The Quantum Setting with Randomized Queries for Continuous Problems

H. Woźniakowski*

Department of Computer Science, Columbia University, New York, USA, and
Institute of Applied Mathematics and Mechanics, University of Warsaw, Poland

October 2, 2018

Abstract

The standard setting of quantum computation for continuous problems uses deterministic queries and the only source of randomness for quantum algorithms is through measurement. Without loss of generality we may consider quantum algorithms which use only one measurement. This setting is related to the worst case setting on a classical computer in the sense that the number of qubits needed to solve a continuous problem must be at least equal to the logarithm of the worst case information complexity of this problem. Since the number of qubits must be finite, we cannot solve continuous problems on a quantum computer with infinite worst case information complexity. This can even happen for continuous problems with small randomized complexity on a classical computer. A simple example is integration of bounded continuous functions.

To overcome this bad property that limits the power of quantum computation for continuous problems, we study the quantum setting in which *randomized* queries are allowed. This type of query is used in Shor's algorithm. The quantum setting with randomized queries is related to the randomized classical setting in the sense that the number of qubits needed to solve a continuous problem must be at least equal to the logarithm of the randomized information complexity of this problem. Hence, there is also a limit to the power of the quantum setting with randomized queries since we cannot solve continuous problems with infinite randomized information complexity. An example is approximation of bounded continuous functions.

We study the quantum setting with randomized queries for a number of problems in terms of the query and qubit complexities defined as the minimal number of queries/qubits needed to solve the problem to within ε by a quantum algorithm. We prove that for path integration we have an *exponential* improvement for the qubit complexity over the quantum setting with deterministic queries.

*This research has been supported in part by the Defense Advanced Research Projects Agency (DARPA) and the National Science Foundation.

1 Introduction

One of the challenging problems of computational theory is the study of the power of quantum computation. By now there seems to be agreement about the standard setting of quantum computation, see [2, 7, 20]. This setting describes quantum computation as a sequence of unitary $2^k \times 2^k$ matrices acting on an initial state followed by a measurement. Here, k denotes the number of qubits. The unitary matrices in quantum computation are represented by elementary quantum gates, and one of them may represent a *query* which depends on the problem we want to solve. For continuous problems, queries are *deterministic* and depend on function values. Quantum algorithms may have many measurements but it is known that without loss of generality it is enough to consider quantum algorithms with only one measurement, see Remarks 1 and 2 and papers cited there.

In what follows it is important to stress the difference between the cost of an algorithm for solving a given problem and the computational complexity of this problem. The computational complexity (for brevity, the complexity) is the *minimal* computational resources needed to solve the problem. Examples of computational resources which have been studied include memory, time, and communications on a classical computer, and qubits, quantum gates and queries on a quantum computer.

We study quantum computation for continuous problems which are usually defined on spaces of functions. The quantum complexity of continuous problems has been studied in many papers and different queries, such as bit, phase and power queries, have been analyzed in the literature, see e.g., [4, 7, 8, 9, 10, 14, 15, 20, 21, 24, 26, 35]. In this paper we study bit queries, although some results hold for more general queries. What is important for our study is that a query for a continuous problem depends on at most 2^k function values computed at a priori given *deterministic* sample points. This means that in the standard quantum setting for continuous problems, quantum algorithms depend on at most 2^k function values and the only source of randomness comes through measurement.

The *query* complexity has been the focus of research. It is defined as the minimal number of queries needed to solve a given problem to within ε by a quantum algorithm. Since a critical resource for the foreseeable future is the number of qubits, we also study the *qubit* complexity which is defined as the minimal number of qubits needed to solve a given problem to within ε by a quantum algorithm.

We stress that there could be a trade-off between the query and qubit complexities since the minimization of queries may lead to a large number of qubits and vice versa. We do not know of such trade-offs for continuous problems studied so far. It is unknown if such trade-offs occur for some continuous problems.

To compute the quantum speedup one needs to know the worst case and randomized complexities on a classical computer. For continuous problems, the worst case and randomized classical complexities have been thoroughly studied in information-based complexity, see [7, 21, 28, 32, 33, 38]. For our purpose, we need the concept of (non-adaptive) information complexity which is defined as the minimal number of function values needed to solve the problem to within ε. We included two short sections on these classical settings to the extent needed in the rest of the paper.

Our first technical result is a relation between the standard quantum setting and the worst

case classical setting. Namely, it is relatively easy to see that since quantum algorithms are based on at most 2^k function values, they can not have a quantum error smaller than the worst case error of a classical algorithm based on these 2^k function values. This analogy is not complete since in the worst case setting we use deterministic algorithms whereas a quantum algorithm has a random element through measurement. Nevertheless, it is possible to show that the qubit complexity is bounded from below by the logarithm[1] of the worst case information complexity of the problem which we want to solve to within 2ε. We will show that this extra factor 2 takes care of randomness of quantum algorithms[2]. Since the worst case information complexity usually goes to infinity as ε tends to zero, the number of qubits must also increase to infinity although at a much slower rate due to the presence of the logarithm.

We also show that the qubit complexity is bounded from below by the (Kolmogorov) ε-entropy of the solution set. Hence, problems with large entropy of the solution set require a large number of qubits.

When the worst case information complexity or the entropy of the solution set is infinite then a finite number of qubits is not enough and the problem is unsolvable in the standard quantum setting. This can even happen for problems for which the randomized classical complexity is small. An example of such a problem is multivariate integration of continuous d-variate functions defined on, say, $[0,1]^d$, whose absolute values are bounded by 1. It is known that in this case the worst case information complexity is infinite but the Monte Carlo is optimal and the randomized information (as well as the total) complexity is roughly ε^{-2} independent of d.

Why can we solve this problem in the classical randomized setting and not in the standard quantum setting? The reason is that in the randomized setting we use function values at *randomized* points and potentially we can compute the function value at any point whereas in the standard quantum setting we use function values at *deterministic* points. The number of these points can be enormous, up to 2^k. But if we take a continuous function which vanishes at these 2^k points then we are unable to detect whether this function is zero or perhaps takes values equal to 1 or -1 at all points except an arbitrarily small neighborhoods of points at which it vanishes. The true solution may be zero or arbitrarily close to 1 or to -1. That is why any quantum algorithm in the standard quantum setting must fail.

This negative result is our point of departure. To overcome this bad property of quantum algorithms and to enlarge the power of quantum computation we propose a small modification of the standard quantum setting by allowing the use of *randomized* queries and *randomized* unitary matrices. The other assumptions are kept intact. We will call this modification as the *quantum setting with randomized queries* and refer sometimes to the standard quantum setting as the *quantum setting with deterministic queries*.

In fact, the idea of using randomized queries is not new. A particular kind of randomized query is used in Shor's algorithm for factoring of a (large) integer N, see [30] and also [20]. The essential part of Shor's algorithm is order finding which is solved by the query

$$Q_x|j\rangle \;=\; |j\,x \bmod N\,\rangle$$

[1] All logarithms in this paper are base 2.

[2] As indicated in the proof of Theorem 3.1 the extra factor 2 can be often omitted.

for $j = 0, 1, \ldots, 2^{\lceil \log N \rceil} - 1\}$ with a *random* x from $\{2, 3, \ldots, N-1\}$.

The use of randomized queries for continuous problems was also suggested in [24] for integration of non-smooth functions.

The quantum setting with randomized queries is the same as the standard quantum setting with the one important exception that queries as well as all unitary matrices used by a quantum algorithm may now depend on a random element. Hence, we have now two sources of randomness: one affecting unitary matrices and the other affecting measurement.

The quantum setting with randomized queries is an extension of the standard quantum setting. Indeed, if one always selects the same unitary matrices including the query, then we have exactly the standard quantum setting. Obviously, the use of randomized unitary matrices and randomized queries offers a possibility of much more efficient quantum computation. For some continuous problems, this extension is necessary. For example, we will show that multivariate integration of bounded and continuous functions, which cannot be solved in the standard quantum setting, is solvable in the quantum setting with randomized queries by a quantum algorithm that uses of order ε^{-1} queries and $\log \varepsilon^{-1}$ qubits. Hence, we have a quadratic speedup over the randomized setting on a classical computer.

We now comment on the error criteria used in the quantum settings with deterministic and randomized queries. In the standard quantum setting, i.e., in the quantum setting with deterministic queries, two error criteria are studied:

- the first error criterion is defined by taking the average performance of a quantum algorithm with respect to measurements for a worst function from the given class,
- the second error criterion is defined by taking the worst case performance of a quantum algorithm with respect to measurements on a set of measure $1 - \delta$ for a worst function from the given class.

The same error criteria are used in the quantum setting with randomized queries. In this case, randomization is richer and we take the average performance or a set of measure $1 - \delta$ with respect to "measurements, randomized queries and randomized unitary matrices". The first error criterion is studied in the main body of the paper whereas the second one is studied in the appendix.

We define the *randomized* (bit) query and qubit complexities analogously to the randomized setting on a classical computer. The randomized query complexity is defined as the minimum of the average number of randomized queries needed to solve the problem to within ε by a quantum algorithm. By "to within ε", we now mean that the error of a quantum algorithm is at most ε which is defined by taking the average performance with respect to all random elements of the quantum algorithm for a worst function from the given class.

The randomized qubit complexity is defined analogously as the minimal number of qubits for which there is a quantum algorithm whose error is at most ε. We stress that we assume the number of qubits is fixed and does not vary during quantum computation. This is probably a reasonable assumption from a practical point of view since a quantum computer with a random number of qubits seems too much to be expected in the near future. Nevertheless, from a purely theoretical point of view it would be interesting to study also the quantum setting with randomized queries and with random number of qubits and try to minimize the average number of qubits needed to solve the problem.

It is not surprising that the quantum setting with randomized queries is related to the randomized setting on a classical computer in the sense that the randomized qubit complexity is bounded from below by the logarithm of the randomized (non-adaptive) information complexity on a classical computer. If the randomized information complexity of a problem is infinite, the problem cannot be solved in the quantum setting with randomized queries. This happens, for example, for approximation of bounded continuous functions. Of course, the class of problems with infinite randomized information complexity is smaller that the class of problems with infinite worst case information complexity. So we extend the limit of what can be computed by presenting the quantum setting with randomized queries.

We study the quantum setting with randomized queries for a number of problems, and for some of them we prove an *exponential* improvement for the qubit complexity compared to the standard quantum setting. This is especially important since, as already mentioned, the number of qubits is a critical resource for the foreseeable future. In particular, the exponential improvement holds for path integration.

In this paper, we study real and Boolean summation, multivariate integration and path integration. We now briefly state the results obtained for these problems.

The real summation problem lies at the core of many continuous algorithms and plays a major role in the study of continuous problems in the standard quantum setting. The same is true in the quantum setting with randomized queries. It is known, see e.g., [7, 21], that the real summation problem can be reduced to Boolean summation. That is why it is enough to present in detail results for only the latter problem in which we want to approximate

$$\mathcal{B}_N(f) \;=\; \frac{1}{N} \sum_{j=0}^{N-1} f(j)$$

for a Boolean function $f : \{0, 1, \ldots, N-1\} \to \{0, 1\}$. Here N is a large integer which can be assumed to be a power of 2. We want to compute $\mathcal{B}_N(f)$ to within ε. Without loss of generality we may consider $\varepsilon^{-1} \ll N$. We now present the orders of the query and qubit complexities in the quantum settings with deterministic and randomized queries.

	Deterministic Queries	Randomized Queries
Query Complexity	ε^{-1}	ε^{-1}
Qubit Complexity	$\log N$	$\log \varepsilon^{-1}$

Figure 1: Boolean Summation

We stress that the minimal numbers of queries and qubits in a given setting are obtained by essentially the same quantum algorithm. In the quantum setting with deterministic queries,

this is the Boolean summation algorithm of [5] with seven repetitions as proved in [11]. In the quantum setting with randomized queries, we first approximate the Boolean mean $\mathcal{B}_N(f)$ by the Monte Carlo algorithm,

$$\mathrm{MC}_m(f,\omega) \;=\; \frac{1}{m}\sum_{j=1}^{m} f(\omega_j),$$

with m of order ε^{-2} and with independent and uniformly distributed ω_j from $\{0,1,\ldots,N-1\}$, and then use the Boolean summation algorithm with seven repetitions to approximate $\mathrm{MC}_m(f,\omega)$. It is interesting to notice that this algorithm uses randomized queries but the remaining unitary matrices are deterministic. This leads to the corresponding upper bounds. The lower bound proof of the query complexity in the standard case follows from [18]. For the quantum setting with randomized queries, we use the known fact that the randomized errors of quantum algorithms are no smaller than the average case errors with respect to Boolean functions. The latter problem with an appropriate measure on Boolean functions was solved in [25]. The lower bound proof of the qubit complexity is from the relation to the randomized information complexity.

We stress that we have the same order of query complexities in both cases. However, by allowing randomized queries we have an essential improvement in the number of qubits for solving the Boolean summation problem.

For the real summation problem, we want to approximate

$$\mathrm{SUM}_N(f) \;=\; \frac{1}{N}\sum_{j=0}^{N-1} f(j),$$

where $f:\{0,1,\ldots,N-1\}\to[0,1]$ may now take real values. For completeness, in Section 5 we show how the real summation problem may be reduced to the Boolean summation problem. In Corollary 5.1 we show that the results presented in Figure 1 also hold for the real summation problem.

We now turn to multivariate integration for functions which are r times differentiable and uniformly bounded. For $r=0$, the query and qubit complexities are infinity in the quantum setting with deterministic queries. For the quantum setting with randomized queries, they are finite and their orders are given in the following table.

	Deterministic Queries	Randomized Queries
Query Complexity	∞	ε^{-1}
Qubit Complexity	∞	$\log\,\varepsilon^{-1}$

Figure 2: Multivariate Integration for $r=0$

Hence, in this case the improvement of the quantum setting with randomized queries over the standard quantum setting is infinite.

We now assume that $r \geq 1$. Hence, functions are now at least once differentiable. In this case the orders of the query and qubits complexities are the same in both cases.

	Deterministic Queries	Randomized Queries
Query Complexity	$\varepsilon^{-1/(1+r/d)}$	$\varepsilon^{-1/(1+r/d)}$
Qubit Complexity	$\log \varepsilon^{-1}$	$\log \varepsilon^{-1}$

Figure 3: Multivariate Integration for $r \geq 1$

The query complexity in the standard quantum setting is due to [24]. The randomized query complexity has the same order since Boolean and real summation require roughly the same queries in both settings. The qubit complexities are of the same order since the logarithms of the worst case and randomized information complexities of multivariate integration are both proportional to $\log \varepsilon^{-1}$.

Finally, we consider a specific case of path integration studied in [35]. The orders of query and qubit complexities are presented in the following table.

