterministic decision tree) with the above property must make many queries.

Indeed, assume that there is a decision tree for this problem of depth d. Let us consider an accepting path of the decision tree. Assume that the queries and their answers on this path are $f(x_i) = y_i$ for $i = 1, ..., d$ and some $x_1, ..., x_d, y_1, ..., y_d$. Note that since this is an accepting path we must have that all the y_i are distinct. Given such a path we now compare the fraction of 1-inputs that traveled this path (we consider also 1-inputs not in S), to the fraction of 0-inputs that traveled this path. It is easy to see that

$$\frac{1}{N} \cdot \frac{1}{N-1} \cdots \frac{1}{N-d+1}$$

fraction of all the 1-inputs followed that path. Now since the d queries defined at most $\binom{d}{2}$ possible α's we see that at least

$$\frac{N-1-\binom{d}{2}}{N-1} \cdot \frac{1}{N} \cdot \frac{1}{N-1} \cdots \frac{1}{N-d+1} \geq$$
$$(1 - \frac{d^2}{2N}) \cdot \frac{1}{N} \cdot \frac{1}{N-1} \cdots \frac{1}{N-d+1}$$

fraction of the 0-inputs followed that path. Since the fraction of 1-inputs that were accepted by the decision tree is at least $(1 - \frac{2}{4^W}) \cdot \frac{1}{2^W}$ we get that at least

$$(1 - \frac{d^2}{2N}) \cdot (1 - \frac{2}{4^W}) \cdot \frac{1}{2^W}$$

fraction of the 0 inputs were accepted by the decision tree. Now since we assumed that the fraction of accepted 0-inputs is at most $\frac{2}{4^W}$ we must have that $\frac{d^2}{2N} \geq \frac{1}{2}$, hence $d \geq \sqrt{N}$. Hence the depth of the decision tree is at least $\sqrt{N}$, and therefore we get that $O(TW) \geq \Omega(\sqrt{N})$ (remember that

$O(TW)$ is the number of queries performed by the algorithm after reducing the error to $\frac{1}{4W}$). So we get that

$$T + W \geq \Omega(\sqrt[4]{N}) \ .$$

This completes the proof of the theorem. $\square$

Finally we prove Theorem 28.

Theorem 28 *For all total G we have that*

$$D(G) \leq c \cdot \max(QMA(G)^6, QMA(\neg G)^6)$$

for some constant c.

This theorem shows that when considering total functions, if both G and $\neg G$ have small QMA complexity, then they have a small deterministic complexity as well. Actually we prove that for a total function G, either any QMA algorithm for G must make at least $\Omega(\sqrt[6]{D(G)})$ queries (regardless of the length of the witnesses) or any QMA algorithm for $\neg G$ must make at least $\Omega(\sqrt[6]{D(G)})$ queries (regardless of the length of the witnesses). This is a generalization of a similar result known for decision trees that was proved by Nisan [22]. Our proof, is based on his following theorem.

Theorem 39 (Nisan) *Let G be a total function, then the deterministic black box complexity of G, $D(G)$, satisfies*

$$D(G) \leq \mathrm{bs}(G)^3 \ .$$

Proof [of Theorem 28] According to Theorem 35 we have that the number of queries performed by any QMA algorithm for G is at least $\frac{\sqrt{bs^1(G)}}{4}$ and the number of queries performed by any QMA algorithm for $\neg G$ is at least $\frac{\sqrt{bs^1(\neg G)}}{4} = \frac{\sqrt{bs^0(G)}}{4}$ (regardless of the length of the witnesses). Therefore either every QMA algorithm for G makes at least $\frac{\sqrt{bs(G)}}{4}$ queries, or any QMA algorithm for $\neg G$ makes at least $\frac{\sqrt{bs(G)}}{4}$ queries. Combining this with Theorem 39 we get our result. $\square$

Note that by a slight modification of the relation that we used for proving Theorem 27 we get a relation G such that both G and $\neg G$ have polynomial (in n) quantum black box algorithm, but any MA algorithm computing G or $\neg G$ have exponential (in n) complexity. Therefore there is no analog of Theorem 28 for relations.