On Quantum Versions of the Yao Principle

Mart de Graaf[1]* and Ronald de Wolf[2]**

[1] CWI. P.O. Box 94079, 1090 GB Amsterdam, The Netherlands. mgdgraaf@cwi.nl
[2] UC Berkeley. 583 Soda Hall, Berkeley CA 94720, USA. rdewolf@cs.berkeley.edu

Abstract. The classical Yao principle states that the complexity $R_\epsilon(f)$ of an optimal *randomized* algorithm for a function f with success probability $1-\epsilon$ equals the complexity $\max_\mu D^\mu_\epsilon(f)$ of an optimal *deterministic* algorithm for f that is correct on a fraction $1-\epsilon$ of the inputs, weighed according to the hardest distribution μ over the inputs. In this paper we investigate to what extent such a principle holds for quantum algorithms. We propose two natural candidate quantum Yao principles, a "weak" and a "strong" one. For both principles, we prove that the quantum bounded-error complexity is a lower bound on the quantum analogues of $\max_\mu D^\mu_\epsilon(f)$. We then prove that equality cannot be obtained for the "strong" version, by exhibiting an exponential gap. On the other hand, as a positive result we prove that the "weak" version holds up to a constant factor for the query complexity of all symmetric Boolean functions.

1 Introduction

1.1 Motivation

In classical computing, the *Yao principle* [17] gives an equivalence between two kinds of randomness in algorithms: randomness inside the algorithm itself, and randomness on the inputs. Let us fix some model of computation for computing a Boolean function f, like query complexity, communication complexity, etc. Let $R_\epsilon(f)$ be the minimal complexity among all *randomized* algorithms that compute $f(x)$ with success probability at least $1-\epsilon$, for all inputs x. Let $D^\mu_\epsilon(f)$ be the minimal complexity among all *deterministic* algorithms that compute f correctly on a fraction of at least $1-\epsilon$ of all inputs, weighed according to a distribution μ on the inputs. The Yao principle now states that these complexities are equal if we look at the "hardest" input distribution μ:

$$R_\epsilon(f) = \max_\mu D^\mu_\epsilon(f).$$

It is a special case of Von Neumann's minimax theorem in game theory [10].

* Partially supported by EU fifth framework project QAIP, IST–1999–11234, and grant 612.055.001 from the Netherlands Organization for Scientific Research (NWO).
** Supported by Talent grant S 62–565 from NWO. Work done while at CWI.

Since its introduction, the Yao principle has been an extremely useful tool for deriving lower bounds on randomized algorithms from lower bounds on deterministic algorithms: choose some "hard" input distribution μ, prove a lower bound on deterministic algorithms that compute f correctly for "most" inputs, weighed according to μ, and then use $R_\epsilon(f) \geq D^\mu_\epsilon(f)$ to get a lower bound on $R_\epsilon(f)$. This method is used very often, because it is usually much easier to analyze deterministic algorithms than to analyze randomized ones.

In recent years *quantum* computation received a lot of attention. Here quantum mechanical principles are employed to realize more efficient computation than is possible with a classical computer. Famous examples are Shor's efficient quantum factoring algorithm [15] and Grover's search algorithm [8]. However, the field is still young and open questions abound. In particular, there has been a search for good techniques to provide lower bounds on quantum algorithms, particularly in the query model of computation. Two general methods in this direction are the *polynomial method* introduced by Beals, Buhrman, Cleve, Mosca, and de Wolf [3] and the *quantum adversary method* of Ambainis [2]. In this paper we investigate the possibility of a third method, a *quantum Yao principle*. It is our hope that such a principle will prove itself useful as a link between techniques for lower bounds on exact and bounded-error quantum algorithms.