	Deterministic Queries	Randomized Queries
Query Complexity	$\varepsilon^{-1+o(1)}$	$\varepsilon^{-1+o(1)}$
Qubit Complexity	$\varepsilon^{-2} \log \varepsilon^{-1}$	$\log \varepsilon^{-1}$

Figure 4: Path Integration

We thus have the same orders of query complexities and an exponential improvement in the number of qubits.

We stress that in the randomized classical setting and in the quantum setting with randomized queries we permit the use of random elements from a set Ω whose cardinality may be infinite and distribution of points from Ω may be arbitrary. For example, the classical Monte Carlo with n random points for integration of d-variate functions defined over, say, $[0,1]^d$, uses $\Omega = [0,1]^{dn}$ and uniform distribution. Alternatively, it is possible, also for classical computers, to use a restricted form of randomization based on, for example,

random bits or a finite set Ω. This obviously restricts the class of randomized algorithms and it is not clear if positive results for unrestrictive randomization are still true for the restricted case. There is a very interesting stream of work, see [13, 27], studying the minimal number of random bits needed for the solution of continuous problems on a classical computer. There are also general results in [12] showing that as long as Ω is finite then the classical randomized setting is (roughly) equivalent to the standard quantum setting at the expense of adding additional qubits. However, if the cardinality of Ω goes to infinity then the additional number of qubits also goes to infinity. That is why, the standard quantum setting is *not* equivalent to the classical randomized setting without a restriction on Ω.

We hope that the quantum setting with (restricted) randomized queries will be studied for general continuous problems. It would be especially interesting to characterize continuous problems for which this setting offers an exponential improvement in the number of queries and/or qubits over the standard quantum setting.

2 Continuous Problems

The computational complexity of approximate solutions of continuous problems has been studied in information-based complexity, see e.g., [7, 21, 28, 32, 33, 38]. We present a brief outline of this theory in the worst case, randomized and quantum settings to the extent needed for this paper.

Let F be a non-empty subset of a linear space of d-variate functions $f : D_d \to \mathbb{R}$ with $D_d \subset \mathbb{R}^d$. Let G be a normed space with its norm denoted by $\|\cdot\|$. Consider a (linear or non-linear) operator

$$S \,:\, F \;\to\; G.$$

Our goal is to compute $S(f)$ to within ε for $f \in F$.

Example: Multivariate Integration
We illustrate the concepts of this paper by an example of multivariate integration of smooth functions. Let $C^r([0,1]^d)$ denote the class of real functions defined on the d-dimensional unit cube, $f : [0,1]^d \to \mathbb{R}$, all of whose partial derivatives up to order r exist and are continuous. That is, for a multi-index $\alpha = [\alpha_1, \alpha_2, \ldots, \alpha_d]$ with non-negative integers α_j and with $|\alpha| := \alpha_1 + \alpha_2 + \cdots + \alpha_d \;\le\; r$ we know that

$$D^{\,\alpha} f \;=\; \frac{\partial^{\,|\alpha|}}{\partial t_1^{\alpha_1} \partial t_2^{\alpha_2} \cdots \partial t_d^{\alpha_d}}\, f$$

exists and is continuous. The norm in $C^r([0,1]^d)$ is defined as

$$\|f\|_r \;=\; \max_{\alpha:\, |\alpha| \le r} \; \max_{t \in [0,1]^d} |D^{\alpha} f\,(t)|.$$

Then we set

$$F \;=\; F_{d,r} \;=\; \{\, f \in C^r([0,1]^d) :\; \|f\|_r \;\le\; 1 \,\}$$

as the unit ball of $C^r([0,1]^d)$, and $G = \mathbb{R}$.

The multivariate integration problem $S = \mathrm{INT}_{d,r} : C^r([0,1]^d) \to \mathbb{R}$ is defined by

$$\mathrm{INT}_{d,r}(f) = \int_{[0,1]^d} f(t)\, dt.$$

This is an example of a linear problem since the operator $\mathrm{INT}_{d,r}$ depends linearly on f. □

The approximate computation of $S(f)$ can be done as follows. First of all, we specify how information about the function f is used by algorithms. We assume that we can compute finitely many function values[3] $f(t)$ for some sample points t from D_d. That is, any algorithm may use $f(t_1), f(t_2), \ldots, f(t_n)$ for some n and t_j. We stress that the choice of the sample points t_j may be *adaptive*, i.e., t_1 is given a priori, whereas t_j may depend on the already computed values $f(t_1), f(t_2), \ldots, f(t_{j-1})$ for $j = 2, 3, \ldots, n$. The number n can also be chosen adaptively. The sample points may be chosen deterministically or randomly depending on the setting, for details see [32]. The specific form of an algorithm also depends on the setting in which we define the error. We first present two settings for classical computers and then turn our attention to the quantum setting with deterministic and randomized queries.

2.1 Classical Computers: Worst Case Setting

In the worst case setting, we assume that sample points as well as algorithms are *deterministic*. That is, an algorithm that uses n function values has the form

$$A_n(f) = \phi\left(f(t_1), f(t_2), \ldots, f(t_n)\right) \tag{1}$$

for some mapping $\phi : \mathbb{R}^n \to G$. If the sample points are given a priori and are the same for all f from F, then A_n uses *non-adaptive* information. Otherwise, it uses *adaptive* information.

The worst case error of the algorithm A_n is given by its worst case performance with respect to f,

$$e^{\mathrm{wor}}(A_n) = \sup_{f \in F} \|S(f) - A_n(f)\|.$$

Example: Multivariate Integration (continued)
A typical choice of an algorithm for multivariate integration is a linear algorithm, sometimes called a quadrature or cubature,

$$A_n(f) = \sum_{j=1}^{n} a_j f(t_j)$$

for some $a_j \in \mathbb{R}$ and $t_j \in [0,1]^d$.

For $r \geq 1$, it was proven by Bakhvalov already in 1959, see [1] and also [21, 32], that the minimal worst case error of algorithms using n function values is proportional to $n^{-r/d}$. Furthermore, the error of order $n^{-r/d}$ can be achieved by a linear algorithm using non-adaptive information. Hence, if we want to guarantee that $e^{\mathrm{wor}}(A_n) \leq \varepsilon$, then n has to be of

[3] More general information given by arbitrary linear functionals on f has also been extensively studied in information-based complexity, see e.g., [32]

order $\varepsilon^{-d/r}$, and is exponential in d. This is called the *curse of dimensionality* meaning that multivariate integration is *intractable* in the worst case setting if d is much larger that r. The case $r = 0$ will be considered later.

For large d, a popular choice of a_j is n^{-1} which leads to QMC (quasi-Monte Carlo) algorithms. The sample points are often chosen as low discrepancy points, lattice or shifted lattice points, see [19, 29]. For some spaces other than $C^r([0,1]^d)$ the error behavior of such algorithms is only polynomial in d or even independent of d and tends to zero as a positive power of n^{-1}. This is an active research area of information-based complexity dealing with high dimensional problems; the reader may consult [23] for a survey. □

2.2 Classical Computers: Randomized Setting

In the randomized setting, we allow randomized choices of sample points as well as algorithms. That is, we have a probability space of elements ω from some set Ω which are distributed according to some probability measure ρ on Ω, $\rho(\Omega) = 1$. Algorithms using n function values on the average have now the form

$$A_n(f;\omega) \;=\; \phi_\omega\left(f(t_{1,\omega}), f(t_{2,\omega}), \ldots, f(t_{n_\omega,\omega})\right), \tag{2}$$

where $t_{j,\omega}$ are randomized sample points from D_d, and ϕ_ω is a randomized mapping from $\mathbb{R}^{n_\omega}$ to G. Here, n_ω is the randomized number of sample points and its average is n, i.e.,

$$n \;=\; \int_\Omega n_\omega \, \rho(d\omega).$$

We stress that the sample points $t_{j,\omega}$ as well as n_ω can be chosen adaptively as in the worst case setting. This also means that the probability measure ρ may depend on the function f through its computed function values, see Chapter 10 of [32] for details. If the sample points $t_{j,\omega}$ are the same for all f from F, then A_n uses *non-adaptive* randomized information, otherwise it uses *adaptive* randomized information.

The *randomized* error of the algorithm A_n is defined by its worst case performance with respect to f and the average performance with respect to ω,

$$e^{\mathrm{ran}}(A_n) \;=\; \sup_{f\in F}\left(\int_\Omega \|S(f) - A_n(f,\omega)\|^2 \rho(d\omega)\right)^{1/2}. \tag{3}$$

Here, we choose to study the average performance in the L_2-norm; however it is also possible to study it in a more general case of the L_p-norms with $p \in [1,\infty)$.

Example: Multivariate Integration (continued)
Probably the most popular and widely used randomized algorithm is the Monte Carlo algorithm

$$A_n(f,\omega) \;:=\; \mathrm{MC}_n(f,\omega) \;=\; \frac{1}{n}\sum_{j=1}^{n} f(t_{j,\omega}),$$

where $t_{j,\omega}$ are independent and uniformly distributed sample points over $[0,1]^d$. In this case, $\Omega = [0,1]^{dn}$ and ρ is Lebesgue's measure. That is, $\omega = [\omega_1, \omega_2, \ldots, \omega_n]$ with $\omega_j \in [0,1]^d$

and $t_{j,\omega} = \omega_j$. We stress that Monte Carlo uses non-adaptive randomized information with deterministic n and the deterministic mapping $\phi_\omega = \phi$ given by $\phi(y) = n^{-1} \sum_{j=1}^n y_j$. It is well known that

$$\int_{[0,1]^{nd}} \left(\int_{[0,1]^d} f(t)\,dt - \frac{1}{n} \sum_{j=1}^n f(\omega_j) \right)^2 d\omega_1 \cdots d\omega_d \;=\; \frac{\int_{[0,1]^d} f^2(t)\,dt \;-\; \left(\int_{[0,1]^d} f(t)\,dt \right)^2}{n}.$$

Since for $f \in F_{d,r}$ with $r \ge 0$, we have $\int_{[0,1]^d} f^2(t)dt \le 1$, then

$$e^{\rm ran}\,({\rm MC}_n) \;\le\; n^{-1/2}.$$

Hence, $e^{\rm ran}({\rm MC}_n) \le \varepsilon$ for $n = \lceil \varepsilon^{-2} \rceil$ and the curse of dimensionality of the worst case setting is broken by Monte Carlo in the randomized setting. Bakhvalov also proved, see [1], that the minimal randomized error of algorithms using n function values is of order $n^{-1/2+r/d}$, and the latter error bound is achieved by a linear algorithm using non-adaptive information. This bound also holds if we use n function values on the average as proven by Novak in [22]. Thus, Monte Carlo almost minimizes the randomized error if d is much larger than r. □

The errors of randomized algorithms may be defined differently than (3). This corresponds to the probabilistic errors which are related to the quantum setting error commonly used in many papers. To simplify the presentation of the paper, we deal with the probabilistic errors in the appendix.

2.3 Complexity and Information Complexity

As already mentioned, we want to compute $S(f)$ to within ε. That is, we are looking for an algorithm A_n whose error in the worst case or randomized setting is at most ε,

$$e^{\rm wor/ran}(A_n) \;\le\; \varepsilon. \tag{4}$$

We would like to guarantee (4) with the *minimal cost* of computing $A_n(f)$. This minimal cost is called the (total) *ε-complexity of S*, and denoted by ${\rm comp}^{\rm wor/ran}(\varepsilon, S)$. The cost of computing $y = A_n(f)$ is defined by counting the cost of n function values plus all operations needed to obtain y. The abstraction typically used in information-based complexity (and in scientific computation) is the real number model of computation in which we assume we can perform arithmetic operations and comparisons of real numbers with unit cost independently of the size of numbers, again see [32, 33] for details. The reader is referred to [31] for the motivation behind the real number model and comparison with the Turing model of computation.

As we shall see, for quantum computation the complexity of S is less relevant than the non-adaptive information complexity. The latter is defined as the minimal number n of non-adaptive function values in a given setting needed to find an algorithm $A_n^{\rm nad}$ with error at most ε. More precisely, in the worst case setting, algorithms $A_n^{\rm nad}$ are of the form (1) with a priori given sample points t_j, whereas in the randomized setting, they are of the form (2)

with sample points $t_{j,\omega}$ independent of f and depending only on ω with fixed $n_\omega = n$. Hence, the *non-adaptive information complexity* is defined by

$$\mathrm{comp}^{\mathrm{inf-wor/ran}}(\varepsilon, S) \;=\; \min\left\{\, n \;:\; \exists\, A_n^{\mathrm{nad}} \text{ such that } e^{\mathrm{wor/ran}}(A_n^{\mathrm{nad}}) \;\le\; \varepsilon \,\right\}. \tag{5}$$

We stress that the minimum in (5) is taken over *all* algorithms using non-adaptive information. That is, over all possible sample points t_j and functions ϕ in (1) in the worst case setting, and over all probability measures ρ, sample points $t_{j,\omega}$ and functions ϕ_ω in (2) in the randomized setting.

Surprisingly, for many continuous problems the non-adaptive information complexity is practically the same as the total complexity. There are, however, continuous problems for which the use of adaptive information is crucial and the non-adaptive information complexity is much larger than the total complexity, see [34] pp. 165-170. There are also continuous problems for which the reverse is true. That is, non-adaptive information complexity is small but it is impossible to combine it in a finite number of operations, and therefore the total complexity is infinite, see [36].

Example: Multivariate Integration (continued)
Bakhvalov's results mean that the total and non-adaptive information complexities of multivariate integration for the unit ball of $C^r([0,1]^d)$ are of order

$$\begin{aligned} \varepsilon^{-d/r} \quad & \text{in the worst case setting with } r \ge 1, \\ \varepsilon^{-2/(1+2r/d)} \quad & \text{in the randomized setting with } r \ge 0. \end{aligned}$$

□

3 Quantum Setting with Deterministic Queries

We describe the standard quantum setting by presenting a general form of quantum algorithms used in this setting and the definition of their errors. Quantum algorithms can be characterized, in particular, by the number of queries and qubits they use. If they use n queries and k qubits, we will denote them by $A_{n,k}$. Queries and qubits are deterministic and the only source of randomness is through measurement. We stress that quantum algorithms use *non-adaptive* information about the functions f.

The quantum algorithms $A_{n,k}$ are of the following form, see [5, 7, 21]. All computations are done on unit vectors in the complex space $\mathcal{C}^{2^k}$. Here, k denotes the number of qubits. We assume that the initial state is a unit vector $|\psi_0\rangle$ from $\mathcal{C}^{2^k}$. For $f \in F$, the final state $|\psi_f\rangle$ is equal to

$$|\psi_f\rangle \;=\; U_n Q_f\, U_{n-1} Q_f \,\cdots\, U_1 Q_f\, U_0 |\psi_0\rangle, \tag{6}$$

where $U_0, U_1, \ldots, U_n$ are $2^k \times 2^k$ unitary matrices which are independent of f. Usually it is required that each U_j is represented by a relatively small number of elementary quantum gates. This will not be important for our considerations and we permit the use of arbitrary unitary matrices U_j. Of course, this makes lower bounds on the number of needed queries and qubits stronger.

