New Lower-Bounds for Quantum Computation with Non-Collapsing Measurements

David Miloschewsky
Department of Computer Science, Stony Brook University, NY, USA

Supartha Podder
Department of Computer Science, Stony Brook University, NY, USA

Abstract

Aaronson, Bouland, Fitzsimons and Lee [5] introduced the complexity class PDQP (which was original labeled naCQP), an alteration of BQP enhanced with the ability to obtain non-collapsing measurements, samples of quantum states without collapsing them. Although SZK $\subseteq$ PDQP, it still requires $\Omega(N^{1/4})$ queries to solve unstructured search.

We formulate an alternative equivalent definition of PDQP, which we use to prove the positive weighted adversary lower-bounding method, establishing multiple tighter bounds and a trade-off between queries and non-collapsing measurements. We utilize the technique in order to analyze the query complexity of the well-studied majority and element distinctness problems. Additionally, we prove a tight $\tilde{\Theta}(N^{1/3})$ bound on search. Furthermore, we use the lower-bound to explore PDQP under query restrictions, finding that when combined with non-adaptive queries, we limit the speed-up in several cases.

2012 ACM Subject Classification Theory of computation → Quantum query complexity

Keywords and phrases Non-collapsing measurements, Quantum lower-bounds, Quantum adversary method

Digital Object Identifier 10.4230/LIPIcs.CCC.2025.12

Related Version *Full Version*: https://arxiv.org/abs/2411.04085

Funding Supported by US Department of Energy (grant no DE-SC0023179) and partially supported by US National Science Foundation (award no 1954311).

Acknowledgements We want to thank Kunal Marwaha for helpful discussions. We thank the anonymous reviewers for their valuable comments and suggestions.

1 Introduction

1.1 Background and motivation

The study of the quantum query complexity model has lead to a rewarding avenue of research in understanding quantum speed-ups [29, 12, 11, 19]. In this model, one is given access to an input x through a black-box and the goal is to evaluate $f(x)$ for some function f while minimizing the number of quantum queries to x. A variety of tools have been developed to lower-bound quantum query complexity, notably the polynomial method [11], where one uses the fact that a polynomial may describe the amplitudes of a quantum state, the hybrid method [12] and its generalization the adversary method [7, 33], which quantifies the difficulty of distinguishing between inputs with different images. The study and application of these techniques has been instrumental in understanding the complexity class characterizing quantum computation, BQP (Bounded-error Quantum Polynomial-time). Specifically, as one may be given access to x through an oracle O, quantum query complexity studies BQP^O and its relations to other complexity classes with access to O.

© David Miloschewsky and Supartha Podder;
licensed under Creative Commons License CC-BY 4.0
40th Computational Complexity Conference (CCC 2025).
Editor: Srikanth Srinivasan; Article No. 12; pp. 12:1–12:23
Leibniz International Proceedings in Informatics
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

One of the goals of studying BQP is to characterize the classes of functions which may be efficiently computable once fault-tolerant quantum computers have been realized. The difficulty of this realization and the need to mimic our current hardware has lead us to place various restrictions on BQP. One prominent issue, the presence of noise, motivated the study of noise in the quantum query complexity setting, i.e., to introduce noise to the oracle calls. This was explored in [26], where the authors showed that Grover's algorithm loses its speed-up even if only a single qubit is randomly affected with depolarizing noise. Furthermore, [14] defined the class NISQ (Noisy Intermediate-Scale Quantum), the class of problems solved by a BPP algorithm with access to a noisy quantum device, emulating some aspects of today's state of quantum computation. They showed an oracle separation between both BPP and BQP, and showed that NISQ algorithms cannot solve search as fast as BQP. These results point to the conclusion that the presence of noise considerably weakens the computational power.

Another intriguing limited setting of the query model is the non-adaptive query model [23]. Here we restrict the model to a single round a parallel queries and analyze the necessary width of the round, imposing that the queries are independent of each other. It was shown in [23] that this restriction thwarts the quantum speed-up in solving the search and element distinctness problems, implying that adaptivity is necessary in order to obtain polynomial speed-ups for some problems.

On the other hand, augmenting BQP with a metaphysical ability have also been a fruitful avenue in understanding the power of computation. One of the first considered variations of quantum mechanics was adding post-selection to quantum computation, PostBQP (Post-selected BQP). Post-selection is the ability that, given a quantum state, we may *select* a partial measurement outcome, regardless of its probability. It was shown that PostBQP = PP [2], establishing an equivalency between a quantum and classical computational class. However, PostBQP is well above BQP as even QMA, a quantum version of NP, is contained in PP.

One of the theories explaining quantum mechanics is the hidden variable theory, an interpretation where the system is described by a state and a hidden variable which stochastically determines the measurement results. DQP (Dynamical Quantum Polynomial-time) is defined as the set of languages solved by a classical deterministic polynomial-time Turing machine which can process the history of the hidden variables of a quantum algorithm [1]. At the moment, we only know that $\mathsf{BQP} \subseteq \mathsf{DQP} \subseteq \mathsf{EXP}$ and the best known search algorithm uses $O(N^{1/3})$ queries. However, [5] showed that adjusting DQP to CDQP (Circuit-indifferent DQP), adding a restriction to the hidden variable theory, gives us a $\Omega(N^{1/4})$ lower-bound on search.

A well-studied offspring of DQP is PDQP (Product Dynamical Quantum Polynomial-time), quantum computation powered with non-collapsing measurements. By non-collapsing measurements, we mean that given an arbitrary state $|\psi\rangle$, we sample a measurement result without collapsing the state. We may view this as a simplification of DQP as we are given samples from the history of computation. Defined in [5], the authors proved a $\Omega(N^{1/4})$ lower-bound on search and $\mathsf{SZK} \subseteq \mathsf{PDQP} \subseteq \mathsf{BPP}^{\mathsf{PP}}$.

A class related to PDQP is CBQP (Copy BQP), BQP with the ability to create an unentangled copy of any state. Note that CBQP can simulate non-collapsing measurement by creating a copy and measuring it. However, the ability to entangle copies together adds considerable power. It was shown that CBQP can solve any NP-hard problem in polynomial time using a single query and $O(\log(N))$ copies [9]. The class is positioned between $\mathsf{BQP}^{\mathsf{PP}}$ and PSPACE [21].

In our work, we revisit PDQP as, except for CDQP, it is known to be only slightly stronger than BQP as it may at most attain a polynomial speed-up over Grover's algorithm. To understand the lower-bound, we must first describe a caveat. In the standard quantum query model, the goal is to minimize the amount of queries Q. However, in models with an additional ability, such as creating a copy or performing a non-collapsing measurement, we must provide bounds on the uses of the ability, which we denote by P. In the case of non-collapsing measurements, without minimizing both Q and P, every problem of size N could be trivially solved using a single query and followed by exponentially-many non-collapsing measurements which are repeated until we learn enough information about the input. Furthermore, unrestricted non-collapsing measurements lead to quantum cloning, faster than light communication and constant communication complexity [5]. Therefore the measure of complexity is $Q + P$, the sum of the number of queries and number of non-collapsing measurements.

The search lower-bound may lead some to the conclusion that a PDQP algorithm with P non-collapsing measurements may be simulated by BQP with a squared overhead by running P separate computations in parallel. However, this intuition fails as the *deferred measurement principle* does not hold in PDQP. To explain why, consider the collision problem. Here, assuming $N \in \mathbb{N}$, one is given black-box access to a function $f : [N] \to [N]$ with the promise that for any $x \in [N]$, $k = |\{y : f(y) = f(x)\}|$ is either 1 or 2. The algorithm is as follows. First, we query all the indices into a superposition and call the oracle, obtaining,

$$|\psi'\rangle = \sum_{x\in[N]} |x\rangle\,|f(x)\rangle$$

Next, we measure the query output register, collapsing $|\psi'\rangle$ to $|\psi\rangle$ and obtaining $f(x_1)$ as our measurement result. We find ourselves with the following state,

$$|\psi\rangle = \begin{cases} |x_1\rangle\,|f(x_1)\rangle & \text{if } k = 1 \\ \frac{|x_1\rangle+|x_2\rangle}{\sqrt{2}}\,|f(x_1)\rangle & \text{if } k = 2 \end{cases}$$

By repeating $O(1)$ non-collapsing measurements of $|\psi\rangle$, we find k with high probability, solving the problem with both constant queries and non-collapsing measurements. On the other hand, without the partial measurement, the algorithm is no longer efficient. In contrast, $\tilde{\Theta}(N^{1/3})$ queries are optimal in BQP [13, 28]. Note that the proof of SZK $\subseteq$ PDQP uses the same partial measurement trick to solve the Statistical Difference problem, which is SZK-hard [27].

When it was introduced, PDQP was referred to as naCQP (non-adaptive Collapse-free Quantum Polynomial-time). The class naCQP is a restriction of CQP, the class with the ability to adapt the quantum computation based on the non-collapsing measurement results. The power of this class is unknown as there are no known algorithms which utilize the adaptive feature nor do we have any lower-bounds. We note that that the phrase *non-adaptive* in naCQP is not related to non-adaptive queries discussed above, naCQP has the ability to perform a query at any step of the algorithm.

Recently, non-collapsing measurements gained renewed interest. It was shown that the complexity class PDQMA (Product Dynamical Quantum Merlin-Arthur), the quantum counter-part of NP with the power of non-collapsing measurements, equals NEXP [6, 10]. Inspired by their work and several open directions in the original PDQP literature, we revisit the class. There are several settings to study PDQP in. One may focus on adapting unitaries based on non-collapsing measurement results, restricting the query adaptivity or limiting partial measurements. We explore the power of non-collapsing measurements with non-adaptive queries. To avoid any confusion, for the remainder of the paper, any discussion of non-adaptability will be concerned with queries.