The first difficulty one runs into when investigating a quantum version of the Yao principle, is the question what the proper quantum counterparts of $R_\epsilon(f)$ and $D^\mu_\epsilon(f)$ are. Let us fix the error probability at $\epsilon = \frac{1}{3}$ here (any other value in $(0, \frac{1}{2})$ would do as well). The quantum analogue of $R_{1/3}(f)$ is straightforward: let $Q_2(f)$ denote the minimal complexity among all *quantum* algorithms that compute $f(x)$ with probability at least $\frac{2}{3}$, for all inputs x. However, the inherently "random" nature of quantum algorithms prohibits a straightforward definition of "deterministic" quantum algorithms in analogy of deterministic classical algorithms. We therefore propose two different definitions, a weak and a strong one. In the following, let $f : D \to \{0,1\}$ be some function that we want to compute, with $D \subseteq \{0,1\}^N$. If $D = \{0,1\}^N$ then f is a *total* function, otherwise f is a *promise* function. Let A be a quantum algorithm, $P_A(x)$ the acceptance probability of A on input x (the probability of outputting 1 on input x), and $\mu : D \to [0,1]$ a probability distribution over the inputs.

Definition 1. *A is* weakly $\frac{2}{3}$-exact *for f with respect to μ iff $\mu(\{x \mid P_A(x) = f(x)\}) \geq \frac{2}{3}$.*

Definition 2. *A is* strongly $\frac{2}{3}$-exact *for f with respect to μ iff A is weakly $\frac{2}{3}$-exact for f with respect to μ and $P_A(x) \in \{0,1\}$ for all inputs $x \in \{0,1\}^N$.*

The second definition most closely mimics the behavior of a classical deterministic algorithm: the input x fully determines the output bit (even on $x \notin D$) and the algorithm gives correct output $f(x)$ for "most" x. The first definition is more liberal: here we only require this "input-determines-output" behavior to occur for a μ-fraction of at least $\frac{2}{3}$ of the inputs where the algorithm gives the correct output $f(x)$. Note that a strongly $\frac{2}{3}$-exact algorithm for f with respect

to μ actually computes some total function $g : \{0,1\}^N \to \{0,1\}$ with success probability 1, namely the function $g(x) = P_A(x)$.

These two definitions lead to a weak and a strong quantum counterpart to the classical distributional complexity $D^{\mu}_{1/3}(f)$: let $Q^{\mu}_{WE}(f)$ and $Q^{\mu}_{SE}(f)$ denote the minimal complexity among all weakly and strongly $\frac{2}{3}$-exact algorithms for f with respect to μ, respectively. Note that $Q^{\mu}_{WE}(f) \leq Q^{\mu}_{SE}(f)$ for all f and μ. We can now state two potential quantum versions of the Yao principle:

- Weak quantum Yao principle: $Q_2(f) \stackrel{?}{=} \max_{\mu} Q^{\mu}_{WE}(f)$
- Strong quantum Yao principle: $Q_2(f) \stackrel{?}{=} \max_{\mu} Q^{\mu}_{SE}(f)$

In this paper we investigate to what extent these two principles hold.

1.2 Results

Our results are threefold. First, we prove that both principles hold in the '$\leq$'-direction, for all f:

- $Q_2(f) \leq \max_{\mu} Q^{\mu}_{WE}(f) \leq \max_{\mu} Q^{\mu}_{SE}(f)$

The proof of the first inequality is analogous to the classical game-theoretic proof. We emphasize that this result is perfectly general, and applies to all computational models to which the classical Yao principle applies.

In order to investigate to what extent the '$\geq$'-directions of these two quantum Yao principles hold, we instantiate our complexity measures to the query complexity setting. Our second result is an exponential gap between $Q_2(f)$ and $Q^{\mu}_{SE}(f)$ for the query complexity of Simon's problem [16]:

- There exist f and μ such that $Q_2(f)$ is exponentially smaller than $Q^{\mu}_{SE}(f)$.

This shows that the strong quantum Yao principle is false.

Thirdly, we prove that the weak quantum Yao principle holds up to a constant factor for the query complexity of all *symmetric* functions:

- $Q_2(f) = \Theta\left(\max_{\mu} Q^{\mu}_{WE}(f)\right)$ for all symmetric f

For this result we first construct a quantum algorithm that can determine the N-bit input x *with certainty* in $O(\sqrt{kN})$ queries if k is a known upper bound on the Hamming weight of x. We then use that algorithm to construct, for every symmetric function f and distribution μ, a quantum algorithm that computes $f(x)$ with certainty for "most" inputs x. In addition to this result for symmetric functions, we also show that for a particular *monotone* non-symmetric function f (the AND-OR tree), the $\max_{\mu} Q^{\mu}_{WE}(f)$ complexity lies in between the best known bounds for $Q_2(f)$. The gist of this third batch of results is that most known quantum algorithms that are somehow based on Grover's algorithm can be made weakly $\frac{2}{3}$-exact. This may actually be the main contribution of this paper.