The query Q_f is also a $2^k \times 2^k$ unitary matrix and depends on the function f. We assume in this paper that Q_f is a *bit* query, see [7, 8, 9, 10, 20, 24] although results on the qubit complexity also hold for more general queries such as phase and power queries studied in [4, 26].

For a Boolean function $f : \{0, 1, \ldots, 2^m - 1\} \to \{0, 1\}$, with $k = m + 1$, the bit query is defined by

$$Q_f|j\rangle|i\rangle \;=\; |j\rangle|i \oplus f(j)\rangle$$

for all $i \in \{0, 1\}$ and $j \in \{0, 1, \ldots, 2^m - 1\}$, and $\oplus$ denotes the addition modulo 2.

For real functions $f : D_d \to \mathbb{R}$, we assume[4] that $k = m_1 + m_2$, where m_1 qubits are needed to code the arguments of f and m_2 qubits are used for the values of f. The coding is done by two mappings

$$\begin{aligned} \tau \;:\; \{0, 1, \ldots, 2^{m_1} - 1\} \quad &\to \quad D_d, \\ \beta \;:\; f(D_d) \quad &\to \quad \{0, 1, \ldots, 2^{m_2} - 1\}, \end{aligned}$$

and the bit query takes the form

$$Q_f|j\rangle|i\rangle \;=\; |j\rangle|i \oplus \beta(f(\tau(j))\rangle$$

for all $j \in \{0, 1, \ldots, 2^{m_1} - 1\}$ and $i \in \{0, 1, \ldots, 2^{m_2} - 1\}$, and $\oplus$ now means the addition modulo 2^{m_2}, for more details see [7] .

We stress that the bit query depends on at most 2^{m_1} function values computed at some non-adaptive points $t_j = \tau(j)$. Furthermore, although we will not use this fact later, these function values are usually computed with some noise due to the finite range of the coding function β. Usually, $\beta(f(t_j))$ is defined as the m_2 most significant bits of $f(t_j)$. Obviously, $2^{m_1} \leq 2^k$. If $f(D_d)$ is bounded then m_1 and m_2 are usually of the same order. To simplify further considerations we will use 2^k instead of 2^{m_1}.

For our purpose, the most important property of the bit query is that Q_f depends on at most 2^k function values taken at some a priori given (non-adaptive) *deterministic* sample points t_j from D_d,

$$Q_f \;=\; Q_{f(t_1), f(t_2), \ldots, f(t_{2^k})}. \tag{7}$$

The results on the qubit complexity will be derived using the property (7). Therefore they will be valid for *all* queries satisfying (7) which hold, in particular, for bit, phase and power queries.

The bit query Q_f is a *deterministic* $2^k \times 2^k$ unitary matrix, and therefore the final state $|\psi_f\rangle$ is also a deterministic vector of 2^k components which use n times the query Q_f based on non-adaptive information consisting of at most 2^k function values at some sample points. This means that if we consider two functions f_1 and f_2 both from F such that $f_1(t_j) = f_2(t_j)$ for $j = 1, 2, \ldots, 2^k$ then the queries Q_{f_1} and Q_{f_2} are the same, and therefore we obtain the same final states

$$|\psi_{f_1}\rangle \;=\; |\psi_{f_2}\rangle$$

for both f_1 and f_2.

[4]For simplicity we do not consider ancilla qubits.

The only source of randomness is through *measurement*. That is, we obtain an index $j \in \{0, 1, \ldots, 2^k - 1\}$ with probability $p_{f,j}$ which depends on the final state $|\psi_f\rangle$. As before, $p_{f,j}$ depends on the function f only through its values $f(t_1), f(t_2), \ldots, f(t_{2^k})$, and $\sum_{j=0}^{2^k-1} p_{f,j} = 1$. Knowing the index j, we compute on a classical computer

$$A_{n,k}(f, j) = \phi(j) \tag{8}$$

for some mapping $\phi : \{0, 1, \ldots, 2^k - 1\} \to G$. The algorithm $A_{n,k}$ is called a *quantum algorithm*.

We stress that the quantum algorithm that uses k qubits takes at most 2^k different values independently of the number n of queries used.

Remark 1: We add that, in principle, we may use hybrid algorithms that are combinations of classical algorithms using bit operations on classical computers and quantum algorithms with many measurements. However, it is known, see e.g., [3, 7], that such algorithms may be rewritten in the form (6) and (8) with one measurement at the end of computation by linearly increasing the number of queries and qubits. It is also known, see Lemma 1 in [8], that we can sample $\Gamma(f)$ instead of f if $\Gamma(f)$ depends on κ function values of f. Then the query $Q_{\Gamma(f)}$ can be simulated by a quantum algorithm that uses 2κ queries on f. Therefore, without loss of generality we may consider only quantum algorithms with one measurement of the form (6) and (8).

We stress that this is true if hybrid algorithms use only bit operations on classical computers. If we use the real number model of computation then not every algorithm can be written in the form (6) and (8). One reason of this is that we may have infinitely many outputs in the real number model of computation which is impossible to obtain in the quantum setting. □

We now discuss the error of a quantum algorithm. There are at least two natural ways to define the error. One of them is by taking the worst case performance of a quantum algorithm with respect to f and the average case performance with respect to the index j. The other is to take the worst case performance with respect to f and the worst case performance with respect to the index j modulo a set of measure δ for some (usually small) positive δ. For some problems, when S is a linear functional, it is enough to take, say, $\delta = 3/4$ and increase the probability of success by running the quantum algorithm a couple of times and by taking the median as the final result.

We will study both definitions of the error of a quantum algorithm. In the main body of the paper we choose the first option since it is directly related to the error usually studied in the randomized classical setting. The other error, which is probably more popular in the quantum literature, is called the probabilistic error and is studied in the appendix.

Hence, by the error of the quantum algorithm $A_{n,k}$ we mean

$$e^{\mathrm{qua-std}}(A_{n,k}) = \sup_{f \in F} \left(\sum_{j=0}^{2^k-1} p_{f,j} \|S(f) - A_{n,k}(f, j)\|^2 \right)^{1/2}. \tag{9}$$

This concludes the definition of the standard quantum setting which can be summarized by the general form of a quantum algorithm (6) and its error (9).

We are interested in finding quantum algorithms with error at most ε. We would like to achieve this goal with the minimal number of queries and/or qubits.

By the *query complexity in the standard quantum setting* we mean

$$\mathrm{comp}^{\mathrm{que-std}}(\varepsilon, S) \;=\; \min\left\{\, n \;:\; \exists\, A_{n,k} \text{ such that } e^{\mathrm{qua-std}}(A_{n,k}) \;\leq\; \varepsilon \,\right\}. \tag{10}$$

By "there exists $A_{n,k}$" we mean a quantum algorithm using n queries and k qubits with a finite k which can be, however, arbitrarily large. Hence, it may happen that the minimization of the number of queries will be possible at the expense of the number of qubits.

By the *qubit complexity in the standard quantum setting* we mean

$$\mathrm{comp}^{\mathrm{qub-std}}(\varepsilon, S) \;=\; \min\left\{\, k \;:\; \exists\, A_{n,k} \text{ such that } e^{\mathrm{qua-std}}(A_{n,k}) \;\leq\; \varepsilon \,\right\}. \tag{11}$$

In this case, by "there exists $A_{n,k}$" we mean an arbitrary choice of the number n of queries. Obviously, k in (10) must be at least as large as $\mathrm{comp}^{\mathrm{qub-std}}(\varepsilon, S)$, and n in (11) must be at least as large as $\mathrm{comp}^{\mathrm{que-std}}(\varepsilon, S)$.

Although we do not pursue this point in the paper, it is also reasonable to minimize both queries and qubits. For example, we may want to minimize $k\,n$ or $n+\beta\,k$, for a given positive number β, over all quantum algorithms using n queries and k qubits whose quantum error is at most ε. Here, if we choose β small then our emphasis will be on the number of queries, and if β is large then our emphasis will be on the number of qubits.

Remark 2: We stress that query and qubit complexities are defined by minimizing the number of queries/qubits needed to solve the problem by a quantum algorithm with one measurement.

Suppose we have quantum computation which requires the use of a sequence of quantum algorithms A_{n_j,k_j} each with one measurement and uses n_j queries and k_j qubits for $j = 1, 2, \ldots, p$. Then the total number of queries is $n = \sum_{j=1}^{p} n_j$ which is of the same order when all A_{n_j,k_j} are transformed as a quantum algorithm A with one measurement which uses $O(n)$ queries.

The situation is, however, different for qubits since to run all quantum algorithms A_{n_j,k_j} it is enough to have $k = \max_{j=1,2,\ldots,p} k_j$ qubits whereas the quantum algorithm A would require of order $\sum_{j=1}^{p} k_j$ qubits. Obviously, as long as p does not depend on ε, it does not really matter since k must be at least of the same order as the qubit complexity. If, however, p is large and depends on ε, then the qubit complexity needed for quantum algorithms with one measurement may be improved by many measurements.

There is one case for which the size of p does not matter. Namely, when the qubit complexity is infinite which can happen as we see in the next section.

It would be tempting to redefine the qubit complexity as the minimal number of qubits needed to solve the problem by a hybrid algorithm which performs classical and quantum operations with possible many measurements. This minimum is, however, zero since we could simulate all quantum operations on a classical computer with no qubits but at exponential cost of classical operations.

We choose to study the qubit complexity of quantum algorithms with one measurement to eliminate such a possibility. □

Example: Multivariate Integration (continued)
For the class $F = F_{d,r}$ with $r \geq 1$, the minimal error $e^{\mathrm{qua-std}}(A_{n,k})$ of quantum algorithms $A_{n,k}$ is of order $n^{-1-r/d}$ and is achieved by an algorithm that uses of order $\log \varepsilon^{-1}$ qubits. This result follows from reduction of the integration problem to real and then to Boolean summation, and from the fact that the Boolean summation algorithm of [5] using n bit queries with seven repetitions has the error for worst f and average j also proportional to n^{-1} as proved in [11]. This implies that

$$\mathrm{comp}^{\mathrm{que-std}}(\varepsilon, S) \;=\; \Theta\left(\varepsilon^{-1/(1+r/d)}\right) \quad \text{and} \quad \mathrm{comp}^{\mathrm{qub-std}}(\varepsilon, S) \;=\; O\left(\log \varepsilon^{-1}\right). \tag{12}$$

□

3.1 Lower Bounds on Qubit Complexity

We now prove lower bounds on the qubit complexity in the standard quantum setting in terms of the non-adaptive information complexity in the worst case setting as well as in terms of the (Kolmogorov) ε-entropy of the set $S(F)$. Based on these bounds, we conclude that some continuous problems S *cannot* be solved in the standard quantum setting.

Theorem 3.1.

$$\mathrm{comp}^{\mathrm{qub-std}}(\varepsilon, S) \;\geq\; \log \mathrm{comp}^{\mathrm{inf-wor}}(2\varepsilon, S).$$

Proof.
Take an arbitrary quantum algorithm $A_{n,k}$ such that $e^{\mathrm{qua-std}}(A_{n,k}) \leq \varepsilon$ with the minimal number of qubits $k = \mathrm{comp}^{\mathrm{qub-std}}(\varepsilon, S)$. We have

$$\varepsilon^2 \;\geq\; e^{\mathrm{qua-std}}(A_{n,k})^2 \;=\; \sup_{f \in F} \sum_{j=0}^{2^k-1} p_{f,j} \|S(f) - A_{n,k}(f,j)\|^2.$$

The final state as well as probabilities $p_{f,j}$ of the quantum algorithm $A_{n,k}$ are based on the non-adaptive information

$$N(f) \;=\; [f(t_1), f(t_2), \ldots, f(t_{2^k})]$$

for some sample points $t_j \in D_d$. Therefore we can write

$$A_{n,k}(f,j) \;=\; \Phi(f(t_1), f(t_2), \ldots, f(t_{2^k}); j) \qquad \forall\, f \in F,\ \forall\, j \in \{0, 1, \ldots, 2^k - 1\},$$

for some mapping $\Phi : \mathbb{R}^{2^k} \times \{0, 1, \ldots, 2^k - 1\} \to G$.

For an arbitrary $f \in F$, take two functions f_1 and f_2 such that $N(f_1) = N(f_2) = N(f)$. The final state as well as all probabilities $p_{f,j}$ will be the same for f_1 and f_2, and $a_j = A_{n,k}(f_1, j) = A_{n,k}(f_2, j)$ for all j. Hence, for any $f \in F$, we have

$$2\varepsilon^2 \;\geq\; \sum_{j=0}^{2^k-1} p_{f,j} \left(\|S(f_1) - a_j\|^2 + \|S(f_2) - a_j\|^2\right).$$

Observe that

$$\|S(f_1) - S(f_2)\|^2 \leq (\|S(f_1) - a_j\| + \|S(f_2) - a_j\|)^2 \leq 2\left(\|S(f_1) - a_j\|^2 + \|S(f_2) - a_j\|^2\right).$$

Multiplying both sides by $p_{f,j}$ and summing up with respect to j, we conclude

$$\|S(f_1) - S(f_2)\|^2 \leq 4\varepsilon^2.$$

Taking the supremum with respect to $f \in F$ and f_1, f_2 from F with $N(f_1) = N(f_2)$ we have

$$\sup_{f \in F} \sup_{f_1, f_2 \in F,\, N(f_1)=N(f_2)=N(f)} \|S(f_1) - S(f_2)\| \leq 2\varepsilon.$$

The left-hand side of the last inequality is equal to the diameter of information N, see [32] p. 45, which in turn is bounded from below by the radius of information, denoted by $\mathrm{rad}(N)$. Hence, $\mathrm{rad}(N) \leq 2\varepsilon$ which can hold only if the cardinality of N is at least equal to $\mathrm{comp}^{\mathrm{inf-wor}}(2\varepsilon, S)$, see [32] p. 54. Thus, $2^k \geq \mathrm{comp}^{\mathrm{inf-wor}}(2\varepsilon, S)$, as claimed.

We add in passing that for many cases we have $d(N) = 2\mathrm{rad}(N)$. This holds, in particular, if $G = \mathbb{R}$. Then $\mathrm{rad}(N) \leq \varepsilon$ and the extra factor 2 can be omitted. □

Theorem 3.1 states that the number of qubits needed to solve S in the standard quantum setting is related to the non-adaptive information complexity in the worst case setting. For most continuous problems S, the non-adaptive information complexity goes to infinity as ε approaches zero. Then Theorem 3.1 says that the number of qubits also goes to infinity although much more slower due to the presence of the logarithm. We illustrate this point by continuing our example.