It is important to stress that the point of these adjustments is not to argue for alterations of the postulates of quantum mechanics. Instead, we explore quantum computation and complexity from an alternative point. Therefore, the models we define to encompass the adjustments of BQP may use ideas which are not inherently present in quantum computation. This is unavoidable as if they could be described solely using the postulates of quantum mechanics, their complexity class would be equal to BQP.

1.2 Results

1.2.1 Lower-bounding methods

We obtain a trade-off between queries and non-collapsing measurements by developing the adversary method lower-bound in our setting. For simplicity, our results are presented using the basic adversary method from [7], but we prove its generalization, the positive weighted adversary method of [8] (see Subsection 7.1). When discussing query complexity, we will use the name of the complexity class as the identifier of the model. We note that in all cases, one recovers the original adversary method by setting the number of non-collapsing measurements to 1, essentially only measuring the end state. The result below exhibits that non-collapsing measurements provide us with a limited speed-up over BQP in problems which may be lower-bounded by the positive weighted adversary method.

▶ **Result 1** (**Adversary method on BQP with non-collapsing measurements**, directly from Lemma 18). *Let $f : \{0,1\}^{|\Sigma|} \rightarrow \{0,1\}$ be a boolean function and x an input. Define $X \subseteq f^{-1}(0)$, $Y \subseteq f^{-1}(1), R \subseteq X \times Y$. Based on X, Y, R, let m, m' be the minimum number of elements (x, y) for each $x \in X$ and $y \in Y$ respectively and l, l' are the most elements (x, y) where $x(i) \neq y(i)$ for each $i \in \Sigma$ and for each $x \in X$ and $y \in Y$ respectively.*

Any algorithm solving $f(x)$ in a PDQP *setting with Q queries and P non-collapsing measurements satisfies,*

$$QP = \Omega\left(\sqrt{\frac{mm'}{ll'}}\right)$$

We develop a similar trade-off between Q and P in the non-adaptive query setting. Defined in Subsection 5.1, we denote it with the superscript nq.

▶ **Result 2** (**Adversary method on non-adaptive BQP with non-collapsing measurements**, directly from Lemma 24). *Define f, x, m, m', l, l' as in Result 1. Any algorithm solving $f(x)$ in a* PDQP^{nq} *setting with Q queries and P non-collapsing measurements satisfies,*

$$QP = \Omega\left(\max\left\{\frac{m}{l}, \frac{m'}{l'}\right\}\right)$$

Additionally, we obtained a variation of Result 1 for CBQP by applying a similar proof technique.

▶ **Result 3** (**Adversary method on BQP with the ability to copy states**, directly from Lemma 26). *Define f, x, m, m', l, l' as in Result 1. Any algorithm solving $f(x)$ in a* CBQP *setting with Q queries and P copies satisfies,*

$$Q \cdot 2^P = \Omega\left(\sqrt{\frac{mm'}{ll'}}\right)$$

Table 1 Summary of bounds on all the computational models explored. The non-trivial bounds we establish or improve in this paper are colored red. The BQP bounds are added for reference.

Setting / Problem	BQP	PDQP	PDQP^{nq}
Unstructured search	$\tilde{\Theta}(\sqrt{N})$	$\tilde{\Theta}(N^{1/3})$	$\tilde{\Theta}(\sqrt{N})$
Majority	$\Theta(N)$	$\tilde{\Theta}(\sqrt{N})$	$\tilde{\Theta}(\sqrt{N})$
Collision	$\tilde{\Theta}(N^{1/3})$	$\Theta(1)$	$\Theta(1)$
Element distinctness	$\tilde{\Theta}(N^{2/3})$	$O(\sqrt{N})$ $\Omega(N^{1/4})$	$\tilde{\Theta}(\sqrt{N})$

1.2.2 Applications of the lower-bounding methods

We apply the results above to specific problems in order to draw inference about PDQP. Specifically, we consider the bounds on search, majority and element distinctness in the settings discussed above. A summary of the bounds is in Table 1. Proof of the applications of the bounds above may be found in Subsection 7.1.

In the search problem, we are given access to a function $f : [N] \to \{0, 1\}$ and the goal is to decide whether there exists an $x \in [N]$ such that $f(x) = 1$. The lower-bound on search in PDQP was one of the main results in [5], but it was not tight. As the method in Result 1 provides us with the same lower-bound, we prove a tighter bound in Lemma 19.

▶ **Result 4** (**Bounds on search**). *We obtain tight bounds for unstructured search. In* PDQP*, the bound is* $\tilde{\Theta}(N^{1/3})$*, while in* PDQP^{nq} *it is* $\tilde{\Theta}(\sqrt{N})$*.*

Next, we consider the majority problem. In this problem, assuming $n \in \mathbb{N}$ and $N = 2^n$, we are given $f : [N] \to \{0, 1\}$ and must decide whether $s = |\{x : f(x) = 1\}| \geq 2^{n-1}$. As majority may be solved efficiently in PP, we obtain an oracle separation between PP and PDQP.[1]

▶ **Result 5** (**Bounds on majority**). *We obtain tight bounds for the majority problem. In both* PDQP *and* PDQP^{nq}*, it is* $\tilde{\Theta}(\sqrt{N})$*.*

While the collision problem has $O(1)$ bounds, it is interesting that the same is not the case for the element distinctness problem. The problem is to decide, given $f : [N] \to [N]$, whether there exist x, y such that $x \neq y$ and $f(x) = f(y)$. In BQP, we have a clear connection between the two classes, as their bounds may be derived from each other [15]. However, this is not the case in any PDQP setting. It would be very interesting to characterize what causes this difference.

▶ **Result 6** (**Bounds on element distinctness**). *We obtain bounds for the element distinctness problem. In* PDQP*, the bound is between* $O(\sqrt{N})$ *and* $\Omega(N^{1/4})$*, while in* PDQP^{nq} *it is* $\tilde{\Theta}(\sqrt{N})$*.*

Finally, let us consider the CBQP setting. [9] showed an algorithm which uses a single query and $O(\log(N))$ copies to solve the search problem. Applying Result 3 shows that this algorithm is optimal.

[1] The lower-bound on search implies a separation between NP and PDQP, therefore implying this separation. However, we don't expect majority to separate PDQP and NP nor QMA.

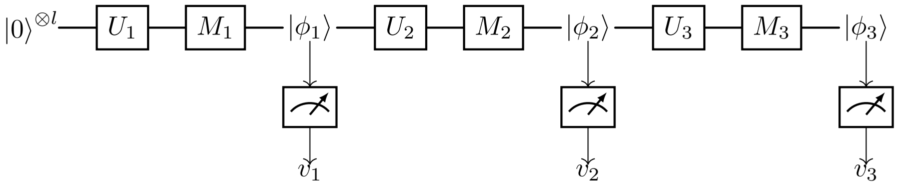


Figure 1 Example of the process of the oracle $\mathcal{Q}_P$ with $P = 3$. Given a circuit C, the oracle $\mathcal{Q}_P$ returns $\{v_i\}_i^3$ by measuring the states $\{\phi_i\}_i^3$ at each step of C.

1.3 Proof techniques

We describe the ideas behind our proofs. The key part of this paper is Definition 13, an alternative definition of the PDQP model. It allowed us to prove Lemma 15, enabling us to ignore partial measurement in the proofs of our lower-bounds.

1.3.1 Alternative definition of computation with non-collapsing measurements

Before explaining our alternative definition, we discuss the original definition of the model with non-collapsing measurements from [5]. PDQP is defined as the set of languages solvable by a deterministic Turing machine with one query to an oracle $\mathcal{Q}_P$, which runs a quantum circuit C with P steps. At step i, when the computation is in the state $|\phi_i\rangle$, the oracle obtains a measurement sample v_i of $|\phi_i\rangle$ without collapsing it. At the end, the machine receives $\{v_i\}_i$ and processes it. An example of how the oracle runs C is in Figure 1. We define an alternative oracle $\mathcal{Q}_P^*$ which inputs circuit C and outputs $\{v_i\}_i$ as well. However, we explicitly describe the quantum circuit C^* which $\mathcal{Q}_P^*$ runs. The goal is to describe C^* such that it outputs the tensor product of each state before the non-collapsing measurement. Therefore, we may obtain the results by measuring the end state.

Assume the input circuit C is an alternating combination of unitaries U_i and (potentially empty) partial measurement gates M_i. We create the variants U_i^* and M_i^* of these gates in C^*. The goal is that at every step i, we have $r_i = P - i + 1$ copies of the state $|\phi_i\rangle$ and our gates affect r_i states, each of whose goal is to end in $|\phi_j\rangle$ where $j \geq i$. An example of the circuit C^* is in Figure 2. The unitary U_i^* is a parallel application of U_i over each of the r_i registers separately. The measurement M_i^* is more complex as it must ensure that the partial measurement results are the same for all r_i registers in order for them to remain copies of each other. We define it as a gate which may post-select on measuring the same outcome for each register with an adjusted probability distribution.

Let us explain the probability adjustment. Suppose we are measuring a single qubit in each register and want to ensure that we measure $|0\rangle^{\otimes r_i}$ or $|1\rangle^{\otimes r_i}$. Let the probability of measuring 1 at step i of C be a. If we post-selected on this to happen in a PostBQP manner, the probability of measuring $1^{\otimes r_i}$ is $\frac{a^{r_i}}{a^{r_i}+(1-a)^{r_i}}$, not a as desired. Therefore we must adjust the probabilities of measuring $1^{\otimes r_i}$ by $d = a^{r_i - 1}$.[2] It is important to mention that M_i^* is not a valid quantum operator as its projectors don't sum to the identity. The only measurement which is allowed by quantum mechanics is the measurement of the states at the end of C^*. This is why PDQP is more powerful than BQP.

[2] This is the reason why it is difficult to directly show PDQP $\subseteq$ PostBQP.