2 Preliminaries

In this section we formalize the notion of query complexity, define several complexity measures, and state Von Neumann's minimax theorem.

2.1 Query Complexity

We assume familiarity with classical computation theory and briefly sketch the basics of quantum computation; an extensive introduction may be found in the book by Nielsen and Chuang [12]. Quantum algorithms operate on *qubits* as opposed to bits in classical computers. The state of an m-qubit quantum system can be written as $|\phi\rangle = \sum_{i\in\{0,1\}^m} \alpha_i |i\rangle$, where $|i\rangle$ denotes the basis state i, which is a classical m-bit string. The α_i's are complex numbers known as the *amplitudes* of the basis states $|i\rangle$ and we require $\sum_{i\in\{0,1\}^m} |\alpha_i|^2 = 1$. Mathematically, the state of a system is thus described by a 2^m-dimensional complex unit vector. If we measure the value of $|\phi\rangle$, then we will see the basis state $|i\rangle$ with probability $|\alpha_i|^2$, after which the system collapses to $|i\rangle$. Operations that are not measurements correspond to *unitary transformations* on the vector of amplitudes.

In the *query model* of computation, the goal is to compute some function $f : D \to \{0,1\}$ on an input $x \in D \subseteq \{0,1\}^N$, using as few accesses ("queries") to the N input bits as possible. It is by now standard to formalize a quantum query as an application of a unitary transformation O that acts as $O|i,b,z\rangle = |i, b \oplus x_i, z\rangle$. Here $i \in \{1,\ldots,N\}$, $b \in \{0,1\}$, $\oplus$ denotes the exclusive-or function, and z denotes the workspace of the algorithm, which is not affected by O. A T-query quantum algorithm A then has the form $A = U_T O U_{T-1} O \cdots U_1 O U_0$, with each U_i a fixed unitary transformation independent of the input x. Algorithm A is assumed to start in the all-zero state $|0\ldots0\rangle$, and its output (0 or 1) is obtained by measuring the rightmost bit of its final state $A|0\ldots0\rangle$. The *acceptance probability* $P_A(x)$ of a quantum algorithm A is defined as the probability of getting output 1 on input x. Its *success probability* $S_A(x)$ is the probability of getting the correct output $f(x)$ on input x.

A quantum algorithm A computes a function $f : D \to \{0,1\}$ *exactly* if $S_A(x) = 1$ for all inputs $x \in D$. Algorithm A computes f with *bounded-error* if $S_A(x) \geq \frac{2}{3}$ for all $x \in D$. We use $Q_E(f)$ and $Q_2(f)$ to denote the minimal number of queries required by exact and bounded-error quantum algorithms for f, respectively. These complexities are the quantum versions of the classical deterministic and bounded-error decision tree complexities $D(f)$ and $R_2(f)$, respectively. For completeness, we repeat our two alternative quantum versions of the classical distributional complexity $D^\mu(f)$ from the introduction. Let μ be a probability distribution on the set of all possible inputs. An algorithm A is *weakly $\frac{2}{3}$-exact for f with respect to μ* if $\mu(\{x \mid P_A(x) = f(x)\}) \geq \frac{2}{3}$, and A is *strongly $\frac{2}{3}$-exact for f with respect to μ* if A is weakly $\frac{2}{3}$-exact for f with respect to μ and $P_A(x) \in \{0,1\}$ for all $x \in \{0,1\}^N$. By $Q^\mu_{SE}(f)$ and $Q^\mu_{WE}(f)$ we denote the minimal number of queries needed by strongly and weakly $\frac{2}{3}$-exact quantum algorithms for f with respect to μ, respectively. Note that $Q^\mu_{WE}(f) \leq Q^\mu_{SE}(f)$ for all f and μ, hence in particular $\max_\mu Q^\mu_{WE}(f) \leq \max_\mu Q^\mu_{SE}(f)$.