Example: Multivariate Integration (continued)
Consider $F = F_{d,r}$ with $r \geq 1$. We know that

$$\mathrm{comp}^{\mathrm{inf-wor}}(2\varepsilon, S) = \Theta\left(\varepsilon^{-d/r}\right).$$

Then Theorem 3.1 supplies a lower bound on the qubit complexity,

$$\mathrm{comp}^{\mathrm{qub-std}}(\varepsilon, \mathrm{INT}_{d,r}) \geq \tfrac{d}{r} \log \varepsilon^{-1} + \Omega(1)$$

with the term $\Omega(1)$ independent of ε but dependent on d and r.

As we already discussed, $\frac{d}{r}$ was the exponent of the worst case complexity of the integration problem $\mathrm{INT}_{d,r}$ and caused the curse of dimensionality. Its role for the qubit complexity is mitigated since it effects the lower bound on the qubit complexity only linearly.

Due to (12) the lower bound on the qubit complexity is sharp with respect to ε, and we have

$$\mathrm{comp}^{\mathrm{qub-std}}(\varepsilon, \mathrm{INT}_{d,r}) = \Theta\left(\log \varepsilon^{-1}\right).$$

The dependence on ε is very weak although for ε tending to zero, the qubit complexity slowly goes to infinity. □

Remark 3: Theorem 3.1 was formally proved for the class of quantum algorithms with one measurement. We now show that a similar result holds for the much more larger class of hybrid algorithms which use non-adaptive or adaptive function values on a classical computer and many measurements on a quantum computer. More precisely, consider the following class of hybrid algorithms:

- For $i = 1, 2, \ldots, p$ do
  - Use a classical algorithm with m_i non-adaptive or adaptive function values to get an initial state $|\psi_{0,i}\rangle$,
  - Use a quantum algorithm A_{n_i,k_i} with one measurement starting with the initial state $|\psi_{0,i}\rangle$ and with n_i bit queries and k_i qubits, and let

    $$A_{n_i,k_i}(f,j) \;=\; \phi_i(j)$$

    with a function ϕ_i which can now be dependent on ℓ_i non-adaptive or adaptive function values.

Observe that the total number of function values used by the hybrid algorithm is at most

$$N \;:=\; \sum_{i=1}^{p} \left(m_i + 2^{k_i} + \ell_i\right),$$

and up to $\sum_{i=1}^{p}(m_i + \ell_i)$ of them can be computed adaptively. The hybrid algorithm uses k qubits, where

$$k \;=\; \max_{i=1,2,\ldots,p} k_i.$$

Assume that the error of the hybrid algorithm is ε. Then as in the proof of Theorem 3.1 we conclude

$$N \;\geq \mathrm{comp}^{\mathrm{inf-wor}}(2\varepsilon, S),$$

where now $\mathrm{comp}^{\mathrm{inf-wor}}(2\varepsilon, S)$ stands for the worst case (adaptive) information complexity defined as in (5) with the exception that now A_n^{nad} is replaced by an arbitrary algorithm A_n using at most n adaptive function values.

Hence, as long as there are two positive numbers a_1 and a_2 such that

$$N \;\leq\; a_1\, 2^{\,a_2\, k}$$

then

$$k \;\geq\; a_2^{-1} \log \mathrm{comp}^{\mathrm{inf-wor}}(2\varepsilon, S) \;-\; \log\, a_1.$$

Hence, even for hybrid algorithms, the logarithm of the worst case (adaptive) information complexity is essential and tells us how many qubits are needed. □

We now consider the case when $\mathrm{comp}^{\mathrm{inf-wor}}(\varepsilon, S) = \infty$, i.e., when we cannot solve the problem in the worst case setting. Then the qubit complexity is also infinite. We summarize this fact in the following corollary.

Corollary 3.1. *If the non-adaptive information complexity of S in the worst case setting is infinity then S cannot be solved in the standard quantum setting.*

We illustrate Corollary 3.1 by multivariate integration for $r = 0$.

Example: Multivariate Integration (continued)
Assume now that $r = 0$. Hence, $F = F_{d,0}$ is the unit ball of continuous functions with the

norm $\|f\|_0 = \max_{x\in[0,1]^d} |f(t)|$ bounded by one. It is known, and easy to see, that for any algorithm A_n we have

$$e^{\rm wor}(A_n) \,\geq\, 1 \qquad \forall\, n.$$

Indeed, as already explained in the introduction, it is enough to take two continuous functions from F vanishing at the sample points t_j used by the algorithm A_n, $j = 1, 2, \ldots, n$, such that the integral of the first function is almost 1, and the integral of the other function is almost -1. Since these functions are indistinguishable for the algorithm A_n, the best we can do is to approximate their integrals by zero with error arbitrarily close to one. Hence, the worst case error of any algorithm is at least one, as claimed. This implies that

$$\mathrm{comp}^{\rm inf-wor}(\varepsilon, S) \,=\, \infty \qquad \forall\, \varepsilon < 1.$$

Theorem 3.1 says that multivariate integration for $r = 0$ is *unsolvable* in the standard quantum setting.

As already mentioned, this problem is, however, easy in the randomized setting. The randomized error of the Monte Carlo is bounded by $n^{-1/2}$ which is optimal due to lower bounds of Bakhvalov and Novak. Therefore the non-adaptive randomized information complexity as well as the total randomized complexity are both of order $\varepsilon^{-1/2}$.

So why does the standard quantum setting fail for the problem which is relatively easy in the randomized setting? As we shall see in the next section, the reason is that we use *deterministic* queries in the standard quantum setting. This bad property will disappear if we allow the use of *randomized* queries also in the quantum setting. □

Before we proceed to the quantum setting with randomized queries, we briefly present another lower bound on the qubit complexity in the standard quantum setting. This bound relates the qubit complexity to the (Kolmogorov) ε-entropy of the set $S(F)$. We first recall the notion of ε-entropy in normed spaces, see e.g., [17]. Let B be a subset of G. We want to cover the subset B by the minimal number of subsets of G whose diameters do not exceed 2ε. That is, let

$$n(\varepsilon, B) \,=\, \min\left\{\, n \,:\, \exists\, B_j \subset G \text{ such that } \mathrm{diam}(B_j) \leq 2\varepsilon,\ B \subset \cup_{j=1}^{n} B_j \,\right\},$$

where $\mathrm{diam}(B_j) = \sup_{b_1, b_2 \in B_j} \|b_1 - b_2\|$. Then the ε-entropy of B is

$$\mathrm{Ent}(\varepsilon, B) \,=\, \log\, n(\varepsilon, B).$$

It is easy to prove the following theorem.

Theorem 3.2.

$$\mathrm{comp}^{\rm qub-std}(\varepsilon, S) \,\geq\, \mathrm{Ent}(\varepsilon, S(F)).$$

Proof.
The proof relies on the fact that in the standard quantum setting any quantum algorithm which uses $k = \mathrm{comp}^{\rm qub-std}(\varepsilon, S)$ qubits produces at most 2^k different elements $A_{n,k}(f, j) = \phi(j)$ from G for $j = 0, 1, 2, \ldots, 2^k - 1$ with ϕ independent of f.

We take an arbitrary quantum algorithm $A_{n,k}$ with error $e^{\mathrm{qua-std}}(A_{n,k}) \leq \varepsilon$. For any $f \in F$, we have

$$\min_{j=0,1,\ldots,2^k-1} \|S(f) - \phi(j)\| \;\leq\; \left(\sum_{j=0}^{2^k-1} p_{f,j} \|S(f) - \phi(j)\|^2 \right)^{1/2} \;\leq\; \varepsilon.$$

Let $B(\phi(j), \varepsilon) = \{ g \in G \,:\, \|g - \phi(j)\| \leq \varepsilon \}$ be the ball in G of center $\phi(j)$ and radius ε. Obviously, $\mathrm{diam}(B(\phi(j), \varepsilon)) \leq 2\varepsilon$. Then $S(f) \in \cup_{j=0}^{2^k-1} B(\phi(j), \varepsilon)$ and therefore $S(F) \subset \cup_{j=0}^{2^k-1} B(\phi(j), \varepsilon)$. This means that $2^k \geq n(\varepsilon, S(F))$, and $k \geq \log\, n(\varepsilon, S(F))$, as claimed. □

The essence of Theorem 3.2 is that $S(F)$ must have a finite ε-entropy in order to have S solvable in the standard quantum setting. In particular, this means that the closure of $S(F)$ must be compact. Otherwise, the ε-entropy is infinite and the finite number of qubits is not enough to solve S. We summarize this in the following corollary.

Corollary 3.2. *If the closure of $S(F)$ is not compact then S is not solvable in the standard quantum setting. In particular, if $S(F)$ is unbounded then S is not solvable in the standard quantum setting.*

The unboundedness of $S(F)$ can happen even for problems with relatively small worst case complexity as shown in the following example. This example also shows that lower bounds based on the ε-entropy of $S(F)$ presented in Theorem 3.2 may be quite different than lower bounds based on the non-adaptive information complexity in the worst case setting presented in Theorem 3.1.

Example: Unbounded $S(F)$
Consider the univariate integration problem for Lipschitz functions, i.e.,

$$F \;=\; \{\, f : [0,1] \to \mathbb{R} |\;\; |f(x) - f(y)| \leq |x - y| \;\; \forall\, x, y \in [0,1] \,\},$$

and $S(f) = \int_0^1 f(t)\, dt$ with $G = \mathbb{R}$.

Since all constant functions belong to F, we have $S(F) = \mathbb{R}$ and therefore $\mathrm{Ent}(\varepsilon, \mathbb{R}) = \infty$. It is well known that the worst case complexity is roughly $1/(4\varepsilon)$, see e.g., [33], and the linear algorithm

$$A_n(f) \;=\; \frac{1}{2n} f\left(\frac{1}{2n}\right) \;+\; \frac{1}{n} \sum_{j=2}^{n-1} f\left(\frac{2j-1}{2n}\right) \;+\; \frac{1}{2n} f\left(\frac{2n-1}{2n}\right)$$

with $n = \lceil \varepsilon^{-1}/4 \rceil$ minimizes the worst case error among all algorithms using n function values, and has error at most ε. Observe that for constant functions, $f(t) \equiv c$, we have $A_n(f) = c$ which may be arbitrary large.

In the worst case setting with the real number model, the sizes of numbers do not matter and do not affect the cost analysis. In the standard quantum setting, the situation is different since we can only work on unit vectors and the scaling of numbers *does* matter. That is why we cannot solve unscaled problems in the standard quantum setting.

In many cases, we may rescale the problem by changing F to a set $\tilde{F}$ such that $S(\tilde{F})$ is bounded and its closure is compact. This idea works for our example as follows. For

$f \in F$, define $g(x) = f(x) - f(0)$ for $x \in [0,1]$. Then $g(0) = 0$ and $|g(x)| \le x \le 1$. Hence, $-x \le g(x) \le x$ and therefore $S(g) \in [-\frac{1}{2}, \frac{1}{2}]$, and both bounds are sharp. Define

$$\widetilde{F} \;=\; \{g : [0,1] \to \mathbb{R} |\; g(0) = 0,\; g \in F\}.$$

Then $f \in F$ iff $g \in \widetilde{F}$ for $g(x) = f(x) - f(0)$, and $S(f) = S(g) - f(0)$.

We now have $S(\widetilde{F}) = [-\frac{1}{2}, \frac{1}{2}]$ and therefore

$$\mathrm{Ent}(\varepsilon, S(\widetilde{F})) \;=\; \log \varepsilon^{-1} + O(1).$$

Hence, the number of qubits for approximations of $S(g)$ is now bounded by roughly $\log \varepsilon^{-1}$. In fact it is sharp, since $S(g)$ can be approximated to within ε in the standard quantum setting by $A_{n,k}(q)$ using roughly $n = \varepsilon^{-1/2}$ bit queries and $k = \log \varepsilon^{-1}$ qubits as shown in [24].

Finally, we may approximate $S(f)$ for $f \in F$ by running $A_{n,k}(g)$ for $g(x) = f(x) - f(0)$, and computing $f(0) + A_{n,k}(q)$ on a classical computer. Note that the last step on a classical computer may involve an arbitrarily large number $f(0)$ which is of no relevance as long as we use the real number model of computation. □

4 Quantum Setting with Randomized Queries

Modulo measurements, the standard quantum setting for continuous problems is similar to the worst case setting with non-adaptive information. All unitary matrices including queries, as well as the number of qubits used by quantum algorithms are deterministic. The potential speedup of the standard quantum setting for continuous problems over the worst case setting relies on the fact that quantum algorithms with k qubits may use an exponential number up to 2^k function values with cost proportional to a small power of k. If 2^k functions values are not enough to solve the problem in the worst case setting then the problem remains unsolvable also in the standard quantum setting. As we indicated before, this may happen even for problems with small randomized complexity. Such examples suggest studying more general quantum settings.

In this section, we describe the quantum setting with *randomized queries* in which all unitary matrices including queries may be randomized. Modulo measurements, the quantum setting with randomized queries for continuous problems will be similar to the randomized setting with non-adaptive information. We assume that the number of qubits is fixed and does not depend on randomization. As we already mentioned in the introduction the extension to randomized qubits is left for future study.

We generalize (6) by allowing unitary matrices U_j as well as the query Q_f to be randomly chosen similarly as in the randomized classical setting of Section 2.2. That is, we have random elements ω distributed accordingly to some probability measure ρ on Ω with $\rho(\Omega) = 1$. We stress that ρ does not depend on f and we will be using the same randomization for all f from F.

First we choose k as the number of qubits, take a random element ω, choose a unit vector $|\psi_{0,\omega}\rangle$ from $\mathcal{C}^{2^k}$ as the initial state, and obtain the final k qubit state

$$|\psi_{f,\omega}\rangle \;=\; U_{n_\omega,\omega} Q_{f,\omega}\, U_{n_\omega-1,\omega} Q_{f,\omega} \cdots U_{1,\omega} Q_{f,\omega}\, U_{0,\omega} |\psi_{0,\omega}\rangle. \tag{13}$$

For a fixed ω, we have the same situation as in the standard quantum setting. That is, matrices $U_{j,\omega}$ are arbitrary $2^k \times 2^k$ unitary matrices which are independent of f, and the query $Q_{f,\omega}$, which is also a $2^k \times 2^k$ unitary matrix, depends on at most 2^k sample points which are independent of f and depend only on ω,

$$Q_{f,\omega} \;=\; Q_{f(t_{1,\omega}),f(t_{2,\omega}),\ldots,f(t_{2^k,\omega})}.$$

Hence, $Q_{f,\omega}$ is a randomized query depending on at most 2^k function values at randomized sample points. In full analogy with the standard quantum setting, the measure ρ and sample points $t_{j,\omega}$ are the same for all f from F. That is, we use *non-adaptive* randomized information with fixed cardinality at most 2^k. Let

$$n \;=\; \int_\Omega n_\omega \, \rho(d\omega)$$

be the average number of queries used to obtain the final states.