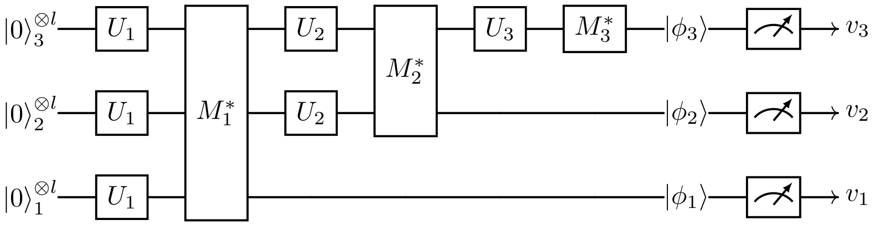


Figure 2 Example of the circuit C^* with $P = 3$ which is run by oracle $\mathcal{Q}_P^*$, producing the same results as oracle $\mathcal{Q}_P$.

1.3.2 Obtaining the lower-bounding techniques

Using our alternative definition, we show in Lemma 15 that given two arbitrary states $(\rho_{1,i}, \rho_{2,i})$ created by C^*, applying M_i^* cannot decrease the fidelity between them. Our adversary method uses fidelity as the progress measure, implying that we may ignore the effect of partial measurement during our proof. Finally, we obtain a positive weighted adversary method [8] for our model by closely following the original proof while accounting for the presence of copies of states, obtaining Lemma 18.

Our remaining lower-bounds use the lemmata we proved above. First, our result on non-adaptive queries with non-collapsing measurements is described in Lemma 22. The proof combines the original adversary method proof for non-adaptive queries [23], replacing the original positive weighted adversary with Lemma 18. Second, our tight $\Omega(N^{1/3})$ lower-bound on search in Lemma 19 is a direct extension of the proof on search in PDQP with no partial *collapsing* measurements (Appendix E in [5]) combined with our ability to ignore the effect of partial measurements which comes directly from Lemma 15.

Additionally, we apply the same idea of using non-standard gates to describe a computational model in order to obtain a lower-bounding technique for quantum computation with copies, CBQP. The techniques remain the same as above, except we allow for the entanglement between copies instead of instantly collapsing them.

1.4 Related work

Since the introduction of non-collapsing measurements [5], there have been several papers analyzing the adjustment in other settings. Specifically, we have $\mathsf{PDQP/qpoly} = \mathsf{ALL}$ [4] and $\mathsf{PDQMA} = \mathsf{NEXP}$ [6, 10]. Furthermore, it was shown in [20] that there exists an oracle O such that $\mathsf{P}^O = \mathsf{BQP}^O = \mathsf{SZK}^O = \mathsf{PDQP}^O \neq (\mathsf{UP} \cap \mathsf{coUP})^O$ and that relative to a random oracle O, $\mathsf{NP}^O \not\subseteq \mathsf{PDQP}^O$. Finally, [16] studied the effect on quantum theory when we may *only* perform non-collapsing measurements, meaning we may never collapse a quantum state.

Another alteration worth mentioning is adding the power to rewind from a collapsed result back to the state prior to measurement. Denoted RwBQP (Rewindable BQP), it was shown to be equal to CBQP and AdPostBQP (Adaptive PostBQP) [21]. Adaptivity in AdPostBQP means that based on partial measurement outcomes, we adapt the post-selection. Equivalently, RwBQP may be described as PDQP with the additional power to decide whether, once we obtain a non-collapsing measurement, we collapse the state to that sample. This means that adding some form of adaptivity to PostBQP and PDQP makes them equivalent.

Furthermore, PDQP without partial measurements may be viewed as BQP with parallel queries. Various bounds on this setting have been developed, such as for Grover's algorithm [32] and the general adversary method [22].

Closely related to parallel queries is adaptivity as non-adaptive algorithms have a single parallel query. Adaptivity of BQP queries has been a subject of prior research. The positive weighted adversary method was adjusted for non-adaptive queries [23] and it was shown that total functions depending on N variables require $\Omega(N)$ queries to the input [24]. Both results showed that non-adaptivity restricts the speed-ups quantum algorithms may achieve. Related to this, the analysis of the number of adaptive queries was studied in [18], where the authors showed that for any $r \in \mathbb{N}$, the $2r$-fold Forrelation problem can be solved with r adaptive rounds using one query per round, while $r-1$ adaptive round would require an exponential number of queries per round.

1.5 Open problems

We list several open questions worth pursuing below.

- In BQP, the bounds between the collision problem and the element distinctness problem are polynomially related. Our results show that this is not true in PDQP. Is it possible to characterize the reason behind this stark difference? We note that it is something which is not recognizable by any BQP algorithm.
- All efforts to lower-bound the query complexity of PDQP used methods derived from BQP. Is it possible to develop lower-bounding techniques which are tailored for non-collapsing measurements? Similarly, is it possible to obtain lower-bounds using an altered version of the polynomial method[3]?
- How does PDQP or other metaphysical classes act under a different restrictions. For example, how would it work with noisy computation? Are there restrictions which only affect BQP, but not PDQP?
- What is the power of PDQP if it can adapt its unitaries based on the non-collapsing measurement result?
- Are there modifications of BQP combined with the power of witnesses which are larger than NEXP?
- There exist oracle separations between PDQP and QMA in both ways[4], but there are other relationships worth exploring. For example, what is the relationship between PDQP and advice classes such as BQP/qpoly?

2 Preliminaries

We assume familiarity with basic quantum computation concepts; for introductory texts, see [25, 31]. Let us define the notation used in this paper. Unless mentioned otherwise, $N \in \mathbb{N}$ specifies the size of an arbitrary input. The set $\{1, .., N\}$ is denoted by $[N]$. For a string $s \in \{0,1\}^n$, $s(i)$ denotes the character in position i and the Hamming weight $|s| = |\{s(i) : s(i) = 1, i \in [n]\}|$. When describing a set $\{a_i\}_i^j$, we implicitly mean $\{a_i\}_{i=1}^j$. On the other hand, $\{a_i\}_i$ indicates a general set enumerated over i. The symbols U and M denote arbitrary unitary and measurement gates. The Kronecker delta function $\delta_{a,b}$ is a boolean function which is 1 if and only if $a = b$. A tensor product of k states ϕ is denoted by $\phi^{\otimes k}$ in order to avoid confusion with other superscripts. The symbol F denotes the fidelity measure, while f denotes an arbitrary boolean function. We list several key properties of fidelity.

[3] We discuss this in Subsection 7.2

[4] $\mathsf{QMA}^O \not\subseteq \mathsf{PDQP}^O$ holds by the lower-bound on search, while $\mathsf{PDQP}^O \not\subseteq \mathsf{QMA}^O$ is by an oracle separation between SZK and QMA [3] and SZK $\subseteq$ PDQP.

- (**Uhlmann's Theorem** [30]) Let ρ_1, ρ_2 be quantum states and $|\phi_1\rangle$ a fixed purification of ρ_1. Then $F(\rho_1, \rho_2) = \max_{|\phi_2\rangle} |\langle\phi_1|\phi_2\rangle|$ where the maximization is over purifications $|\phi_2\rangle$ of ρ_2.
- For any states ρ_1, ρ_2, $0 \leq F(\rho_1, \rho_2) \leq 1$.
- (**Multiplicativity**) For any states ρ_1, σ_1 and ρ_2, σ_2, $F(\rho_1 \otimes \rho_2, \sigma_1 \otimes \sigma_2) = F(\rho_1, \sigma_1)F(\rho_2, \sigma_2)$.
- (**Unitary invariance**) For any states ρ_1, ρ_2 and an arbitrary unitary U, $F(\rho_1, \rho_2) = F(U\rho_1 U^\dagger, U\rho_2 U^\dagger)$.

2.1 Quantum Query Model

We will be using the quantum query model. In this model, the goal is to compute $f(x)$ for some boolean function and an input $x = (x(0), ..., x(N-1))$. We are given access to x via an oracle O_x. We shall assume that we are working with a phase oracle, defined as follows,

$$O_x : |i, b\rangle \to (-1)^{x(i) \cdot b} |i, b\rangle$$

where i is an query input index of x and $b \in \{0, 1\}$. The definition of the quantum query model with non-collapsing measurements is in Section 3. We will use the following lemma.

▶ **Lemma 7** (Hybrid argument of [12]). *Let $|\psi_t\rangle$ and $|\psi_t^x\rangle$ be states of an algorithm solving search using Q queries in total after t steps without partial measurement where respectively there is no marked element or $x \in [N]$ is the marked element. Then,*

$$\sum_{x=1}^{N} \| |\psi_t\rangle - |\psi_t^x\rangle \|_2^2 \leq 4Q^2$$

The adversary method proof technique is developing an upper-bound on the progress an algorithm makes at one step in order to lower-bound the number of necessary steps. We will use the symbol $\Phi(t)$ as the algorithm progress measure at step t. In order to develop our positive weighted adversary lower-bound, we will use the notation from [8].

▶ **Definition 8** (Weight scheme of [8]). *Consider an arbitrary boolean function f. Let $X \subseteq f^{-1}(0)$, $Y \subseteq f^{-1}(1)$ and $R \subseteq X \times Y$. A weight scheme is defined using $w(x, y)$ and $w'(x, y, i)$ such that for all $(x, y) \in R$ and $i \in [N]$ where $x(i) \neq y(i)$,*

$$\begin{aligned} &w(x, y) > 0 \\ &w'(x, y, i) > 0 \\ &w'(y, x, i) > 0 \\ &w'(x, y, i)w'(y, x, i) \geq w^2(x, y) \end{aligned}$$

For a variable x, define the weight $wt(x)$ of a variable and the load of a variable $v(x, i)$ as,

$$\begin{aligned} wt(x) &= \sum_{y:(x,y) \in R} w(x, y) \\ v(x, i) &= \sum_{\substack{y:(x,y) \in R \\ x(i) \neq y(i)}} w'(x, y, i) \end{aligned}$$

▶ **Definition 9** (Weight load of [8]). *Assume a valid weight scheme as in Definition 8 for an arbitrary f. Let $C \in \{X, Y\}$. Then the maximum C-load is defined as,*

$$v_C = \max_{\substack{x \in C \\ i \in [N]}} \frac{v(x,i)}{wt(x)}$$

We define the maximum load of f as,

$$v_{max} = \sqrt{v_A v_B}$$

Additionally, we will use the following property from [8].