One of the first quantum algorithms operating in the query model is Grover's search algorithm [8, 4]. Let $|x|$ denote the Hamming weight (number of 1's) in the input x, and let x_i denote the ith bit of x. If $t = |x| > 0$ then Grover's algorithm uses $\frac{\pi}{4}\sqrt{N/t}$ queries and with high probability outputs an i such that $x_i = 1$. If $|x| = 0$ then the algorithm always outputs 'no solutions'. Brassard, Høyer, Mosca, and Tapp [4] gave an exact version of Grover's algorithm that accomplishes the same task with probability 1 if $|x|$ is known.

A function $f : \{0,1\}^N \to \{0,1\}$ is *symmetric* if its value $f(x)$ depends only on $|x|$. For such f, define $f_k = f(x)$ where $|x| = k$. In [3] it is proven that $Q_2(f) = \Theta(\sqrt{N(N - \Gamma(f))})$, where $\Gamma(f) = \min\{|2k - N - 1| \mid f_k \neq f_{k+1} \text{ and } 0 \leq k \leq N - 1\}$. Informally, the quantity $\Gamma(f)$ (introduced by Paturi [14]) measures the length of the interval around Hamming weight $\frac{N}{2}$ where f is constant. A symmetric function f is a *threshold* function if there is a $0 < t \leq N$, such that $f(x) = 1$ iff $|x| \geq t$. Note that for $t \leq N/2$ we have $Q_2(f) = \Theta(\sqrt{tN})$ as a direct consequence of the bound for symmetric functions. A function $f : \{0,1\}^N \to \{0,1\}$ is *monotone* if $(\forall i\ x_i \leq y_i) \Rightarrow f(x) \leq f(y)$.

2.2 Von Neumann's Minimax Theorem

The book by Owen [13] provides an excellent introduction to game theory. Here we only state Von Neumann's famous minimax theorem [10]. Consider a two-player, zero-sum game with payoff matrix P. Player 1 wants to maximize the payoff, player 2 wants to minimize. Both players have available a finite set of *pure* strategies. If player 1 plays pure strategy i and player 2 plays pure strategy j, then the payoff is $P_{ij} = e_i^T P e_j$, where e_i and e_j are the appropriate unit vectors and superscript-T denotes vector transposition. In addition, they may also use a *mixed* strategy. This is a probability distribution over the set of pure strategies, modeled by a vector of non-negative reals that sum to 1. If player 1 plays mixed strategy ρ and player 2 plays mixed strategy μ, then the expected payoff of the game is $\rho^T P \mu$. The minimax theorem states that the maximal payoff that player 1 can assure if he can base ρ on μ, equals the minimal payoff that player 2 can assure if he can base μ on ρ:

$$\min_{\mu} \max_{\rho} \rho^T P \mu = \max_{\rho} \min_{\mu} \rho^T P \mu.$$

Without loss of generality the "inner" choices can be assumed to be pure strategies, hence

$$\min_{\mu} \max_{i} e_i^T P \mu = \max_{\rho} \min_{j} \rho^T P e_j.$$

As mentioned in the introduction, the classical Yao principle is an easy consequence of this theorem. In the next section we use it to prove one half of the *quantum* Yao principle.

3 Proof of One Half of the Quantum Yao Principle

Here we prove $Q_2(f) \leq \max_\mu Q^\mu_{WE}(f)$. The proof is similar to the derivation of the classical Yao principle, but the details are a bit more messy.

Theorem 1. *For all $f : D \to \{0,1\}$, with D finite, $Q_2(f) \leq \max_\mu Q^\mu_{WE}(f)$.*

Proof. Consider the (infinite) set of all quantum algorithms of complexity $\leq \max_\mu Q^\mu_{WE}(f)$. Let i be any algorithm from this set, and $x \in D$ an input. Consider the quantity $\lfloor S_i(x) \rfloor$, which is 1 if algorithm i computes $f(x)$ with success probability 1, and which is 0 otherwise. Call algorithms i and j *similar* if $\lfloor S_i(x) \rfloor = \lfloor S_j(x) \rfloor$ for all $x \in D$. In this way, similarity is an equivalence relation on the set of all quantum algorithms of complexity $\leq \max_\mu Q^\mu_{WE}(f)$. Note that similarity partitions this set into at most $2^{|D|}$ equivalence classes. From each equivalence class, we choose as a representative an algorithm from that class with the least complexity.