We then perform a measurement which for a fixed ω is the same as for the standard quantum setting. That is, we obtain an index $j \in \{0, 1, \ldots, 2^k - 1\}$ with probability $p_{f,j,\omega}$ depending on the final state $|\psi_{f,\omega}\rangle$, where $\sum_{j=0}^{2^k-1} p_{f,j,\omega} = 1$ for all $\omega \in \Omega$. As before, the dependence on f is only through function values,

$$p_{f,j,\omega} \;=\; p_{f(t_{1,\omega}),f(t_{2,\omega}),\ldots,f(t_{2^k},\omega),j,\omega}.$$

Knowing the index j, we compute on a classical computer

$$A_{n,k}(f,\omega,j) \;=\; \phi_\omega(j)$$

for some mapping $\phi_\omega : \{0, 1, \ldots, 2^k - 1\} \;\to\; G$. The algorithm $A_{n,k}$ is called a *quantum algorithm using randomized queries*, or just a quantum algorithm if it is clear from the context that we are using randomized queries.

Analogously to the standard quantum setting, we consider the error of a quantum algorithm by taking the average performance with respect to both j and ω, see also the appendix where the probabilistic error is discussed. That is, the *error* of an algorithm in the *quantum setting with randomized queries* $A_{n,k}$ is defined by

$$e^{\mathrm{qua-ran}}(A_{n,k}) \;=\; \sup_{f\in F} \left(\int_\Omega \sum_{j=0}^{2^k-1} p_{f,j,\omega} \|S(f) - A_{n,k}(f,\omega,j)\|^2 \rho(d\omega) \right)^{1/2}.$$

Observe that if we choose all matrices $U_{j,\omega}$ and $Q_{f,\omega}$ as well as n_ω independently of ω, then this definition coincides with the error in the standard quantum setting.

This definition of the error leads to the *randomized query complexity* defined by

$$\mathrm{comp}^{\mathrm{que-ran}}(\varepsilon, S) \;=\; \min\left\{\, n \;:\; \exists\, A_{n,k} \text{ such that } e^{\mathrm{qua-ran}}(A_{n,k}) \;\le\; \varepsilon \,\right\},$$

and to the *randomized qubit complexity* defined by

$$\mathrm{comp}^{\mathrm{qub-ran}}(\varepsilon, S) \;=\; \min\left\{\, k \;:\; \exists\, A_{n,k} \text{ such that } e^{\mathrm{qua-ran}}(A_{n,k}) \;\le\; \varepsilon \,\right\}.$$

As in the standard quantum setting, we may have a tradeoff between the minimal number of queries and qubits. Therefore it would be also reasonable to study the minimization of the product or a weighted sum of queries and qubits in the quantum setting with randomized queries, however, we do not pursue the issue in this paper.

It is natural to ask what kind of results can be now achieved and how much we can improve the results from the standard quantum setting. We will study these questions in the next sections.

4.1 Lower Bounds on Randomized Qubit Complexity

For the standard quantum setting, we proved lower bounds on qubit complexity in terms of the worst case setting on a classical computer. We now show that lower bounds on the randomized qubit complexity can be analogously derived in terms of the randomized setting on a classical computer.

Theorem 4.1.

$$\mathrm{comp}^{\mathrm{qub-ran}}(\varepsilon, S) \;\geq\; \log \mathrm{comp}^{\mathrm{inf-ran}}(\varepsilon, S).$$

Proof.
We note that any quantum algorithm $A_{n,k}$ can be regarded as a randomized algorithm whose cardinality is at most 2^k. Indeed, for $f \in F$, let $\bar{\rho}_f$ be a probability measure defined on $B \times J$, where B is a measurable subset of Ω and J is an arbitrary subset of $\{0, 1, \ldots, 2^k - 1\}$ given by

$$\bar{\rho}_f(B \times J) \;=\; \int_\Omega 1_B(\omega) \sum_{j \in J} p_{f,j,\omega}\, \rho(d\omega)$$

with the characteristic function $1_B(\omega) = 1$ for $\omega \in B$ and $1_B(\omega) = 0$ otherwise. Then

$$\begin{aligned} e^{\mathrm{qua-ran}}(A_{n,k})^2 &= \sup_{f \in F} \int_\Omega \sum_{j=0}^{2^k-1} p_{f,j,\omega} \|S(f) - A_{n,k}(f, \omega, j)\|^2 \rho(d\omega) \\ &= \sup_{f \in F} \int_{\Omega \times \{0,1,\ldots,2^k-1\}} \|S(f) - A_{n,k}(f, \omega, j)\|^2 \bar{\rho}_f(d(\omega, j)) \;=\; e^{\mathrm{ran}}(A_{n,k})^2. \end{aligned}$$

Hence, $A_{n,k}$ can be regarded as a randomized algorithm in the randomized classical setting whose cardinality is at most 2^k. Since the sample points $t_{j,\omega}$ are independent of f and dependent only on ω, the algorithm $A_{n,k}$ uses non-adaptive information, and $A_{n,k}$ applied to f uses randomization with the measure $\bar{\rho}_f$. If $e^{\mathrm{qua-ran}}(A_{n,k}) \leq \varepsilon$ then $e^{\mathrm{ran}}(A_{n,k}) \leq \varepsilon$ which may happen only if the cardinality of $A_{n,k}$ is at least $\mathrm{comp}^{\mathrm{inf-ran}}(\varepsilon, S)$. This means that $2^k \geq \mathrm{comp}^{\mathrm{inf-ran}}(\varepsilon, S)$, as claimed. □

Note that the lower bounds in Theorem 4.1 are not larger than the lower bounds in Theorem 3.1 since $\mathrm{comp}^{\mathrm{inf-ran}}(\varepsilon, S) \leq \mathrm{comp}^{\mathrm{inf-wor}}(\varepsilon, S)$. Furthermore for some problems the non-adaptive information complexity in the worst case setting may be infinite whereas its randomized counterpart is relatively small.

We already mentioned that the integration problem for the class $F = F_{d,0}$ is unsolvable in the standard quantum setting and solvable in the randomized classical setting. We now provide a proof and find the randomized qubit complexity. The randomized query complexity for $F_{d,0}$ as well as the integration problem for $F_{d,r}$ for an arbitrary integer $r \geq 0$ will be studied later.

Example: Multivariate Integration for $r = 0$ (continued)
We show that the randomized qubit complexity for $F = F_{d,0}$ is

$$\mathrm{comp}^{\mathrm{qub-ran}}(\varepsilon, \mathrm{INT}_{d,0}) \;=\; \Theta\left(\log \varepsilon^{-1}\right).$$

and is achieved by a quantum algorithm which uses of order ε^{-1} queries.

We know that $\mathrm{comp}^{\mathrm{inf-ran}}(\varepsilon, \mathrm{INT}_{d,0}) = \Theta(\varepsilon^{-2})$. From Theorem 4.1 we conclude that the randomized qubit complexity must be at least of order $\log \varepsilon^{-1}$.

We now provide an upper bound on the randomized qubit complexity. Take the Monte Carlo algorithm with $m = \lceil 4\varepsilon^{-2} \rceil$,

$$\mathrm{MC}_m(f, \omega) \;=\; \frac{1}{m} \sum_{j=1}^{m} f(\omega_j)$$

with $\omega = [\omega_1, \omega_2, \ldots, \omega_m]$ and independent uniformly distributed ω_j from $[0,1]^d$. Then $e^{\mathrm{ran}}(\mathrm{MC}_m) \leq \varepsilon/2$.

We now apply the Boolean summation algorithm $A^*_{n,k}$ of [5] with seven repetitions for real functions for which $|f(x)| \leq 1$, see also Section 5. The algorithm $A^*_{n,k}$ approximates $\mathrm{MC}_m(f)$. It is known, see [11], that the randomized error of this algorithm is bounded by C/n, where C is a number independent on f, n and m. Furthermore, the algorithm uses $k = \Theta(\log m)$ qubits. We set $n = \lceil 2\,C/\varepsilon \rceil$ and obtain

$$\sum_{j=0}^{2^k-1} p_{f,j,\omega} \left[\mathrm{MC}_m(f,\omega) - A^*_{n,k}(f,\omega,j)\right]^2 \;\leq\; \frac{\varepsilon^2}{4} \qquad \forall\, f \in F.$$

Therefore for any $f \in F$ we have

$$\sum_{j=0}^{2^k-1} p_{f,j,\omega} \left[\mathrm{INT}_{d,0}(f) - A^*_{n,k}(f,\omega,j)\right]^2 \;=$$

$$\sum_{j=0}^{2^k-1} p_{f,j,\omega} \left[\mathrm{INT}_{d,0}(f) - \mathrm{MC}_m(f,\omega) + \mathrm{MC}_m(f,\omega) - A^*_{n,k}(f,\omega,j)\right]^2 \;\leq$$

$$2 \sum_{j=0}^{2^k-1} p_{f,j,\omega} \left[\left(\mathrm{INT}_{d,0}(f) - \mathrm{MC}_m(f,\omega)\right)^2 \;+\; \left(\mathrm{MC}_m(f,\omega) - A^*_{n,k}(f,\omega,j)\right)^2 \right] \;\leq$$

$$2 \left[\mathrm{INT}_{d,0}(f) - \mathrm{MC}_m(f,\omega)\right]^2 \;+\; \frac{\varepsilon^2}{2}.$$

Taking the integral over Ω we conclude that

$$e^{\mathrm{ran}}(A^*_{n,k}) \;\leq\; \sqrt{2\frac{\varepsilon^2}{4} + \frac{\varepsilon^2}{2}} \;=\; \varepsilon.$$

Hence, we can solve the integration problem for $r = 0$ using of order $\log \varepsilon^{-1}$ qubits and ε^{-1} randomized queries, as claimed. □

Theorem 4.1 states that the number of qubits in the quantum setting with randomized queries depends on the non-adaptive information complexity $\mathrm{comp}^{\mathrm{inf-ran}}(\varepsilon, S)$. Typically, $\mathrm{comp}^{\mathrm{inf-ran}}(\varepsilon, S)$ goes to infinity with ε tending to zero, so the number of qubits has to go to infinity as well although much slower due to the presence of the logarithm in the bound of Theorem 4.1. However, if $\mathrm{comp}^{\mathrm{inf-ran}}(\varepsilon, S) = \infty$ then the randomized qubit complexity is infinity and the problem cannot be solved. We summarize this fact in the following corollary.

Corollary 4.1. *If the non-adaptive information complexity of S in the randomized setting is infinity then S cannot be solved in the quantum setting with randomized queries.*

We illustrate Corollary 4.1 by a problem with infinite non-adaptive information complexity in the quantum setting with randomized queries.

Example: Multivariate Approximation
Consider the same class $F = F_{d,r}$, $r \geq 0$, as for the multivariate integration problem. Let $G = C([0,1]^d)$ and $S = \mathrm{APP}_{d,r} : C^r([0,1]^d) \to G$ be defined by

$$\mathrm{APP}_{d,r}(f) \;=\; f.$$

It is known, see [32] p. 425, that randomization does not help for this problem and that for small ε we have

$$\mathrm{comp}^{\mathrm{inf-ran}}(\varepsilon, \mathrm{APP}_{d,0}) \;=\; \mathrm{comp}^{\mathrm{inf-wor}}(\varepsilon, \mathrm{APP}_{d,0}) \;=\; \infty,$$

and for $r \geq 1$,

$$\mathrm{comp}^{\mathrm{inf-ran}}(\varepsilon, \mathrm{APP}_{d,r}) \;=\; \Theta\left(\mathrm{comp}^{\mathrm{inf-wor}}(\varepsilon, \mathrm{APP}_{d,r})\right) \;=\; \Theta\left(\varepsilon^{-d/r}\right).$$

Hence, for $r = 0$ we conclude that the approximation problem cannot be solved in the quantum setting with randomized queries. □

5 Boolean and Real Summation

Solution of the real summation problem is a basic module used in the solution of many continuous problems. The real summation problem is very much related to the Boolean summation problem which has been thoroughly studied in the standard quantum setting, see [5, 7, 11, 16, 18]. In this section we study the Boolean and real summation problems in the quantum setting with randomized queries.

5.1 Boolean Summation

For N a (large) power of two, consider the class of Boolean functions

$$F \;=\; F_N \;=\; \{f \mid \;\; f : \{0,1,\ldots,N-1\} \;\to\; \{0,1\}\;\},$$

and take $G = \mathbb{R}$. The Boolean summation problem is defined as $S = \mathcal{B}_N : F_N \to G$ given by

$$\mathcal{B}_N(f) \;=\; \frac{1}{n}\sum_{j=0}^{N-1} f(j).$$

We want to approximate $\mathcal{B}_N$ to within ε. Without loss of generality we may assume that $\varepsilon \geq 1/(2N)$. Indeed, if we know $A_{n,k}(f)$ such that $|\mathcal{B}_N(f) - A_{n,k}(f)| \leq \varepsilon < 1/(2N)$ then we can recover $\mathcal{B}_N(f)$ exactly since

$$\mathcal{B}_N(f) \;=\; \frac{\lceil N\, A_{n,k}(f) + \frac{1}{2}\rceil \;-\; 1}{N}.$$

This follows from the fact that we know a priori that $\mathcal{B}_N(f) = k/N$ for some integer $k \in [0, N]$ with k being the total number of the true assignments of the Boolean function f. Then $NA_{n,k}(f) + \frac{1}{2} = k + x$ with

$$x \;=\; \tfrac{1}{2} + N\,(A_{n,k}(f) - \mathcal{B}_N) \;\in\; \left[\tfrac{1}{2} - N\varepsilon, \tfrac{1}{2} + N\varepsilon\right] \;\subset\; (0,1).$$

Hence, $\lceil k + x\rceil = k + 1$, as claimed.

We first consider the Boolean summation problem in the standard quantum setting. The Boolean summation algorithm $A^*_{n,k}$ of [5] with seven repetitions solves the Boolean summation such that

$$e^{\mathrm{qua-std}}(A_{n,k}) \;\leq\; \varepsilon,$$

using $n = \Theta(\varepsilon^{-1})$ bit queries and $k = \Theta(\log\, N)$ qubits. The query bound is order-optimal, see [11, 18]. The qubit bound is also order-optimal since it is known, see e.g., [24], that in the worst case setting

$$\mathrm{comp}^{\mathrm{inf-wor}}(\varepsilon, S) \;=\; \lceil N(1-2\varepsilon)\rceil \qquad \forall\, \varepsilon \in \left[0, \tfrac{1}{2}\right]$$

which is essentially N for small ε. From Theorem 3.1 we conclude that the qubit complexity is roughly at least log N for small ε. We summarize these results in the following theorem.