▶ **Proposition 10** (Weight identity, proof of Lemma 4 of [8]). *Assume variables defined as in Definitions 8 and 9. Let $\Phi(t) = \sum_{(x,y)\in R} w(x,y) F(\rho_{x,t}, \rho_{y,t})$ and $\alpha_{x,i}$ be the amplitude of $\rho_{x,t}$ associated with the query input index i. Then,*

$$\sum_{\substack{(x,y)\in R \\ i : x(i) \neq y(i)}} 2w(x,y) \left|\alpha_{x,i}\right| \left|\alpha_{y,i}\right| \leq v_{max} \Phi(0)$$

3 Non-collapsing measurement model

We provide a formal definition of the quantum query model with non-collapsing measurements. We begin with the original model from [5] and follow it with our alteration. At the start, we consider the case without an oracle, which we add at the end.

The original definition from [5] is as follows. Let P be the total number of non-collapsing measurements made and $\mathcal{Q}_P$ be a quantum oracle which takes as an input a quantum circuit $C = (U_1, M_1, .., U_P, M_P)$ where U_i is a unitary operator on l qubits and M_i is a collapsing measurement operator on s qubits such that $0 \leq s \leq l$. Let,

$$|\phi_0\rangle = |0\rangle^{\otimes l}$$
$$|\phi_i\rangle = M_i U_i |\phi_{i-1}\rangle$$

Define the oracle $\mathcal{Q}_P$ as a quantum oracle which inputs C and outputs $\{|\phi_i\rangle\}_i^P$. If this oracle is called by a machine which cannot hold quantum states, the machine receives the measurement results $\{v_i\}_i^P$ of $\{|\phi_i\rangle\}_i^P$ (i.e. upon receiving $|\phi_i\rangle$, the state immediately collapses to v_i).

▶ **Definition 11** (Section 2 of [5]). PDQP *is the class of languages L which may be recognized by a polynomial-time deterministic Turing machine with one query to $\mathcal{Q}_P$ with probability of error at most* $1/3$.

We propose an alternative definition of the computational model by altering the oracle $\mathcal{Q}_P$ to an explicit circuit. Let $r_i = P - i + 1$. The circuit C^* starts with P l-qubit registers, each initialized at zero. We label each register with index i which denotes which step of C, state $|\phi_i\rangle$, they compute. At each step of C^*, we apply adjusted gates U_i^* and M_i^* over r_i registers to all registers $j \geq i$, creating $\{|\phi_i\rangle\}_i$ through parallel computation.

Let $a_{i,n}$ be the probability of measuring n from M_i and $d_{i,n} = \frac{1}{a_{i,n}^{r_i - 1}}$. Let,

$$|\psi_0^*\rangle = |0\rangle^{\otimes lP} \tag{1}$$
$$U_i^* = U_i^{\otimes r_i} \tag{2}$$
$$M_i^* = \sum_n d_{i,n} (|n\rangle\langle n|)^{\otimes r_i} \tag{3}$$
$$|\psi_i^*\rangle = M_i^* U_i^* |\psi_{i-1}^*\rangle \tag{4}$$

We define $\mathcal{Q}_P^*$ as a quantum oracle which inputs C, runs C^* and outputs the final state $|\psi_P^*\rangle = \otimes_i^P |\psi_i\rangle$. As with $\mathcal{Q}_P$, if the machine calling $\mathcal{Q}_P^*$ cannot process quantum states, it receives $\{v_i\}_i^P$.

▶ **Proposition 12.** PDQP *is equivalent to the class of languages L which may be recognized by a polynomial-time deterministic Turing machine with one query to $\mathcal{Q}_P^*$ with error probability at most* 1/3.

Proof. By Definition 11, PDQP $=$ BPP$^{\mathcal{Q}_P,1}$ and we want to show PDQP $=$ BPP$^{\mathcal{Q}_P^*,1}$. As both $\mathcal{Q}_P$ and $\mathcal{Q}_P^*$ have the same inputs, we only require their outputs $\{v_i\}_i^P$ and $\{v_i^+\}_i^P$ are sampled from the same distribution. Consider an arbitrary pair v_i and v_i^+ which are the respective measurements of $|\psi_i\rangle$ and $|\psi_i^+\rangle$ where $|\psi_i^+\rangle$ is the resulting state on register i of C^*. As the application of U^* is U_i in parallel, only U_i is applied to register i in C^*. Therefore we only require that the probability distributions p and p^* of the measurement operators on $|\psi_i\rangle$ and $|\psi_i^+\rangle$ are equivalent.

Suppose that for $|\psi_i\rangle$, the probability of measuring an arbitrary string n during M_i is $p(n) = a$. For $|\psi_i^*\rangle$, we have that the probability of measuring $n^{\otimes r_i}$, the equivalent of n in C^*, is,

$$p^*(n) = d_{i,n} \left\langle \psi_i^* \middle| n^{\otimes r_i} \right\rangle \left\langle n^{\otimes r_i} \middle| \psi_i^* \right\rangle = d_{i,n}\, a^{r_i} = a$$

As the probability distributions p and p^* are equivalent, the oracles produce equivalent outputs. ◀

We shall add queries to our model. Let $Q_{i,x}$ the oracle operator applied at step i for input x, Q be the number of queries performed and $r_i = P + Q - i + 1$. In the quantum oracle model with non-collapsing measurements, we insert the oracle gates after the unitary U_i and measurement operator M_i. Without loss of generality, we may assume that we don't obtain a non-collapsing measurement at step i if we use Q_i at this step. This enables the circuit to choose when to perform queries and non-collapsing measurements. In $\mathcal{Q}_P^*$, this means that we apply a gate Q_i^* in parallel over r_i registers. You may see examples of the circuits being run by $\mathcal{Q}_P$ and $\mathcal{Q}_P^*$ in Figure 3.

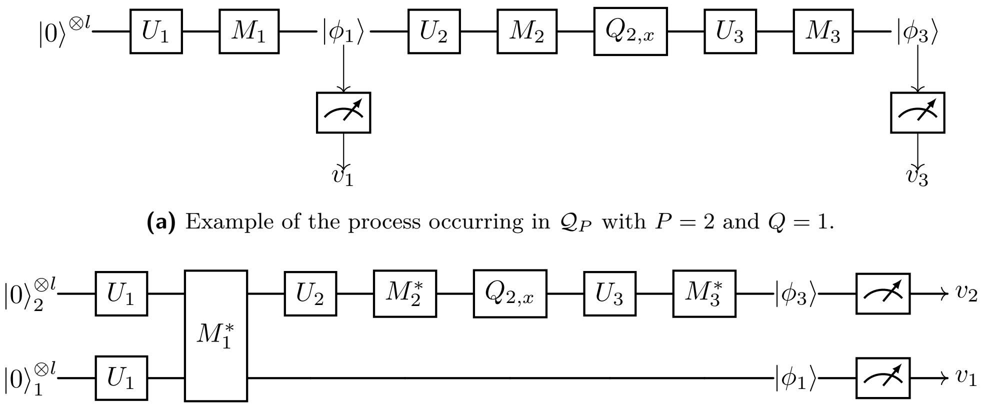
▶ **Definition 13.** *The* PDQP *query model creates a circuit C based on the input x, provides it to $\mathcal{Q}_P$ which runs the circuit C^*. The query complexity of the model is measured by the number of queries Q (steps of the algorithm which perform a query) and non-collapsing measurements P (steps which don't perform a query) in C.*

By using the definition above, we find that the query complexity of the two models is equivalent.

▶ **Lemma 14.** *The query complexity of the original* PDQP *query model is equivalent to the query complexity of the adjusted* PDQP *query model.*

Proof. Suppose that we query a circuit which performs Q queries and P non-collapsing measurements to the original oracle $\mathcal{Q}_P$. By our explanation of the query model above, we know that $\mathcal{Q}_P^*$ simulates the queries Q_i by the query gate Q_i^*, obtaining the same query complexity.

On the other hand, suppose we have a circuit that is run by $\mathcal{Q}_P^*$. Each unitary U_i^*, measurement M_i^* and query Q_i^* may be run by the original oracle PDQP by only running the gates on the first register, leading to the same complexity. ◀



(a) Example of the process occurring in $\mathcal{Q}_P$ with $P = 2$ and $Q = 1$.

(b) The circuit C^* in $\mathcal{Q}_P^*$ based on the example in **(a)**.

Figure 3 Examples of the quantum oracle model with non-collapsing measurements based on **(a)** Definition 11 and **(b)** Definition 13.

We note that one obtains that the gate complexity of the models is equal in a similar way. Furthermore, it is important to mention that by going from $\mathcal{Q}_P$ to $\mathcal{Q}_P^*$ and vice-versa under the same input, we adjust the width of the circuits while maintaining equal depth. By the characterization of fidelity using Bhattacharyya coefficients [17], we know that fidelity between states may not decrease after measurement. We find that the same holds for our adjusted measurement gates, allowing us to restrict the power of M^* in this model.