Now consider the game in which player 1 wants to compute f, and as pure strategies he has available the (finite) set of representatives of the equivalence classes. Player 2 is an adversary that chooses hard inputs $x \in D$ to f. Let S be the matrix of success probabilities ($S_{ix} = S_i(x)$). Define the payoff matrix as $P_{ix} = \lfloor S_{ix} \rfloor$. Now consider the quantity $\max_i e_i^T P \mu$. This represents the μ-fraction of inputs on which the best weakly $\frac{2}{3}$-exact quantum algorithm for f with respect to that μ is correct. This quantity is at least $\frac{2}{3}$ for all μ, since we've been considering all quantum algorithms of complexity up to $\max_\mu Q^\mu_{WE}(f)$. From the minimax theorem we now obtain

$$\frac{2}{3} \leq \min_\mu \max_i e_i^T P \mu = \max_\rho \min_x \rho^T P e_x \leq \max_\rho \min_x \rho^T S e_x.$$

Here the last term can be interpreted as the success probability of a quantum algorithm formed by a probability distribution ρ over the set of representatives of the equivalence classes (such a distribution can be easily realized in a quantum algorithm using a superposition). By the above inequality, this algorithm has success probability $\geq \frac{2}{3}$ for all inputs $x \in D$. Since it is a probability distribution over algorithms of complexity $\leq \max_\mu Q^\mu_{WE}(f)$, its complexity is at most $\max_\mu Q^\mu_{WE}(f)$. Hence $Q_2(f) \leq \max_\mu Q^\mu_{WE}(f)$. □

Corollary 1. *For all $f : D \to \{0,1\}$, with D finite, $Q_2(f) \leq \max_\mu Q^\mu_{SE}(f)$.*

We again emphasize that this result applies to all computational models where the classical Yao principle applies.

4 A Counterexample for the Strong Quantum Yao Principle

From here on, we will instantiate our complexity measures to the query complexity setting. Ambainis [1] has proven that for almost all Boolean functions f we have $Q_2(f) = \Omega(N)$. This result immediately implies that both the strong and weak quantum Yao principle hold up to a constant factor for almost all Boolean functions in the query complexity setting.

However, the strong quantum Yao principle does not hold in general. Below we exhibit a function f and distribution μ where $Q_2(f)$ is exponentially less than $Q^{\mu}_{SE}(f)$. The function is Simon's problem [16], and our separation is based on Simon's classical lower bound combined with the result that classical and quantum query complexity are polynomially related for all total functions [3].

Theorem 2. *There exist a problem f on $N = n2^n$ bits and a distribution μ such that $Q_2(f) = O(n^2)$ and $Q^{\mu}_{SE}(f) = \Omega(2^{\frac{n}{8}})$.*

Proof. Consider Simon's problem: given a function $\phi : \{0,1\}^n \to \{0,1\}^n$ with the promise that there is an $s \in \{0,1\}^n$ such that $\phi(a) = \phi(b)$ iff $a \oplus b = s$, decide whether $s = 0$ or not. This function ϕ is given as an input x of $N = n2^n$ bits, using n 1-bit entries for each function value $\phi(\cdot)$. The input bits can be queried in the usual way. Using Simon's bounded-error quantum algorithm, this problem can be solved in $O(n^2)$ queries, and hence $Q_2(Simon) = O(n^2)$. Now define a distribution μ which uniformly places half the total weight on inputs with $s = 0$ and half the total weight on inputs with $s \neq 0$. Simon proved that under this distribution, any classical algorithm that is correct on a fraction $\geq \frac{2}{3}$ requires $\Omega(\sqrt{2^n})$ queries. Now take any strongly $\frac{2}{3}$-exact T-query quantum algorithm A for this problem, then A computes some *total* function g. Since $D(g) = O(Q_E(g)^4)$ [3], there exists a deterministic classical algorithm that computes g using $O(T^4)$ queries. But this classical algorithm is then correct on a μ-fraction $\frac{2}{3}$ of all Simon inputs. Simon's lower bound on classical algorithms now implies that $O(T^4) = \Omega(\sqrt{2^n})$, and hence $Q^{\mu}_{SE}(Simon) = \Omega(2^{\frac{n}{8}})$. □

5 A Positive Result for the Weak Quantum Yao Principle

In this section we show that the weak quantum Yao principle holds for all symmetric functions. We start with the special case of threshold functions.