Theorem 5.1. *The complexities of the Boolean summation problem in the standard quantum setting satisfy*

$$\mathrm{comp}^{\mathrm{que-std}}(\mathcal{B}_N) \;=\; \Theta\left(\varepsilon^{-1}\right) \qquad \mathrm{comp}^{\mathrm{qub-std}}(\mathcal{B}_N) \;=\; \Theta\left(\log\, N\right).$$

Furthermore, these bounds are both attained by the Boolean summation algorithm with seven repetitions.

We now consider the Boolean summation problem in the quantum setting with randomized queries. We prove the following theorem.

Theorem 5.2. *The complexities of the Boolean summation problem in the quantum setting with randomized queries satisfy*

$$\mathrm{comp}^{\mathrm{que-ran}}(\mathcal{B}_N) \,=\, \Theta\left(\varepsilon^{-1}\right) \qquad \mathrm{comp}^{\mathrm{qub-ran}}(\mathcal{B}_N) \,=\, \Theta\left(\log\,\varepsilon^{-1}\right).$$

Furthermore, these bounds are both attained by the Boolean summation algorithm with seven repetitions applied to

$$\frac{1}{m}\sum_{j=1}^{m} f(\omega_j)$$

with $m = \Theta(\varepsilon^{-2})$ and with independent uniformly distributed ω_j from $\{0, 1, \ldots, N-1\}$.

Proof.
We first consider lower bounds and start with the randomized query complexity. We use the known proof technique of using the average case error as a lower estimate of the randomized error. More precisely, take an arbitrary quantum algorithm $A_{n,k}$ that uses n randomized bit queries and k qubits, and consider its randomized error

$$e^{\mathrm{qua-ran}}(A_{n,k})^2 \,=\, \sup_{f\in F_N} \int_\Omega \sum_{j=0}^{2^k-1} p_{f,j,\omega} |\mathcal{B}_N(f) - A_{n,k}(f,\omega,j)|^2 \rho(d\omega).$$

We now replace the supremum over f by an average over f. That is, we assume that a Boolean function f from F_N occurs with probability p_f with non-negative p_f such that $\sum_{f\in F_N} p_f = 1$. Observe that this is a well defined measure since F_N consists of finitely many Boolean functions, in fact, we have 2^N functions in F_N. Then

$$\begin{aligned} e^{\mathrm{qua-ran}}(A_{n,k})^2 \;&\geq\; \sum_{f\in F_N} p_f \int_\Omega \sum_{j=0}^{2^k-1} p_{f,j,\omega} |\mathcal{B}_N(f) - A_{n,k}(f,\omega,j)|^2 \rho(d\omega) \\ &=\; \int_\Omega \left(\sum_{f\in F_N} p_f \sum_{j=0}^{2^k-1} p_{f,j,\omega} |\mathcal{B}_N(f) - A_{n,k}(f,\omega,j)|^2 \right) \rho(d\omega). \qquad (14) \end{aligned}$$

The following result proved by Papageorgiou in [25] will be needed for our consideration. Take the uniform distribution for $\mathcal{B}_N(f)$, i.e., $p_f = \binom{N}{j}/2^N$ for all f with $\mathcal{B}_N(f) = j/N$. Then there are two positive numbers c_1 and c_2 with the following properties. For any algorithm $A_{n,k}$ in the standard quantum setting with n bit queries, such that $n \leq c_1 N$, and k qubits, let $p_{f,j}$ denote the probability of obtaining the index j through measurement. Let

$$\mu_f(J) \,=\, \sum_{j\in J} p_{f,j}$$

denote the probability of a subset J of $\{0, 1, \ldots, 2^k - 1\}$. Then it is proved in [25] that

$$\sum_{f\in F_N} p_f\, \mu_f\left(\{j :\, |\mathcal{B}_N(f) - A_{n,k}(f,j)| \,\geq\, c_2/n\,\}\right) \,\geq\, 0.25.$$

From Chebyshev's inequality we conclude that

$$\sum_{f\in F_N} p_f \sum_{j=0}^{2^k-1} |\mathcal{B}_N(f) - A_{n,k}(f,j)|^2 \;\geq\; \frac{1}{4}\left(\frac{c_2}{n}\right)^2.$$

We apply the last inequality for an arbitrary algorithm $A_{n,k}$ from the quantum setting with randomized queries and with a fixed ω. Here, we use the fact that the algorithm $A_{n,k}(\cdot,\omega,\cdot)$ can be regarded as an algorithm from the standard quantum setting. We thus have

$$\sum_{f\in F_N} p_f \sum_{j=0}^{2^k-1} p_{f,j,\omega} |\mathcal{B}_N(f) - A_{n,k}(f,\omega,j)|^2 \geq \frac{1}{4}\left(\frac{c_2}{n}\right)^2.$$

Since the right-hand side is independent of ω, from (14) we obtain

$$e^{\mathrm{qua-ran}}(A_{n,k}) \;\geq\; \frac{c_2}{2n}.$$

Hence, $e^{\mathrm{qua-ran}}(A_{n,k}) \leq \varepsilon$ implies that $n = \Omega(\varepsilon^{-1})$ and

$$\mathrm{comp}^{\mathrm{que-ran}}(\mathcal{B}_N) \;=\; \Omega\left(\varepsilon^{-1}\right).$$

To prove a lower bound on the randomized qubit complexity, we use Theorem 4.1. In the randomized setting on a classical computer it is known that the randomized complexity is of order ε^{-2}, see [24]. Then Theorem 4.1 yields

$$\mathrm{comp}^{\mathrm{qub-ran}}(\mathcal{B}_N) \;=\; \Omega\left(\varepsilon^{-1}\right).$$

We turn to upper bounds. The idea is the same as for multivariate integration for $r = 0$ which was studied before. That is, we apply the Boolean summation algorithm with seven repetitions to the Monte Carlo algorithm $m^{-1}\sum_{j=1}^m f(\omega_j)$ with independently and uniformly distributed ω_j over $\{0,1,\ldots,N-1\}$. We stress that this algorithm uses randomized queries $Q_{f,\omega}$ and the rest of unitary matrices are deterministic, i.e., $U_{j,\omega} = U_j$ in (13).

The same analysis done for $r = 0$ yields that the randomized error is ε. Since this algorithm uses of order ε^{-1} randomized queries and $\log \varepsilon^{-1}$ qubits, we obtain upper bounds which match the lower bounds. This completes the proof. □.

It is interesting to compare the complexities of the Boolean summation problem in the quantum settings with deterministic and randomized queries. As we see, the query complexities are roughly the same in both settings. The qubit complexities, however, are quite different. For deterministic queries, the number of qubits depends on the common domain of Boolean functions, and we need roughly $\log N$ qubits which can be arbitrary large for large N. For randomized queries, the number of qubits does not depend on the common domain of Boolean functions. It depends on the error parameter through $\log \varepsilon^{-1}$. As we shall see in the following sections, there is sometimes an exponential difference between $\log N$ and $\log \varepsilon^{-1}$.

5.2 Real Summation

We finish this section by a brief note on the real summation problem. We now consider $f : \{0, 1 \ldots, N-1\} \to [0,1]$ and want to approximate

$$\mathrm{SUM}_N(f) \;=\; \frac{1}{N} \sum_{j=0}^{N-1} f(j). \tag{15}$$

A known idea is to replace the real number $f(j)$ from $[0,1]$ by its binary expansion,

$$f(j) \;=\; \sum_{i=1}^{\infty} 2^{-i}\, f(i,j) \quad \text{with } f(i,j) \in \{0,1\},$$

We define $K = \lceil \log \varepsilon^{-2} \rceil$ and truncate $f(j)$ to K bits. Let

$$S_K(f) \;=\; \frac{1}{N} \sum_{j=0}^{N-1} \sum_{i=1}^{K} 2^{-i}\, f(i,j).$$

Clearly, $|\mathrm{SUM}_N(f) - S_K(f)| \le 2^{-K} \le \varepsilon^2$. To obtain a Boolean function we finally define the set

$$D \;=\; \left\{ (i,j,p) \,:\, i = 1,2,\ldots,K,\; j = 0,1,\ldots,N-1,\; p = 1,2,\ldots,2^{K-i} \right\}$$

of cardinality $N(2^K - 1)$ and a Boolean function $b_f : D \to \{0,1\}$ by

$$b(i,j,k) \;=\; \delta_{f(i,j),1},$$

where $\delta_{i,j}$ is the Kronecker delta. Then

$$\sum_{i=1}^{K} 2^{-i} f(i,j) \;=\; 2^{-K} \sum_{i=1}^{K} 2^{K-i} f(i,j) \;=\; 2^{-K} \sum_{i=1}^{K} \sum_{p=1}^{2^{K-i}} b(i,j,p).$$

Thus

$$S_K(f) \;=\; \mathcal{B}_{N2^K}(b_f) \;=\; \frac{1}{N2^K} \sum_{j=0}^{N-1} \sum_{i=1}^{K} \sum_{p=1}^{2^{K-i}} b(i,j,p)$$

is a Boolean summation problem. We compute $\mathcal{B}_{N2^K}(b_f)$ with error $\varepsilon - \varepsilon^2$ by the Boolean summation algorithm with seven repetitions as explained in Theorem 5.2 and obtain $A_{n,k}(b_f)$ which uses of order ε^{-1} randomized queries and $\log \varepsilon^{-1}$ qubits. It is easy to check that $A_{n,k}(b_f)$ approximates $S(f)$ with the randomized error at most ε. We summarize this in the corollary.

Corollary 5.1. *The complexities of the real summation problem (*15*) in the quantum setting with randomized queries satisfy*

$$\mathrm{comp}^{\mathrm{que-ran}}(\mathrm{SUM}_N) \;=\; \Theta\left(\varepsilon^{-1}\right) \qquad \mathrm{comp}^{\mathrm{qub-ran}}(\mathrm{SUM}_N) \;=\; \Theta\left(\log \varepsilon^{-1}\right).$$

Furthermore, these bounds are both attained by the Boolean summation algorithm with seven repetitions applied to

$$\frac{1}{m} \sum_{\ell=1}^{m} f(i_\ell, j_\ell, p_\ell)$$

with $m = \Theta(\varepsilon^{-2})$ and with independent uniformly distributed (i_ℓ, j_ℓ, p_ℓ) over D.

6 Multivariate Integration

In this section, we consider the multivariate integration problem $\mathrm{INT}_{d,r}$ for the class $F_{d,r}$ which we used throughout as an illustrative example. The purpose of this section is to study this problem in the quantum settings with deterministic and randomized queries.

We begin with deterministic queries. For $r = 0$ the problem is unsolvable. For $r \geq 1$, sharp bounds on the bit query follow from [24],

$$\mathrm{comp}^{\mathrm{que-std}}(\varepsilon, \mathrm{INT}_{d,r}) \;=\; \Theta\left(\varepsilon^{-1/(1+r/d)}\right).$$

This bound is achieved by the Boolean summation algorithm with seven repetitions and uses of order $\log \varepsilon^{-1}$ qubits, as shown in the previous section.

Observe that the lower bound on the qubit complexity is of order $\log \varepsilon^{-1}$ due to Theorem 4.1 and the fact that the worst case information complexity is of order $\varepsilon^{-d/r}$. Hence, we have

$$\mathrm{comp}^{\mathrm{qub-ran}}(\varepsilon, \mathrm{INT}_{d,r}) \;= \Theta\left(\log\, \varepsilon^{-1}\right).$$

We now turn to randomized queries. The case $r = 0$ has already been covered and we know that we can solve the problem using of order ε^{-1} randomized bit queries and $\log \varepsilon^{-1}$ qubits.

For $r \geq 1$, we use the same quantum algorithm as in [24]. Since the Boolean summation algorithm uses the same order of bit queries for the randomized and probabilistic quantum errors, we obtain the same upper bounds on the number of bit queries and qubits.

The lower bound on the number of randomized queries can be derived as in [24] and using the results on the Boolean and real summation problems of the previous section. This yields that the randomized bit query complexity is of order $\varepsilon^{-1/(1+r/d)}$. The lower bound on the number of qubits follows from Theorem 4.1 and the fact that the randomized complexity is of order $\varepsilon^{-2/(1+2r/d)}$.

We summarize these results in the following theorem.

Theorem 6.1. *Consider the multivariate integration problem* $\mathrm{INT}_{d,r}$ *for the class* $F_{d,r}$.

- *Let* $r = 0$.
  - *In the quantum setting with deterministic queries, we have*

$$\begin{aligned} \mathrm{comp}^{\mathrm{que-std}}(\varepsilon, \mathrm{INT}_{d,0}) &= \infty, \\ \mathrm{comp}^{\mathrm{qub-std}}(\varepsilon, \mathrm{INT}_{d,0}) &= \infty. \end{aligned}$$

  - *In the quantum setting with randomized queries, we have*

$$\begin{aligned} \mathrm{comp}^{\mathrm{que-ran}}(\varepsilon, \mathrm{INT}_{d,0}) &= \Theta\left(\varepsilon^{-1}\right), \\ \mathrm{comp}^{\mathrm{qub-ran}}(\varepsilon, \mathrm{INT}_{d,0}) &= \Theta\left(\log\, \varepsilon^{-1}\right). \end{aligned}$$

- *Let* $r \geq 1$.

– *In the quantum setting with deterministic queries, we have*

$$\begin{aligned} \mathrm{comp}^{\mathrm{que-std}}(\varepsilon, \mathrm{INT}_{d,r}) &= \Theta\left(\varepsilon^{-1/(1+r/d)}\right), \\ \mathrm{comp}^{\mathrm{qub-std}}(\varepsilon, \mathrm{INT}_{d,r}) &= \Theta\left(\log \varepsilon^{-1}\right). \end{aligned}$$

– *In the quantum setting with randomized queries, we have*

$$\begin{aligned} \mathrm{comp}^{\mathrm{que-ran}}(\varepsilon, \mathrm{INT}_{d,r}) &= \Theta\left(\varepsilon^{-1/(1+r/d)}\right), \\ \mathrm{comp}^{\mathrm{qub-ran}}(\varepsilon, \mathrm{INT}_{d,r}) &= \Theta\left(\log \varepsilon^{-1}\right). \end{aligned}$$

Hence, for $r = 0$ we see a big difference between the two settings for the multivariate integration problem, whereas for $r \geq 1$, the two settings lead to the same order of bit and qubit complexities.

7 Path Integration

Path integration can be regarded as integration of functions of infinitely many variables or, more formally, as integration over some class of functions; for more information and references see [35]. Path integrals occur in quantum physics, chemistry and mathematical finance. They are also the solutions of certain differential equations and mathematical finance problems.