▶ **Lemma 15.** *Let $(\rho'_{1,i}, \rho'_{2,i})$ be the pair of states with different query oracles at step i of C^* before the application of M_i^*. Let $c \in \{1, 2\}$ and $\rho_{c,i} = M_i^* \rho'_{c,i} (M_i^*)^\dagger$. Then,*

$$F(\rho'_{1,i}, \rho'_{2,i}) \leq F(\rho_{1,i}, \rho_{2,i})$$

Proof. Let $c \in \{1, 2\}$. The proof of the statement is done by explicitly defining the states $\rho'_{c,i}$ as the sum of pure states whose purification results in the maximal fidelity, per Uhlmann's theorem. We show that for any such purification, the same purification process increases the fidelity when applied to $\rho_{c,i}$. This lower-bounds $F(\rho_{1,i}, \rho_{2,i})$ by Uhlmann's theorem, proving our statement.

Let k be an arbitrary integer. We define the state at each register the same due to the multiplicativity of fidelity. As the gate M_i^* only affects r_i registers, we shall ignore the rest. Each of the r_i registers is the following mixed state,

$$\sigma_{c,i} = \sum_{k \in [k]} b_{c,k} \left|\lambda_{c,k}\right\rangle \left\langle\lambda_{c,k}\right|$$

$$\left|\lambda_{c,k}\right\rangle = \sum_{j} \alpha_{c,j,k} \left|j\right\rangle_{\mathcal{M}} \left|\chi_{c,j,k}\right\rangle$$

where $b_{c,k} \in \mathbb{R}$, $\alpha_{c,j,k} \in \mathbb{C}$, $\mathcal{M}$ is the register being measured by M_i^* and $\chi_{c,j,k}$ is its associated workspace. The definition of $|\lambda_{c,k}\rangle$ in $\sigma_{c,i}$ is such that if we purify it as we do below, we obtain the maximum fidelity as shown by Uhlmann's theorem. Let $A = \otimes_j^{r_i} [k]_j$, $a \in A$ and $b^*_{c,a} = \prod_{r \in a} b_{c,r}$. Therefore the entire state $\rho'_{c,i}$ and its purification may be described as,

$$\begin{aligned}
\rho'_{c,i} &= \sigma_{c,i}^{\otimes r_i} = \sum_{a\in A} b^*_{c,a}(\otimes_{s\in a} |\lambda_{c,s}\rangle \langle\lambda_{c,s}|) \\
|\psi'_{c,i}\rangle &= \sum_{a\in A} \sqrt{b^*_{c,a}}(\otimes_{s\in a} |\lambda_{c,s}\rangle) |s\rangle
\end{aligned}$$

Let $a^*_{c,j,a} = \prod_{s\in A} \|\alpha_{c,j,s}\|_2^2$, d be the probability of measuring j in $\sigma_{c,i}$, $d_{i,j} = \frac{1}{d^{r_i-1}}$ and $\alpha^*_{c,j,a} = \prod_{s\in A} \alpha_{c,j,s}$. We apply M_i^* to $\rho'_{c,i}$, which adjusts the amplitudes and ensures that the register $\mathcal{M}$ measures to the same value j over all r_i registers. By using the definition of M_i^* from Line 3,

$$\begin{aligned}
\rho_{c,i} &= \sum_{a\in A,j} a^*_{c,j,a} d_{i,j} b^*_{c,a}(\otimes_{s\in a} |\chi_{c,j,k}\rangle \langle\chi_{c,j,k}|) \\
|\psi_{c,i}\rangle &= \sum_{a\in A,j} \alpha^*_{c,j,a}\sqrt{d_{i,j} b^*_{c,a}}(\otimes_{s\in a} |\chi_{c,j,k}\rangle) |s\rangle
\end{aligned}$$

By using our assumption about $\rho'_{c,i}$, we find,

$$\begin{aligned}
F(\rho'_{1,i},\rho'_{2,i}) &= \langle\psi'_{1,i}|\psi'_{2,i}\rangle && (5)\\
&= \sum_{\substack{a_1\in A_1\\ a_2\in A_2}} \sqrt{b^*_{1,a_1}b^*_{2,a_2}} \prod_{\substack{s_1\in a_1\\ s_2\in a_2}} \langle\lambda_{1,s_1}|\lambda_{2,s_2}\rangle\, \delta_{s_1,s_2} && (6)\\
&= \sum_{a\in A,j} \sqrt{b^*_{1,a}b^*_{2,a}} \prod_{s\in a} \langle\chi_{1,j,s}|\chi_{2,j,s}\rangle\, \alpha_{1,j,s}\alpha_{2,j,s} && (7)\\
&\leq \sum_{a\in A,j} \sqrt{b^*_{1,a}b^*_{2,a}\alpha^*_{1,j,a}\alpha^*_{2,j,a}d_{i,j}} \prod_{s\in a} \langle\chi_{1,j,s}|\chi_{2,j,s}\rangle && (8)\\
&= \langle\psi_{1,i}|\psi_{2,i}\rangle && (9)\\
&\leq \langle\psi_{1,i}^{max}|\psi_{2,i}^{max}\rangle && (10)\\
&= F(\rho_{1,i},\rho_{2,i}) && (11)
\end{aligned}$$

Lines 5- 7 are by our definitions above, Line 8 follows as $d_{i,j} \geq 1$, and Line 10 follows by Uhlmann's theorem where we define $(\psi_{1,i}^{max}, \psi_{2,i}^{max})$ as the pair of purifications which maximize $F(\rho_{1,i},\rho_{2,i})$.

4 Lower-bounds on non-collapsing measurements

In this section we expand the bounds established in [5]. First, we generalize their lower-bound to a wider selection of problems. Next, we show that the $\tilde{O}(N^{1/3})$ search algorithm from [5] is optimal up to logarithmic factors by obtaining a $\Omega(N^{1/3})$ lower-bound.

4.1 Weighted adversary method

Our proof closely follows the original proof of the positive-weighted adversary method [8]. There are two main adjustments. We employ Lemma 15 to argue that using partial measurements won't affect the progress measure. Additionally, we consider the effect of non-collapsing measurements by using the model in Definition 13. In order to accommodate the second change, we will require the following Proposition.

▶ **Proposition 16.** *For any $k \in \mathbb{N}$ and $r, s \in \mathbb{R}$ where $0 \leq r \leq 1$ and $0 \leq s \leq 2r$,*

$$ks \geq r^k - (r-s)^k$$

Proof. Let $A_{k,s} = ks - r^k + (r-s)^k$. Note that the statement holds when $k = 1$. Therefore let $k \geq 2$.

Assume $s \leq r$. We have,

$$\begin{aligned} A_{n+1,s} - A_{n,s} &= s - r^n(r-1) + (r-s)^n(r-s-1) \\ &= s - r^n s + r^n s - r^n(r-1) + (r-s)^n(r-s-1) \\ &= s(1-r^n) + ((r-s)^n - r^n)(r-1-s) \end{aligned}$$

By the bounds on r, s and the assumption that $s \leq r$, we have that $s(1-r^n) \geq 0$, $(r-1-s) \leq 0$ and $((r-s)^n - r^n) \leq 0$. As $A_{n+1,s} - A_{n,s} \geq 0$, the statement holds.

Assume $s > r$. Additionally, without loss of generality, assume k is odd. We find,

$$\begin{aligned} A_{k,s} &= ks - (r^k - (r-s)^k) \\ &\geq ks - (r^k - (r-2r)^k) \\ &= ks - 2r^k \end{aligned}$$

As $k \geq 2$ and $s > r$, $A_{k,s} \geq 0$. ◀

We define the progress measure as,

$$\Phi(t) = \sum_{(x,y)\in R} w(x,y) F(\rho_{x,t}, \rho_{y,t})$$

We bound the progress made at each step using the following lemma.

▶ **Lemma 17.** *Let f be a boolean function with a valid weight scheme and its associated maximum load v_{max}. Then the progress made by a* PDQP *algorithm each query $t \in [Q]$ is bounded by,*

$$\Phi(t-1) - \Phi(t) \leq P\, v_{max} \Phi(0)$$

Proof. By the unitary invariance of fidelity, we will ignore the application of unitaries. Similarly, by Lemma 15, we won't consider partial measurements as their effect may not decrease the progress measure. Therefore, without loss of generality, we assume our states are always pure. Additionally, we assume that during query t, we have D non-collapsing measurements left and have already performed $d = P - D$.

Let $|\psi_{x,t}\rangle$ denote the state of the algorithm on input x at query t and $\left|\psi'_{x,t}\right\rangle$ denote the same before the application of the oracle. Additionally, let $\left|\psi^*_{x,t}\right\rangle$ be the state during query i of C^* as defined in Definition 13, while $\left|\psi'^*_{x,t}\right\rangle$ is the state in C^* before the query. Using these definitions, we obtain the following identity,

$$\begin{aligned} \Phi(t-1) - \Phi(t) &= \sum_{(x,y)\in R} w(x,y) F(\psi^*_{x,t-1}, \psi^*_{y,t-1}) - \sum_{(x,y)\in R} w(x,y) F(\psi^*_{x,t}, \psi^*_{y,t}) && (12) \\ &\leq \sum_{(x,y)\in R} w(x,y) \left| \left\langle \psi^*_{x,t-1} \middle| \psi^*_{y,t-1} \right\rangle - \left\langle \psi^*_{x,t} \middle| \psi^*_{y,t} \right\rangle \right| && (13) \\ &= \sum_{(x,y)\in R} w(x,y) \left| \left\langle \psi'^*_{x,t} \middle| \psi'^*_{y,t} \right\rangle - \left\langle \psi^*_{x,t} \middle| \psi^*_{y,t} \right\rangle \right| && (14) \end{aligned}$$

Let $|\psi_{x,t}\rangle = \sum_i \alpha_{x,i} |i\rangle_{\mathcal{Q}} |\phi_{x,i}\rangle_{\mathcal{W}}$ where $\mathcal{Q}$ is the query register and $\mathcal{W}$ is the workspace register associated with that query register. As we have D non-collapsing measurements left to perform, an oracle call will affect the D copies of the state labeled by $\mathcal{O}$ and neglect the remaining d.