5.1 Equality up to a Constant Factor for Threshold Functions

Consider a threshold function with threshold $t \leq N/2$. For every distribution μ, we will exhibit a weakly $\frac{2}{3}$-exact quantum algorithm for f with respect to μ with $O(\sqrt{tN})$ queries. This, together with Theorem 1 and the known fact that $Q_2(f) = \Theta(\sqrt{tN})$ for threshold functions f [3], gives the desired result.

Note that given a threshold function $f : \{0,1\}^N \to \{0,1\}$ with threshold t, in order to be sure that $f(x) = 1$, it suffices to find at least t 1's in the input. The crucial idea behind our algorithm is that if the number of 1's in the input is large enough, then for each distribution μ over the inputs, we can pick a substantially smaller part of the input such that there are between t and $100t$ 1's in this sub-part for a large μ-fraction of the inputs. This idea is formally stated in the following technical lemma.[1]

[1] We need the condition $i \geq 10$ in this lemma in order to be able to approximate the hypergeometric distribution by a binomial distribution with sufficient accuracy.

Lemma 1. *Let t be a threshold, μ a probability distribution over the $x \in \{0,1\}^N$, and i an integer such that $10 \leq i \leq \log N - \log t - 1$. Denote the event $t2^i \leq |x| \leq t2^{i+1}$ by I, and let $x \wedge y$ denote the bitwise AND of x and y. There is a $y \in \{0,1\}^N$ with $|y| = \min\{\frac{10N}{2^i}, N\}$, such that $\Pr_\mu[t \leq |x \wedge y| \leq 100t \mid I] > 0.7$.*

Proof. We assume $\frac{10N}{2^i} \leq N$, for otherwise the lemma trivially holds. Consider any $x \in \{0,1\}^N$ with $t2^i \leq |x| \leq t2^{i+1}$. We claim that if we pick a $y \in \{0,1\}^N$ with $|y| = \frac{10N}{2^i}$ uniformly at random, then $\Pr[t \leq |x \wedge y| \leq 100t] > 0.7$. To prove this claim, note that $|x \wedge y|$ is hypergeometrically distributed, with expected value $E(|x \wedge y|) = \frac{|x||y|}{N} \in [10t, 20t]$. By Markov's inequality it follows directly that $\Pr[|x \wedge y| > 100t] \leq 0.2$.

We can approximate the above distribution with a binomial distribution since the number of draws is small compared to the size of the sample space, see e.g. [11], and we shall henceforth treat $|x \wedge y|$ as if it were binomially distributed, with success probability $\theta = \frac{|x|}{N}$ and number of draws $n = |y|$. To bound $\Pr[|x \wedge y| < t]$, we use the Chernoff bound as explained in [9, pp.67-73]:

$$\Pr[|x \wedge y| < (1-\delta)E(|x \wedge y|)] < e^{\frac{-\delta^2 E(|x \wedge y|)}{2}}.$$

Choosing $\delta = \frac{9}{10}$, we obtain $\Pr[|x \wedge y| < t] < e^{-\frac{810t}{200}} < 0.1$. Combining the previous two inequalities, it then follows that $\Pr[t \leq |x \wedge y| \leq 100t] > 0.7$, which proves the above claim.

Now imagine a matrix whose rows are indexed by the x satisfying $t2^i \leq |x| \leq t2^{i+1}$ and whose columns are indexed by the $M = \binom{N}{|y|}$ different y of weight $|y| = \frac{10N}{2^i}$. We give the (x,y) entry of this matrix value $\mu(x|I)$ if $t \leq |x \wedge y| \leq 100t$ and value 0 otherwise. By the above claim, each row will contain at least 70% non zero entries, so the sum of the entries of the x-row is at least $0.7M\mu(x|I)$. Hence, the sum of all entries in the matrix is equal to $\sum_x 0.7M\mu(x|I) = 0.7M$. But then there must be a column with $\mu(\cdot \mid I)$-weight at least 0.7. The y corresponding to this column is the y we are looking for in this lemma. □