Here we consider a specific example of path integration studied in [35]. We take the space $X := C([0,1])$ of continuous functions defined on $[0,1]$ with the norm $\|x\| = \max_{t\in[0,1]} |x(t)|$. The space X is equipped with the classical Wiener measure w for which

$$\int_X x(t)\, w(dt) \;=\; 0 \quad \forall\, t \in [0,1] \quad \text{and} \quad \int_X x(t)x(u)\, w(dx) \;=\; \min(t,u) \quad \forall\, t,u \in [0,1].$$

We consider the class F of real valued w-integrable functions $f : X \to \mathbb{R}$ which are bounded and satisfy a Lipschitz condition. More precisely, let the norm of f be given by $\|f\| = \sup_{x\in X} |f(x)|$. Then the class F is defined as

$$F \;=\; \left\{\, f \;:\; \|f\| \;\leq\; 1,\; |f(x) - f(y)| \;\leq\; \|x-y\|_{L_2([0,1])}\; \forall\, x,y \in X \,\right\}.$$

Let $G = \mathbb{R}$. The path integration $S := \mathrm{PATH}$ is given by

$$\mathrm{PATH}(f) \;=\; \int_X f(x)\, w(dx).$$

We first consider the standard quantum setting. It was shown in [35] that we can compute an ε-approximation for path integrals from the class F with probability $\frac{3}{4}$ using of order ε^{-1} bit queries and $\varepsilon^{-2} \log \varepsilon^{-1}$ qubits. The bound on the number of bit queries is sharp in the sense that for any positive α it cannot be smaller that $\varepsilon^{1-\alpha}$ as ε goes to zero. The sharpness of the number of qubits was not discussed. These bounds are obtained by reducing the path

integration problem to the summation problem which was solved by the Boolean summation algorithm.

We consider the error defined by (9) in the standard quantum setting. Due to the fact that the Boolean summation algorithm enjoys optimality properties also in this setting, we conclude that the same bounds as above also hold for the error (9). Furthermore, the bound on the number of qubits is sharp since the worst case information complexity is of order $\varepsilon^{-c\,\varepsilon^{-2}}$ for some positive c as proved in [6]. Then Theorem 3.1 yields that the number of qubits must be of order $\varepsilon^{-2}\log\varepsilon^{-1}$.

We now consider the quantum setting with randomized queries. It was proven in [37] that the non-adaptive information complexity in the randomized classical setting is of the form

$$\mathrm{comp}^{\mathrm{inf-ran}}(\varepsilon,\mathrm{PATH}) \;=\; \Theta\left(\varepsilon^{-2(1+o(1))}\right) \quad \text{as } \varepsilon \;\to\; 0.$$

This and Theorem 4.1 yields that the number of qubits is at least of order $\log\varepsilon^{-1}$.

We now show that in the quantum setting with randomized queries, we can solve the path integration problem by using of order ε^{-1} bit queries and $\log\varepsilon^{-1}$ qubits. The space X can be embedded in the Hilbert space $L_2([0,1])$ for which the embedding $\mathrm{Im}: X \to L_2([0,1])$, $\mathrm{Im}\,x = x$ for all $x \in X$, is a continuous linear operator. Let $\nu = w\,\mathrm{Im}^{-1}$ be a zero mean Gaussian measure on $L_2([0,1])$. Then the covariance operator C_ν of the measure ν has eigenpairs, $C_\nu\eta_i = \lambda_i\eta_i$, where

$$\eta_i(x) \;=\; \sqrt{2}\,\sin\left(\frac{2i-1}{2}\pi\,x\right), \qquad \lambda_i \;=\; \frac{4}{\pi^2(2i-1)^2}.$$

As in [35] we first approximate $\mathrm{PATH}(f)$ by

$$\mathrm{INT}_d(f) \;=\; \int_{\mathbb{R}^d} f_d(t)\,\mu_d(dt),$$

where

$$f_d(t) \;=\; f\left(\mathrm{Im}^{-1}(t_1\eta_1 + t_2\eta_2 + \cdots + t_d\eta_d)\right)$$

and μ_d is the d dimensional Gaussian measure with the mean zero and variances λ_i. That is, its density function is of the form

$$\frac{1}{(2\pi)^{d/2}\sqrt{\lambda_1\lambda_2\cdots\lambda_d}}\,\exp\left(-t_1^2/(2\lambda_1) - \cdots - t_d^2/(2\lambda_d)\right).$$

In [35], it is proved that for $d = \Theta(\varepsilon^{-2})$ we have $|\mathrm{PATH}(f) - \mathrm{INT}_d(f)| \le \varepsilon/3$ for all $f \in F$.

The integral $\mathrm{INT}_d(f)$ can be approximated by the Monte Carlo

$$\frac{1}{n}\sum_{j=1}^{n} f_d(t_j)$$

with iid points t_j distributed according to the measure μ_d. Note that for $f \in F$, we have $|f_d(t_j)| \le 1$ and clearly the variance of f_d is bounded by 1. Therefore for $n = \lceil 9\varepsilon^{-2}\rceil$, the randomized error of approximating $\mathrm{INT}_d(f)$ is at most $\varepsilon/3$, and the randomized error of

approximating $\mathrm{INT}(f)$ is at most $2\varepsilon/3$. Finally, it is enough to use the Boolean summation algorithm to approximate the last sum with the randomized error $\varepsilon/3$ which can be done with of order ε^{-1} bit queries and $\log n = \Theta(\log \varepsilon^{-1})$ qubits. The randomized error of approximation $\mathrm{INT}(f)$ is at most ε, as claimed.

This and the previous lower bound on the number of queries yield that the randomized qubit complexity of path integration is of order $\log \varepsilon^{-1}$. For the randomized query complexity we have so far an upper bound of order ε^{-1}. We can get a lower bound by applying the same proof technique as in Theorem 3 of [35]. That is, the path integration problem is reduced to the real summation problem for which we use a lower bound presented in Corollary 5.1. This yields that

$$\lim_{\varepsilon \to 0} \varepsilon^{1-\alpha} \, \mathrm{comp}^{\mathrm{que-ran}}(\varepsilon, \mathrm{PATH}) \, = \, \infty \quad \forall \, \alpha \in (0,1).$$

We summarize these results in the following theorem.

Theorem 7.1. *Consider path integration equipped with the Wiener measure for the class F of Lipschitz functions.*

- *In the quantum setting with deterministic queries, we have*

$$\begin{aligned} \mathrm{comp}^{\mathrm{que-std}}(\varepsilon, \mathrm{PATH}) &= \Theta\left(\varepsilon^{-1+o(1)}\right), \\ \mathrm{comp}^{\mathrm{qub-std}}(\varepsilon, \mathrm{PATH}) &= \Theta\left(\varepsilon^{-2} \log \varepsilon^{-1}\right). \end{aligned}$$

- *In the quantum setting with randomized queries, we have*

$$\begin{aligned} \mathrm{comp}^{\mathrm{que-ran}}(\varepsilon, \mathrm{PATH}) &= \Theta\left(\varepsilon^{-1+o(1)}\right), \\ \mathrm{comp}^{\mathrm{qub-ran}}(\varepsilon, \mathrm{PATH}) &= \Theta\left(\log \varepsilon^{-1}\right). \end{aligned}$$

The essence of this theorem is that for path integration we have an exponential improvement in the number of qubits in the quantum setting with randomized queries whereas the number of queries remains roughly the same in both settings. We stress that the optimal bounds for bit queries and qubits are both attained by the same quantum algorithm based on the Boolean summation algorithm.

8 Appendix: Probabilistic Errors

We briefly indicate what kind of results are possible if one studies probabilistic errors in the randomized classical setting and in the quantum settings with deterministic and randomized queries.

We begin with the randomized classical setting. Instead of the randomized error (3) we now consider the *probabilistic* error of the algorithm A_n which is defined by the worst case performance with respect to f and the worst case performance with respect to ω modulo a set of measure δ for some (usually small) $\delta \in (0,1)$. That is,

$$e^{\mathrm{ran}}(A_n, \delta) \, = \, \sup_{f \in F} \, \inf_{B \subset \Omega, \, \rho(B) \le \delta} \, \sup_{\omega \in \Omega \setminus B} \, \|S(f) - A_n(f, \omega)\|. \tag{16}$$

From Chebyshev's inequality we have

$$e^{\rm ran}(A_n,\delta) \;\le\; \frac{1}{\sqrt{\delta}}\, e^{\rm ran}(A_n). \tag{17}$$

Better estimates with respect to δ are available under additional assumptions on S. In any case, the dependence on δ is quite modest and everything depends on the randomized error $e^{\rm ran}(A_n)$. This is probably why the probabilistic error (16) has not been as widely studied as the randomized error (3) for continuous problems on a classical computer.

The probabilistic error yields the *(non-adaptive) information complexity* defined by

$$\mathrm{comp}^{\rm inf-ran}(\varepsilon,\delta,S) \;=\; \min\left\{\, n \;:\; \exists\, A_n^{\rm nad} \text{ such that } e^{\rm ran}(A_n^{\rm nad},\delta) \;\le\; \varepsilon \,\right\}.$$

Clearly, (17) implies that

$$\mathrm{comp}^{\rm inf-ran}(\varepsilon,\delta,S) \;\le\; \mathrm{comp}^{\rm inf-ran}(\varepsilon\sqrt{\delta},S),$$

We now turn to the standard quantum setting. The usual way of defining the error in this setting is analogous to the probabilistic error. That is, the *probabilistic* error of $A_{n,k}$ is defined as the smallest α for which

$$\|S(f) - A_{n,k}(f,j)\| \;\le\; \alpha$$

holds with probability at most $1-\delta$ with respect to j for every f from F. Here, $\delta \in (0,1)$. This definition can be formalized as follows. For $f \in F$ and an arbitrary subset J of $\{0,1,\ldots,2^k-1\}$, let $\mu_f(J) = \sum_{j\in J} p_{f,j}$ be a measure of J. Then the probabilistic error of $A_{n,k}$ in the standard quantum setting is

$$e^{\rm qua-std}(A_{n,k},\delta) \;=\; \sup_{f\in F} \min_{J:\,\mu_f(J)\le\delta} \max_{j\in\{0,1,\ldots,2^k-1\}\setminus J} \|S(f) - A_{n,k}(f,j)\|.$$

For some operators S, such as linear functionals, it is typical to take, say, $\delta = \frac{1}{4}$, and obtain a quantum algorithm working with probability $1-\delta$ by repeating several times the quantum algorithm working with $\delta = \frac{1}{4}$ and by taking the median as the final result. If the number of repetitions is large enough we can boost probability of success to $1-\delta$. Details can be found in [7].

The *probabilistic query complexity in the standard quantum setting* is defined as

$$\mathrm{comp}^{\rm que-std}(\varepsilon,\delta,S) \;=\; \min\left\{\, n \;:\; \exists\, A_{n,k} \text{ such that } e^{\rm qua-std}(A_{n,k},\delta) \;\le\; \varepsilon \,\right\}, \tag{18}$$

and the *probabilistic qubit complexity in the standard quantum setting* is defined as

$$\mathrm{comp}^{\rm qub-std}(\varepsilon,\delta,S) \;=\; \min\left\{\, k \;:\; \exists\, A_{n,k} \text{ such that } e^{\rm qua-std}(A_{n,k},\delta) \;\le\; \varepsilon \,\right\}. \tag{19}$$

Example: Multivariate Integration (continued)
Assume that $r \ge 1$. For $F = F_{d,r}$, it has been proven by Novak, see [24], that for $\delta = \frac{1}{4}$ the minimal error of quantum algorithms $A_{n,k}$ is of order $n^{-1-r/d}$, and is achieved by an algorithm

that uses of order $\log \varepsilon^{-1}$ qubits. The idea of the proof was to reduce the integration problem to real and then Boolean summation and apply the the Boolean summation algorithm of [5]. This algorithm for the summation of N terms with n queries, $n \ll N$, has probabilistic error of order n^{-1} which is optimal due to [18].

This implies that the query complexity $\mathrm{comp}^{\mathrm{que-std}}(\varepsilon, \frac{1}{4}, \mathrm{INT}_{d,r})$ is of order $\varepsilon^{-1/(1+r/d)}$. For d much larger than r, we thus obtain roughly a quadratic speedup over the randomized setting, and an exponential speedup over the worst case setting.

For arbitrary δ, we can use roughly $\log \delta^{-1}$ repetitions of the algorithm used for $\delta = \frac{1}{4}$ and take the median of computed results as the final result, again see [7, 21]. This implies that $\mathrm{comp}^{\mathrm{que-std}}(\varepsilon, \delta, \mathrm{INT}_{d,r})$ is of order $\varepsilon^{-1/(1+r/d)} \log \delta^{-1}$.

For $r = 0$, it is easy to see by the same argument which we used for the randomized errors, that the integration problem cannot be solved for the probabilistic error in the standard quantum setting. □

As before, Chebyshev's inequality yields

$$\begin{aligned} \mathrm{comp}^{\mathrm{que-std}}(\varepsilon, \delta, S) &\leq \delta^{-1/2}\, \mathrm{comp}^{\mathrm{que-std}}(\varepsilon, S) \\ \mathrm{comp}^{\mathrm{qub-std}}(\varepsilon, \delta, S) &\leq \delta^{-1/2}\, \mathrm{comp}^{\mathrm{qub-std}}(\varepsilon, S). \end{aligned}$$

Again the dependence on δ can be improved for some S. Note, however, that even for general S, the dependence on δ is quite weak.

We now show lower bounds on the probabilistic qubit complexity in terms of the non-adaptive information complexities in the worst case and randomized settings as well as in terms of the ε-entropy.

Theorem 8.1.

$$\begin{aligned} \mathrm{comp}^{\mathrm{qub-std}}(\varepsilon, \delta, S) &\geq \log\ \mathrm{comp}^{\mathrm{inf-wor}}(2\varepsilon, S) \quad \forall\, \delta \in (0, \tfrac{1}{2}), \\ \mathrm{comp}^{\mathrm{qub-std}}(\varepsilon, \delta, S) &\geq \log\ \mathrm{comp}^{\mathrm{inf-ran}}(\varepsilon, \delta, S), \\ \mathrm{comp}^{\mathrm{qub-std}}(\varepsilon, \delta, S) &\geq \mathrm{Ent}(\varepsilon, S(F)). \end{aligned}$$

Proof.
(1) To prove the first inequality, we take a quantum algorithm $A_{n,k}$ which uses the minimal number of qubits $k = \mathrm{comp}^{\mathrm{qub-std}}(\varepsilon, \delta, S)$ with $e^{\mathrm{qua-std}}(A_{n,k}, \delta) \leq \varepsilon$. We have

$$\begin{aligned} \varepsilon \geq e^{\mathrm{qua-std}}(A_{n,k}, \delta) &= \sup_{f \in F} \min_{J:\, \mu_f(J) \leq \delta} \max_{j \in \{0,1,\ldots,2^k-1\} \setminus J} \|S(f) - A_{n,k}(f,j)\| \qquad (20) \\ &= \sup_{f \in F} \max_{j \in \{0,1,\ldots,2^k-1\} \setminus J(f)} \|S(f) - A_{n,k}(f,j)\|, \end{aligned}$$

where $J(f)$ is a subset of $\{0, 1, \ldots, 2^k - 1\}$, $\mu_f(J(f)) \leq \delta$, for which the corresponding minimum is attained. Such a set exists since we have finitely many such subsets, however, $J(f)$ is not necessarily unique. Let

$$M(f) = \{\, 0, 1, \ldots, 2^k - 1 \,\} \setminus J(f).$$

Clearly, $\mu_f(M(f)) \geq 1 - \delta$.