$$\left|\psi^*_{x,t}\right\rangle = |\psi_{x,1}\rangle \otimes .. \otimes |\psi_{x,d}\rangle \otimes (|\psi_{x,t}\rangle_{\mathcal{O}})^{\otimes D}$$

For the sake of simplicity, let,

$$\begin{aligned}
\hat{S}_i &= \sum_i \alpha_{x,i}\alpha_{y,i} \langle \phi_{x,i}|\phi_{y,i}\rangle \\
\hat{S}_{x(i)=y(i)} &= \sum_{i:x(i)=y(i)} \alpha_{x,i}\alpha_{y,i} \langle \phi_{x,i}|\phi_{y,i}\rangle \\
\hat{k} &= \sum_{i:x(i)\neq y(i)} \alpha_{x,i}\alpha_{y,i} \langle \phi_{x,i}|\phi_{y,i}\rangle
\end{aligned}$$

We find that,

$$\begin{aligned}
\Phi(t-1) - \Phi(t) &\leq \sum_{(x,y)\in R} w(x,y) \left|\left\langle \psi'^*_{x,t}\middle|\psi'^*_{y,t}\right\rangle - \left\langle \psi^*_{x,t}\middle|\psi^*_{y,t}\right\rangle\right| && (15)\\
&= \sum_{(x,y)\in R} w(x,y) |\langle \phi_{x,1}|\phi_{y,1}\rangle \ldots \langle \phi_{x,d}|\phi_{y,d}\rangle| \\
&\qquad \left|\hat{S}_{i_1}..\hat{S}_{i_D} - \left(\hat{S}_{x(i_1)=y(i_1)} - \hat{k}_1\right) \ldots \left(\hat{S}_{x(i_D)=y(i_D)} - \hat{k}_D\right)\right| && (16)\\
&\leq \sum_{(x,y)\in R} w(x,y) \left|\hat{S}_i^D - \left(\hat{S}_i - 2\hat{k}\right)^D\right| && (17)\\
&\leq \sum_{(x,y)\in R} 2w(x,y)D\hat{k} && (18)\\
&= D \sum_{\substack{(x,y)\in R\\ i:x(i)\neq y(i)}} 2w(x,y) |\alpha_{x,i}| |\alpha_{y,i}| && (19)\\
&\leq P\, v_{max} \Phi(0) && (20)
\end{aligned}$$

Line 15 was shown on Line 14, Line 16 uses the multiplicativity of fidelity, Line 17 holds as the inner product of two states is bounded by 1, Line 18 uses Proposition 16 by setting $r = \hat{S}_i$ and $s = 2\hat{k}$, and Line 20 is by Proposition 10 and $D \leq P$. ◀

Lemma 17 is sufficient to obtain the bound below.

▶ **Lemma 18.** *Let f be a boolean function with a weight scheme and its associated maximum load v_{max}. In a* PDQP *setting, the following holds,*

$$QP = \Omega\left(\frac{1}{v_{max}}\right)$$

Proof. Assuming success probability of $1-\epsilon$, we have that $\Phi(T) \leq 2\sqrt{\epsilon(1-\epsilon)}\Phi(0)$. By Lemma 17,

$$Q \geq \frac{\Phi(0) - \Phi(T)}{\Phi(t-1) - \Phi(t)} \geq \frac{1 - 2\sqrt{\epsilon(1-\epsilon)}}{Pv_{max}}$$ ◀

4.2 Lower-bound on search

We use the proof of a $\Omega(N^{1/3})$ lower-bound on PDQP without collapsing measurements from [5] and apply Lemma 15 in order to introduce partial measurement.

▶ **Lemma 19.** *Any algorithm solving the search problem in* PDQP *setting using Q queries and and P non-collapsing measurements satisfies,*

$$Q + P = \Omega(N^{1/3})$$

Proof. Consider a PDQP algorithm where $Q = P = o(N^{1/3})$. We compare the cases when there are no marked items versus when x is a marked item. Let their respective quantum states at step t be ρ_t, ρ_t^x and ψ_t and ψ_t^x be states of the same algorithm without partial measurement. By Lemma 7, we have,

$$\sum_x \|\psi_t - \psi_t^x\|_2^2 \leq 4Q^2$$

By summing over P non-collapsing measurements and using an averaging argument, there exists an x such that,

$$\sum_t \|\psi_t - \psi_t^x\|_2^2 \leq \frac{4PQ^2}{N}$$

By our assumption on the number of queries, this norm must be constant. Let's say that $\|\psi_t - \psi_t^x\|_2^2 \leq 0.01$. Let $\Phi = \otimes_t \rho_t$ represent all the states before measurement and V the distribution of their measurements. In order to notice the presence of a marked item, we require $|V - V_x|_1 \geq \Omega(1)$. Therefore,

$$\begin{aligned}
\Omega(1) \leq |V - V_x|_1 &\leq \|\Phi - \Phi_x\| && (21)\\
&\leq \sqrt{1 - F(\Phi, \Phi_x)^2} && (22)\\
&= \sqrt{1 - |\Pi_t F(\rho_t, \rho_t^x)|^2} && (23)\\
&\leq \sqrt{1 - |\Pi_t \langle\psi_t|\psi_t^x\rangle|^2} && (24)\\
&\leq \sqrt{1 - \Pi_t e^{-\|\psi_t - \psi_t^x\|_2^2}} && (25)\\
&\leq \sqrt{1 - e^{-\frac{4PQ^2}{N}}} && (26)\\
&= o(1) && (27)
\end{aligned}$$

Line 22 follows by the definition of trace norm, Line 24 holds as by Lemma 15, we may assume we did not perform partial measurements, and Line 25 uses the inequality

$$|\langle\psi_t|\psi_t^x\rangle| \geq \mathrm{Re}(\langle\psi_t|\psi_t^x\rangle) \geq e^{-\|\psi_t - \psi_t^x\|_2^2}$$

which holds as $1 - x \geq e^{-2x}$ when $0 \leq x \leq 0.01$. ◀

5 Non-collapsing measurements and non-adaptive queries

5.1 Non-adaptive query model

In this section, we describe the non-adaptive query model and prove its lower-bounds. Informally, the model performs all of its queries at the start of the computation, meaning that they are independent of the input x. We obtain a trade-off between Q and P.

The notation and proof closely follows that of [23]. Assume we perform Q non-adaptive queries. Given an oracle operator O_x for an input x, we define the query as,

$$O_x^Q |i_1, b_1, ...i_Q, b_Q\rangle \to (-1)^s |i_1, b_1, ...i_Q, b_Q\rangle$$

where $s = \sum_{j\in[Q]} x(i_j) \cdot b_j$. The algorithm may be described as follows,

$$M_P U_P ... M_1 U_1 O_x^Q U_0 |0\rangle$$

Assume we perform a non-collapsing measurement after every unitary except U_0. The label nq describes a variable in the non-adaptive query setting. For example, P^{nq} describes the number of non-collapsing measurements performed. Next, we define a variation of the weight scheme from Definition 8 for non-adaptive queries. The main idea of the proof is lower-bounding the minimum size of the initial parallel query by arguing that a smaller query would require at least two rounds of parallel queries. We use the superscript k to denote a variable for the parallel query of size $k \in \mathbb{N}$.

▶ **Definition 20** (Weight scheme of [23]). *Consider an arbitrary boolean function f and an input x. We define ${}^k x = x^k$, ${}^k f({}^k x) = f(x)$ and an index-tuple $I = (i_1, ...i_k)$. Therefore $x(I) = (x(i_1), ..., x(i_k))$.*

Let w, wt and v be the weight variables as in Definition 8 for f. Then W, WT and V are their variations for ${}^k f$. We let $W({}^k x, {}^k y) = w(x, y)$.

We will use the following lemma from [23],

▶ **Lemma 21** (Lemma 2 in [23]). *Let x be an input and w a weight function. For any $k \in \mathbb{N}$ and an index-tuple $I = (i_1, .., i_k)$,*

$$\frac{WT({}^k x)}{V({}^k x, I)} \geq \frac{1}{k} \min_{j\in[k]} \frac{wt(x)}{v(x, i_j)}$$

5.2 Weighted adversary method with non-adaptive queries

We prove the results on non-adaptive queries using the technique from [23], resulting in the lemma below.

▶ **Lemma 22.** *Consider a function f with a weight scheme w. Then any* PDQP^{nq} *algorithm with Q queries and P non-adaptive measurements satisfies,*

$$QP = \Omega\left(\max_{w} \max_{s\in\{0,1\}} \min_{\substack{x,i \\ f(x)=s}} \frac{wt(x)}{v(x,i)}\right)$$

In order to establish Lemma 22, we will first establish the following result which connects Lemma 18 and weights for non-adaptive inputs.

▶ **Lemma 23.** *Let w be a weight function for f. Then W is a valid weight function for ${}^k f$ and any algorithm solving ${}^k f$ using Q queries and P non-collapsing measurements satisfies,*

$$QP \geq C_\epsilon \min_{{}^k x, {}^y x, I} \left(\frac{WT({}^k x) WT({}^k y)}{V({}^k x, I) V({}^k y, I)}\right)^{1/2}$$

where $C_\epsilon = \sqrt{\frac{1-2\sqrt{\epsilon(1-\epsilon)}}{2}}$.