For an arbitrary $f \in F$, we take two functions f_1 and f_2 such that $N(f_1) = N(f_2) = N(f)$. Note that the measures μ_{f_1} and μ_{f_2} are the same. For $\delta < \frac{1}{2}$, there exists an index j^* which belongs to $M(f_1) \cap M(f_2)$. Indeed, otherwise the sets $M(f_1)$ and $M(f_2)$ would be disjoint and

$$\begin{aligned} 1 \geq \mu_f(M(f_1) \cup M(f_2)) &= \mu_f(M(f_1)) + \mu_f(M(f_2)) \\ &= \mu_{f_1}(M(f_1)) + \mu_{f_2}(M(f_2)) \geq 2(1-\delta) > 1 \end{aligned}$$

is a contradiction. For this index j^*, we have $a = A_{n,k}(f_1, j^*) = A_{n,k}(f_2, j^*)$. From (20) we get

$$\varepsilon \geq \tfrac{1}{2}\left(\|S(f_1) - a\| + \|S(f_2) - a\|\right) \geq \tfrac{1}{2}\|S(f_1) - S(f_2)\|.$$

Hence,

$$\sup_{f \in F} \sup_{f_1, f_2 \in F,\, N(f_1) = N(f_2) = N(f)} \|S(f_1) - S(f_2)\| \leq 2\varepsilon,$$

and the rest of the proof is the same as in the proof of Theorem 3.1.

(2) To prove the second inequality, we compare the qubit complexity to the randomized non-adaptive information complexity. Observe that any quantum algorithm $A_{n,k}$ may be regarded as a randomized algorithm which uses non-adaptive deterministic information of cardinality at most 2^k with randomized elements $\omega \in \{0, 1, \ldots, 2^k - 1\}$ taking values j with probability $p_{f,j}$. Furthermore, the probabilistic error of $A_{n,k}$ is exactly the same as the error in the probabilistic randomized setting. Therefore, $e^{\mathrm{ran}}(A_{n,k}, \delta) \leq \varepsilon$ can hold only if the cardinality of $A_{n,k}$ is $\mathrm{comp}^{\mathrm{inf-ran}}(\varepsilon, \delta, S)$. This means that $2^k \geq \mathrm{comp}^{\mathrm{inf-ran}}(\varepsilon, \delta, S)$, as claimed.

(3) To prove the third inequality, observe that we now have for any $f \in F$,

$$\max_{j \in M(f)} \|S(f) - \phi(j)\| \leq \varepsilon,$$

where the subset $M(f)$ of $\{0, 1, \ldots, 2^k - 1\}$ is defined as above. Hence,

$$\min_{j = 0, 1, \ldots, 2^k - 1} \|S(f) - \phi(j)\| \leq \varepsilon,$$

and the rest is as in the proof of Theorem 3.2. □

We now turn to the quantum setting with randomized queries. Instead of (9), we consider the *probabilistic* error of $A_{n,k}$ which is defined as the smallest α for which

$$\|S(f) - A_{n,k}(f, \omega, j)\| \leq \alpha$$

holds with probability at least $1 - \delta$ with respect to j and ω for all f from F. More precisely, as before, for $J \in \{0, 1, \ldots, 2^k - 1\}$ we define the measure of J by $\mu_{f,\omega}(J) = \sum_{j \in J} p_{f,j,\omega}$. Then the probabilistic error of $A_{n,k}$ in the quantum setting with randomized queries is

$$e^{\mathrm{qua-ran}}(A_{n,k}, \delta) = \sup_{f \in F} \inf_{B \in \Omega,\, \rho(B) \leq \delta} \sup_{\omega \in \Omega \setminus B} \min_{J :\, \mu_{f,\omega}(J) \leq \delta} \max_{j \in \{0, 1, \ldots, 2^k - 1\} \setminus J} \|S(f) - A_{n,k}(f, \omega, j)\|.$$

Observe that if we choose $A_{n,k}$ independently of ω then, modulo measurement, everything will be deterministic and the last definition coincides with the probabilistic error in the standard quantum setting.

The probabilistic query/qubit complexity in the quantum setting with randomized queries are defined, analogously as in the standard quantum setting, by minimizing the number of queries/qubits needed to find a quantum algorithm whose probabilistic error is at most ε. That is, the *probabilistic query complexity in the quantum setting with randomized queries* is defined by

$$\mathrm{comp}^{\mathrm{que-ran}}(\varepsilon, \delta, S) \;=\; \min\left\{\, n \;:\; \exists\, A_{n,k} \text{ such that } e^{\mathrm{qua-ran}}(A_{n,k}, \delta) \;\leq\; \varepsilon \,\right\},$$

and the *probabilistic qubit complexity in the quantum setting with randomized queries* is defined by

$$\mathrm{comp}^{\mathrm{qub-ran}}(\varepsilon, S, \delta) \;=\; \min\left\{\, k \;:\; \exists\, A_{n,k} \text{ such that } e^{\mathrm{qua-ran}}(A_{n,k}, \delta) \;\leq\; \varepsilon \,\right\}.$$

We now show that Chebyshev's inequality implies

$$e^{\mathrm{qua-ran}}(A_{n,k}, \delta) \;\leq\; \delta^{-1}\; e^{\mathrm{qua-ran}}(A_{n,k}). \tag{21}$$

Indeed, let

$$E(f, \omega) \;=\; \sum_{j=0}^{2^k-1} p_{f,j,\omega} \|S(f) - A_{n,k}(f, \omega, j)\|^2.$$

Hence,

$$e^{\mathrm{qua-ran}}(A_{n,k})^2 \;=\; \sup_{f \in F} \int_\Omega E(f, \omega)\, \rho(d\omega).$$

Define the sets $B(f)$ and $J(f, \omega)$ by

$$\begin{aligned} \Omega \setminus B(f) &= \left\{\omega :\; E(f, \omega) \;\leq\; \delta^{-1} e^{\mathrm{qua-ran}}(A_{n,k})^2 \,\right\}, \\ \left\{0, 1, \ldots, 2^{k_\omega} - 1\right\} \setminus J(f, \omega) &= \left\{j :\; \|S(f) - A_{n,k}\|^2 \;\leq\; \delta^{-1} E(f, \omega) \,\right\}. \end{aligned}$$

Chebyshev's inequality tells us that $\rho(B(f)) \leq \delta$ and $\mu_{f,\omega}(J(f, \omega)) \leq \delta$. Then

$$e^{\mathrm{qua-ran}}(A_{n,k}, \delta)^2 \;\leq\; \sup_{f \in F} \sup_{\omega \in \Omega \setminus B(f)} \delta^{-1}\; E(f, \omega) \;\leq \delta^{-2}\; e^{\mathrm{qua-ran}}(A_{n,k})^2,$$

as claimed.

The probabilistic and randomized query and qubit complexities are related. From (21) we have

$$\begin{aligned} \mathrm{comp}^{\mathrm{que-ran}}(\varepsilon, \delta, S) &\leq \mathrm{comp}^{\mathrm{que-ran}}(\varepsilon\delta, S), \\ \mathrm{comp}^{\mathrm{qub-ran}}(\varepsilon, \delta, S) &\leq \mathrm{comp}^{\mathrm{qub-ran}}(\varepsilon\delta, S). \end{aligned}$$

It is also easy to see to check that

$$\begin{aligned} \mathrm{comp}^{\mathrm{qub-ran}}(\varepsilon, S) &\geq \log\, \mathrm{comp}^{\mathrm{inf-ran}}(\varepsilon, S), \\ \mathrm{comp}^{\mathrm{qub-ran}}(\varepsilon, \delta, S) &\geq \log\, \mathrm{comp}^{\mathrm{inf-ran}}(\varepsilon, \delta, S). \end{aligned}$$

9 Acknowledgment

I am grateful for many discussions and valuable comments from S. Heinrich, M. Kwas, E. Novak, A. Papageorgiou and J. F. Traub.

References

[1] N. S. Bakhvalov (1959), *On approximate calculation of integrals* (in Russian), Vestnik MGV, Ser. Mat. Mekh. Aston. Fiz. Khim., 4, 3-18.

[2] R. Beals, H. Buhrman, R. Cleve, R. Mosca, and R. de Wolf (1988), *Quantum lower bounds by polynomials*, Proceedings FOCS'98, 352-361. Also quant-ph/9802049.

[3] E. Bernstein, and U. Vazirani (1997), *Quantum complexity theory*, SIAM J. Computing, 26, 1411-1473.

[4] A. J. Bessen (2005), *A lower bound for phase estimation on a quantum computer*, Physical Review A, 71(4): 042313. Also quant-ph/0412008.

[5] G. Brassard, P. Hoyer, M. Mosca, and A. Tapp (2002), *Quantum Amplitude Amplification and Estimation* Contemporary Mathematics, Vol. 305, Am. Math. Soc., 53–74. Also http://arXiv.org/quant-ph/0005055.

[6] F. Curbera (200), *Delayed curse of dimension for Gaussian integration*, J. Complexity, 16, 474-506.

[7] S. Heinrich (2002), *Quantum Summation with an Application to Integration*, J. Complexity, 18(1), 1–50. Also http://arXiv.org/quant-ph/0105116.

[8] S. Heinrich (2003), *Quantum integration in Sobolev spaces*, J. Complexity, 19, 19–42.

[9] S. Heinrich (2004), *Quantum Approximation I. Embeddings of Finite Dimensional L_p Spaces,* J. Complexity, 20, 5–26. Also http://arXiv.org/quant-ph/0305030.

[10] S. Heinrich (2004), *Quantum Approximation II. Sobolev Embeddings,* J. Complexity, 20, 27–45. Also http://arXiv.org/quant-ph/0305031.

[11] S. Heinrich, M. Kwas, and H. Woźniakowski (2004), *Quantum Boolean summation with repetitions in the worst-average case setting,* in Monte Carlo and Quasi-Monte Carlo Methods 2002, ed. H. Niederreiter, 243-258, Springer Verlag, Berlin.

[12] S. Heinrich, *On the power of quantum algorithms for vector valued mean computation*, submitted for publication, 2004. See http://archiv.org/quant-ph/04031109.

[13] S. Heinrich, E. Novak, and H. Pfeiffer (2004) *How many random bits do we need for Monte Carlo integration?*, in Monte Carlo and Quasi-Monte Carlo Methods 2002, ed. H. Niederreiter, Proceedings of a conference held at the national University of Singapore, November, 2002, Springer, Berlin, 27-49.

[14] P. Jaksch and A. Papageorgiou (2003), *Eigenvector approximation leading to exponential speedup of quantum eigenvalue calculation*, Phys. Rev. Lett., 91, 257902. Also http://arXiv.org/quant-ph/0308016.

[15] B. Z. Kacewicz (2005), *Improved bounds on the randomized and quantum complexity of initial value problems*, J. Complexity, 21, 740-756.

[16] M. Kwas, and H. Woźniakowski (2004), *Sharp error bounds on quantum Boolean summation in various settings*, J. Complexity, 20,669-698.

[17] G. G. Lorentz (1966), *Approximation of functions*, Holt, New York.

[18] A. Nayak and F. Wu (1999), *The quantum query complexity of approximating the median and related statistics*, Proceedings of the 31th Annual ACM Symposium on the Theory of Computing (STOC), 384-393. LANL preprint quant-ph/9804066.

[19] H. Niederreiter (1992), *Random number generation and Quasi-Monte Carlo methods*, vol. 63 of SIAM CBMS-NSF Regional Conference Series in Applied Mathematics, SIAM, Philadelphia.

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

[21] E. Novak (1988), *Deterministic and Stochastic Error Bounds in Numerical Analysis*, Lecture Notes in Mathematics 1349, Springer-Verlag, Berlin.

[22] E. Novak (1992) *Optimal linear randomized methods for linear operators in Hilbert spaces*, J. Complexity, 8, 22-36.

[23] E. Novak and H. Woźniakowski (2001) *When are integration and discrepancy tractable?* Foundation of Computational Mathematics, Oxford, 1999, eds. R. A. DeVore, A. Iserles and E. Süli, Cambridge University Press, Cambridge, 211-266.

[24] E. Novak (2001), *Quantum complexity of integration,* J. Complexity, 17, 2-16. Also http://arXiv.org/quant-ph/0008124.

[25] A. Papageorgiou (2004) *Average case quantum lower bounds for computing the Boolean mean*, J. Complexity, 20, 713-731.

[26] A. Papageorgiou and H. Woźniakowski (2005), *Classical and quantum complexity of the Sturm-Liouville eigenvalue problem*, Quantum Information Processing, 4(2), 87-127. Also quant-ph/0502054.

[27] H. Pfeiffer (2005) *Monte Carlo with few random bits*, PhD Thesis, University of Jena, Shaker Verlag, Aachen.

[28] L. Plaskota (1996), *Noisy information and computational complexity*, Cambridge University Press, Cambridge.

[29] I. H. Sloan and S. Joe (1994), *Lattice methods for multiple integration*, Clarendon press, Oxford.

[30] P. W. Shor (1997), *Polynomial-time algorithms for prime factorization and discrete logarithm on a quantum computer*, SIAM J. Comput., 26(5), 1484-1509.

[31] J. F. Traub (1999), *A continuous model of computation*, Physics Today, May, 39-43.

[32] J. F. Traub, G. W. Wasilkowski and H. Woźniakowski (1988), *Information-Based Complexity*, Academic Press, New York.

[33] J. F. Traub and A. G. Werschulz (1998), *Complexity and Information*, Cambridge University Press, Cambridge.

[34] J. F. Traub and H. Woźniakowski (1980), *A general theory of optimal algorithms*, Academic Press, New York, 1980.

[35] J. F. Traub and H. Woźniakowski (2002), *Path integration on a quantum computer,* Quantum Information Processing, 1(5), 365-388, 2002. Also http://arXiv.org/quant-ph/0109113.

[36] G. W. Wasilkowski and H. Woźniakowski (1993), *There exists a linear problem with infinite combinatory complexity*, J. Complexity, 9, 326-337.

[37] G. W. Wasilkowski and H. Woźniakowski (1996), *On tractability of path integration*, j. of Math. Physics, 37(4), 2071-2088,

[38] A. G. Werschulz (1991), *The computational complexity of differential and integral equations: an information-based approach*, Oxford University Press, New York.