Proof. By Definition 20, as $W({}^k x, {}^k y) = w(x, y)$, we may apply Lemma 18 to obtain the bound. ◀

We will prove the following lemma:

▶ **Lemma 24.** *Let w be a valid weight function and x, y such that $f(x) \neq f(y)$. Then,*

$$Q^{nq}P^{nq} \geq C_\epsilon^2 \min_{x,y} \; \max \left(\min_i \frac{wt(x)}{v(y,i)}, \min_i \frac{wt(y)}{v(x,i)} \right)$$

Proof. Let $k \in \mathbb{N}$ be such that

$$k < C_\epsilon^2 \min_{x,y} \max \left(\min_i \frac{wt(x)}{v(x,i)}, \min_i \frac{wt(y)}{v(y,i)} \right) \tag{28}$$

We shall show that performing k non-adaptive queries would require us to repeat the computation more than once, implying that $Q^{nq}P^{nq} > k$. By using Lemma 21 and the fact that $WT({}^k x) \geq V({}^k x, I)$ for any x and I, we have that,

$$\begin{aligned} \frac{WT({}^k x)}{V({}^k x, I)} \frac{WT({}^k y)}{V({}^k y, I)} &\geq \max \left(\frac{WT({}^k x)}{V({}^k x, I)}, \frac{WT({}^k y)}{V({}^k y, I)} \right) \\ &\geq \frac{1}{k} \max \left(\min_{j \in [k]} \frac{wt(x)}{v(x, i_j)}, \min_{l \in [k]} \frac{wt(y)}{v(y, i_l)} \right) \end{aligned}$$

Therefore we have that,

$$\begin{aligned} \min_{{}^k x, {}^k y, I} \frac{WT({}^k x)}{V({}^k x, I)} \frac{WT({}^k y)}{V({}^k y, I)} &\geq \frac{1}{k} \min_{{}^k x, {}^k y, I} \; \max \left(\min_{j \in [k]} \frac{wt(x)}{v(x, i_j)}, \min_{l \in [k]} \frac{wt(y)}{v(y, i_l)} \right) \\ &\geq \frac{1}{k} \min_{x,y} \max \left(\min_i \frac{wt(x)}{v(x,i)}, \min_i \frac{wt(y)}{v(y,i)} \right) \\ &\geq \frac{1}{C_\epsilon^2} \end{aligned}$$

Where we applied our assumption on k. Therefore according to Lemma 23, $Q^{nq}P^{nq} > 1$, proving the lemma. ◀

Lemma 24 directly implies Lemma 22.

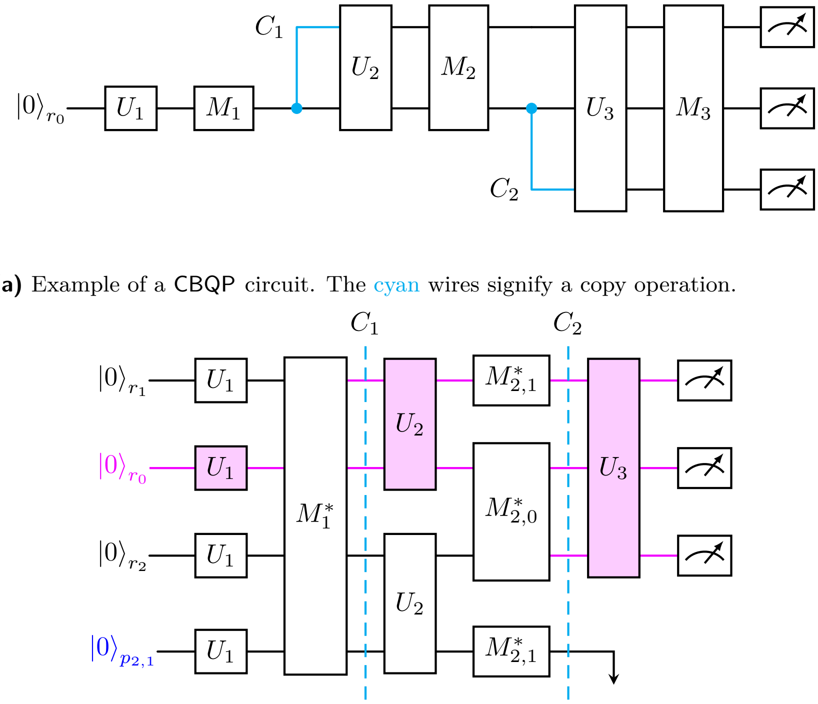
6 Lower-bounding quantum computation with copies

In this section, we apply the ideas from Lemma 18 to CBQP, quantum computation which is able to create a copy of any state. Another way to consider CBQP is as the computational class representing a world where the no-cloning theorem does not hold. The point of this section is to explicitly showcase the power difference between non-collapsing measurements and copying a state. As most steps in our analysis are identical, we only focus on the differences.

6.1 Computational model with copies

The key difference between PDQP and CBQP is that the latter has the ability to apply unitaries to the copies after creating them, while PDQP must immediately collapse them. As these copies may become entangled, purifying the algorithm in a similar manner as in

Definition 13, meaning explicitly creating copies of each state in order to analyze it, will require additional registers. Therefore, as we explain below, instead of using $O(P)$ registers as in PDQP, we will need $O(2^P)$ registers.[5]



(a) Example of a CBQP circuit. The cyan wires signify a copy operation.

(b) A purified version of the circuit above. The magenta register and wires are the same as in the circuit above, while the blue register is only to assist in the creation of a perfect copy and is not part of the original circuit above. Note that the gate $M_{2,1}^*$ is over registers r_1 and $p_{2,1}$.

Figure 4 Example of purifying a CBQP circuit.

Assume we perform P copies in total. Let r_0 be the starting register and for $i > 0$, r_i be its copies at step i. Without loss of generality, assume we only copy and apply the oracle to r_0. The gate C_i symbolizes a copy gate, which creates a copy, allowing us to apply unitaries across all registers r_j for $j \leq i$. Prior to applying the C_i gate, we must ensure that r_i and r_0 are copies, meaning that similarly to Definition 13, we apply unitary U in parallel to both. Finally, we account for the effect of partial measurement by using a partial measurement gate M_j^*, ensuring that partial measurement results before applying C_i are the same across all registers. When adding queries to this model, we follow the same steps as in Definition 13.

Finally, as each copy must mimic the register it is copying, we need to add further purification registers $p_{i,j}$ for each copy i. These registers mimic the process of the register until it has been copied. As each copy may be entangled, with each additional copy, the amount of purification registers doubles. In order to describe copy i of the computation, we require 2^i registers including the register r_i. Therefore the total number of registers is r^P. An example of a circuit with the ability to copy and its purification is in Figure 4.

[5] P denotes either the number of non-collapsing measurements or copies. This will be obvious from context.

6.2 Adversary lower-bound

We prove a version of Lemma 17 for CBQP.

▶ **Lemma 25.** *Let f be a boolean function with a valid weight scheme and its associated maximum load v_{max} as in Definition 9. Then the progress made by a* CBQP *algorithm each query t is bounded by,*

$$\Phi(t-1) - \Phi(t) \leq 2^P v_{max} \Phi(0)$$

where P is the number of copies.

Proof. First, we shall ignore the effect of unitaries and partial measurement by the same arguments and definitions of variables as in Lemma 17. In this case $D \leq 2^P$. We obtain the following result,

$$\Phi(t-1) - \Phi(t) \leq \sum_{(x,y)\in R} w(x,y) \left| \langle \psi'^*_{x,t} | \psi'^*_{y,t} \rangle - \langle \psi^*_{x,t} | \psi^*_{y,t} \rangle \right| \tag{29}$$

$$\leq \sum_{(x,y)\in R} w(x,y) \left| \hat{S}_i^D - \left(\hat{S}_i - 2\hat{k} \right)^D \right| \tag{30}$$

$$\leq 2^P v_{max} \Phi(0) \tag{31}$$

◀

We find the following relationship between the number of queries and copies.

▶ **Lemma 26.** *Let f be a boolean function with a weight scheme and its associated maximum load v_{max}. A* CBQP *algorithm with Q queries and P copies satisfies,*

$$Q 2^P = \Omega \left(\frac{1}{v_{max}} \right)$$

Lemma 26 implies that the algorithm in [9] to solve search is optimal and explains the discrepancy between CBQP and PDQP.

7 Bounds on problems with non-collapsing measurements

7.1 Proofs of applications of the lower-bounding techniques

We shall apply Lemmata 18 and 24 on concrete problems in order to compare the two models. First, let us note that the positive-weighted adversary technique may be reduced to the basic adversary technique by setting the weights $w(x,y)$ to 1.

▶ **Corollary 27.** *Let $f : \{0,1\}^n \to \{0,1\}$ be a boolean function, X, Y be two subsets of 1-inputs and 0-inputs respectively and R a relation $R \subseteq X \times Y$. Furthermore, let m, m', l, l' be the same as in Result 1. Then a* PDQP *query algorithm solving f with Q queries and P non-collapsing measurements satisfies,*

$$QP = \Omega \left(\sqrt{\frac{mm'}{ll'}} \right)$$

Furthermore, a PDQP *query algorithm with* non-adaptive *queries solving f with Q^{nq} queries and P^{nq} non-collapsing measurements satisfies,*

$$Q^{nq} P^{nq} = \Omega \left(\max \left\{ \frac{m}{l}, \frac{m'}{l'} \right\} \right)$$

Proof. In both Lemmata 18 and 24, let $w(x, y) = 1$. Then $w(x) \geq m$ or m' and $v(x, i) \leq l$ or l' depending on whether $x \in X$ or $x \in Y$. ◀

Let us obtain the values m, m', l, l' for the search, majority, collision and element distinctness problems. For each problem, we shall define the sets X, Y and their relation R which maximizes m, m' while minimizing l, l'. In addition to that, we also describe their algorithm. As has been done during the entirety of the document, we let N be the size of the input. Finally, we note that a summary of the results described below may be found in Table 1.

Search problem. We know that the query complexity of the search problem in PDQP is $\tilde{\Theta}(N^{1/3})$ due to the algorithm in [5], while our Lemma 19 proves that this algorithm is tight up to logarithmic factors. However, the same technique does not apply to PDQP with non-adaptive queries. We may split the inputs for search as follows:

- Let $X = \{0\}$ as $f(x) = 0$ if and only if $x = 0$.
- Let $Y = \{y : |y| = 1\}$ as these are the only values which only differ by 1 character from any input in X.

Therefore we find that $m = N$ as we may choose any index $i \in [N]$ to go from X to Y. Furthermore, $m' = l = l' = 1$ as each of these connections between X and Y only occur on one index. By Corollary 27, the query complexity of the search problem in the PDQP setting with non-adaptive measurements is $\Omega(\sqrt{N})$. This is tight as one may run an algorithm where we partition $[N]$ into $\sqrt{N}$ sets and query each separately. Afterwards, we may perform non-collapsing measurements on all partitions at once which, with $\tilde{O}(\sqrt{N})$ would return a marked item with high probability if there exists one.

Majority problem. To lower-bound the query complexity of the majority problem, we shall split the sets X and Y ones whose Hamming weight differs by one. Due to the structure of the problem, this is only possible when the Hamming weight is around $N/2$. Therefore let $X = \{x : |x| = N/2\}$, $Y : \{y : |y| = N/2 + 1\}$ and $(x, y) \in R$ if and only if $|\{x(i) : x(i) \neq y(i)\}| = 1$. This means that $m = N/2$, $m' = N/2+1$ and $l = l' = 1$. Therefore, by Corollary 27, the lower-bounds for PDQP is $\Omega(\sqrt{N})$ in both PDQP cases, regardless of the adaptivity of queries.

Similarly to the non-adaptive algorithm for search, the best algorithm for the problem is performing $\sqrt{N}$ queries partitions of the input, each of size $\sqrt{N}$. Afterwards, we may perform $\tilde{O}(\sqrt{N})$ non-collapsing measurements. By the coupon collector's problem, we will obtain all values with high probability, meaning the complexity of the problem is $\tilde{O}(\sqrt{N})$.

Element Distinctness problem. To analyze the element distinctness problem, we let X be the set of injective functions and Y the set of functions which are almost injective. By almost injective, we mean that there exists exactly one pair of inputs (x, x') such that $f(x) = f(x')$. Then we may define the relation between these sets of inputs as those which only require one change of outputs. Explicitly, $(f, f') \in R$ if and only if $|\{x : f(x) \neq f'(x)\}| = 1$. Thus every injective function may become almost injective by changing any one of its image to another image which already has a preimage. Therefore $m = N$ and $l = 1$. On the other hand, one may only choose one of two indices if they want to turn an almost injective function into an injective function by changing a single index. Therefore $m' = 2$ and $l' = 1$. Thus we find that the element distinctness problem in the PDQP query models with and without adaptive queries have a $\Omega(N^{1/4})$ and $\Omega(\sqrt{N})$ lower-bound respectively. Finally, the algorithm is the same as for non-adaptive search and the majority problem, giving us a query bound of $\tilde{O}(\sqrt{N})$.

7.2 Polynomial method lower-bounding technique

We applied a BQP lower-bounding technique to PDQP, but it is not tight in all situations. This may be due to the fact that the lower-bound establishes a QP relationship instead of $Q + P$. We note that the lower-bound on search in [5] had the same issue. It would be interesting if one was able to apply the polynomial method to the setting. One approach towards this would be going in the spirit of Definition 13. Let $p_i(x)$ be the polynomial associated with the amplitudes of the state at register i. One may attempt to restrict each $p_i(x)$ in two ways. First, for each i, one would ensure that $p_i(x)$ builds on $p_{i-1}(x)$ in that if we omitted the last query from $p_i(x)$, we would obtain $p_{i-1}(x)$. Second, the polynomial $p(x)$ describing the amplitude of the entire state is a product of all $p_i(x)$ with a caveat. Without partial measurement, $p(x) = \prod_i p_i(x)$. However, after partial measurement, $p_i(x)$ would become the sum of several polynomials, $p_i(x) = \sum_j p_i^j(x)$ where the superscript j signals a measurement result. In order to ensure partial measurement remains consistent along all polynomials, we would require $p(x) = \sum_j \prod_i p_i^j(x)$. We note that a list of other open problems we found interesting may be found in Subsection 1.5.

References

1 Scott Aaronson. Quantum computing and hidden variables ii: The complexity of sampling histories, 2004. arXiv:quant-ph/0408119.

2 Scott Aaronson. Quantum computing, postselection, and probabilistic polynomial-time, 2004. arXiv:quant-ph/0412187.

3 Scott Aaronson. Impossibility of succinct quantum proofs for collision-freeness, 2011. arXiv:1101.0403.

4 Scott Aaronson. PDQP/qpoly = ALL, 2018. arXiv:1805.08577.

5 Scott Aaronson, Adam Bouland, Joseph Fitzsimons, and Mitchell Lee. The space "just above" BQP, 2014. arXiv:1412.6507.

6 Scott Aaronson, Sabee Grewal, Vishnu Iyer, Simon C. Marshall, and Ronak Ramachandran. PDQMA = DQMA = NEXP: QMA with hidden variables and non-collapsing measurements, 2024. doi:10.48550/arXiv.2403.02543.

7 Andris Ambainis. Quantum lower bounds by quantum arguments, 2000. arXiv:quant-ph/0002066.

8 Andris Ambainis. Polynomial degree vs. quantum query complexity, 2004. arXiv:quant-ph/0305028.

9 Ning Bao, Adam Bouland, and Stephen P. Jordan. Grover search and the no-signaling principle. *Physical Review Letters*, 117(12), September 2016. doi:10.1103/physrevlett.117.120501.

10 Roozbeh Bassirian and Kunal Marwaha. Superposition detection and QMA with non-collapsing measurements, 2024. doi:10.48550/arXiv.2403.02532.

11 Robert Beals, Harry Buhrman, Richard Cleve, Michele Mosca, and Ronald de Wolf. Quantum lower bounds by polynomials, 1998. arXiv:quant-ph/9802049.

12 Charles H. Bennett, Ethan Bernstein, Gilles Brassard, and Umesh Vazirani. Strengths and weaknesses of quantum computing. *SIAM Journal on Computing*, 26(5):1510–1523, October 1997. doi:10.1137/s0097539796300933.

13 Gilles Brassard, Peter Høyer, and Alain Tapp. *Quantum cryptanalysis of hash and claw-free functions: Invited paper*, pages 163–169. Springer Berlin Heidelberg, 1998. doi:10.1007/bfb0054319.

14 Sitan Chen, Jordan Cotler, Hsin-Yuan Huang, and Jerry Li. The complexity of NISQ, 2022. doi:10.48550/arXiv.2210.07234.

15 Andrew Childs. Lecture notes on quantum algorithms, 2017.

16 Vincenzo Fiorentino and Stefan Weigert. A quantum theory with non-collapsing measurements, 2023. arXiv:2303.13411.

17 Christopher A. Fuchs and Carlton M. Caves. Mathematical techniques for quantum communication theory, 1996. `arXiv:quant-ph/9604001`.

18 Uma Girish, Makrand Sinha, Avishay Tal, and Kewen Wu. The power of adaptivity in quantum query algorithms, 2023. `doi:10.48550/arXiv.2311.16057`.

19 Lov K. Grover. A fast quantum mechanical algorithm for database search, 1996. `arXiv:quant-ph/9605043`.

20 Henrique Hepp, Murilo V.G. da Silva, and Leandro M. Zatesko. Oracle separations for non-adaptive collapse-free quantum computing. *Theoretical Computer Science*, 1030:115078, 2025. `doi:10.1016/j.tcs.2025.115078`.

21 Ryo Hiromasa, Akihiro Mizutani, Yuki Takeuchi, and Seiichiro Tani. Rewindable quantum computation and its equivalence to cloning and adaptive postselection. *Theory of Computing Systems*, 69(1), January 2025. `doi:10.1007/s00224-024-10208-5`.

22 Stacey Jeffery, Frederic Magniez, and Ronald de Wolf. Optimal parallel quantum query algorithms, 2015. `arXiv:1309.6116`.

23 Pacal Koiran, Jürgen Landes, Natacha Portier, and Penghui Yao. Adversary lower bounds for nonadaptive quantum algorithms, 2008. `arXiv:0804.1440`.

24 Ashley Montanaro. Nonadaptive quantum query complexity, 2010. `arXiv:1001.0018`.

25 Michael A. Nielsen and Isaac L. Chuang. *Quantum Computation and Quantum Information: 10th Anniversary Edition*. Cambridge University Press, 2010.

26 Ansis Rosmanis. Addendum to "Quantum search with noisy oracle", 2024. `arXiv:2405.11973`.

27 Amit Sahai and Salil Vadhan. A complete problem for statistical zero knowledge. Cryptology ePrint Archive, Paper 2000/056, 2000. URL: `https://eprint.iacr.org/2000/056`.

28 Yaoyun Shi. Quantum lower bounds for the collision and the element distinctness problems. In *The 43rd Annual IEEE Symposium on Foundations of Computer Science, 2002. Proceedings.*, SFCS-02, pages 513–519. IEEE Comput. Soc, 2001. `doi:10.1109/sfcs.2002.1181975`.

29 Peter W. Shor. Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. *SIAM Journal on Computing*, 26(5):1484–1509, October 1997. `doi:10.1137/s0097539795293172`.

30 A. Uhlmann. The "transition probability" in the state space of a *-algebra. *Reports on Mathematical Physics*, 9(2):273–279, 1976. `doi:10.1016/0034-4877(76)90060-4`.

31 John Watrous. The theory of quantum information, 2018.

32 Christof Zalka. Grover's quantum searching algorithm is optimal. *Phys. Review A*, 60(4):2746–2751, October 1999. `doi:10.1103/physreva.60.2746`.

33 Robert Špalek and Mario Szegedy. All quantum adversary methods are equivalent, 2006. `arXiv:quant-ph/0409116`.