We will use the fact stated in the previous lemma to successively search for t 1's in exponentially smaller parts of the inputs, assuming the presence of increasingly more 1's in the original input. The following lemma states that this searching can be done efficiently:

Lemma 2. *There exists a quantum algorithm that can find all the 1's in an input x of size N with probability 1, using at most $\frac{\pi}{2}\sqrt{kN}$ queries, if k is a known upper bound on the number of 1's in x.*

Proof. Assume an upper bound k on the number of 1's in x. Suppose we run the exact version of Grover's algorithm assuming $|x| = k$. Either we find a solution, in which case we can remove that solution from the search space, lower our upper bound k by 1 and continue; or we do not find a solution, in which case we know that $|x|$ must be less than k, so we can safely lower our upper bound k by 1

and continue. Accordingly, it easily follows by induction on k that Algorithm 1 below finds all $|x|$ solutions with certainty. The number of queries it uses is

$$\sum_{i=1}^{k} \frac{\pi}{4}\sqrt{\frac{N}{i}} \leq \frac{\pi}{4}\sqrt{N}\int_0^k \frac{\mathrm{d}i}{\sqrt{i}} = \frac{\pi}{2}\sqrt{kN}.$$

□

Algorithm 1

for $i = k$ down to 1 **do**
  Apply the exact version of Grover's algorithm, assuming
    there are i solutions.
  if a solution has been found **then**
    mark its index as a zero in the search space
  end if
end for
output the positions of all solutions found

We are now ready to prove an upper bound on $Q^{\mu}_{WE}(f)$:

Lemma 3. *For threshold function f with threshold t, and for every distribution μ, we have $Q^{\mu}_{WE}(f) = O(\sqrt{tN})$.*

Proof. Fix a distribution μ. Invoking Lemmas 1 and 2, our algorithm (Algorithm 2 below) is as follows. First we count the number of 1's in the input using Algorithm 1, assuming an upper bound of $2^{10}t$ 1's. If after that we haven't found at least t 1's yet, then we successively assume that there are between $t2^i$ and $t2^{i+1}$ 1's in the input, with i going up from 10 to $\log N - \log t - 1$. For each of these assumptions, we search a smaller part of the input. If we have reached the i for which $t2^i \leq |x| \leq t2^{i+1}$, then Lemma 1 guarantees that for a large μ-fraction of those inputs we can find a small sub-part containing between t and $100t$ 1's. We then count the number of 1's in this sub-part using Algorithm 1. This algorithm will be correct on all inputs x with $|x| < t$ and will produce a correct answer on at least a μ-fraction 0.7 of all inputs x with $|x| \geq t$ as guaranteed by Lemma 1. Hence it will be correct on a μ-fraction at least $\mu(\{x \mid |x| < t\}) + 0.7(1 - \mu(\{x \mid |x| < t\}) \geq 0.7$. Furthermore, its query complexity is

$$O(\sqrt{tN}) + \sum_{i=10}^{\log N - \log t - 1} O\left(\sqrt{\frac{tN}{2^i}}\right) = O(\sqrt{tN}),$$

where the first term corresponds to the cost of searching the entire space once with a small upper bound, and the summation corresponds to searching consecutively smaller sub-parts $y^{(i)}$. □

Algorithm 2

Count the number of 1's in the input using Algorithm 1, assuming an upper bound of $2^{10}t$ 1's
if at least t 1's are found **then**
  output 1
end if
for $i = 10$ to $\log N - \log t - 1$ **do**
  Let $y^{(i)} \in \{0,1\}^N$ be a string of weight $\min\{N, \frac{10N}{2^i}\}$ satisfying Lemma 1
  Using Algorithm 1, count the number of solutions in the sub-part
    of the input induced by $y^{(i)}$, assuming an upper bound of $100t$ 1's.
  if at least t 1's are found **then**
    output 1
  end if
end for
output 0

Recall that for threshold functions $f : \{0,1\}^N \to \{0,1\}$ with threshold $t \leq N/2$, we have $Q_2(f) = \Theta(\sqrt{tN})$. By Theorem 1 it then follows that $\max_\mu Q^\mu_{WE}(f) = \Omega(\sqrt{tN})$. In combination with Lemma 3, this yields:

Lemma 4. *For all threshold functions $f : \{0,1\}^N \to \{0,1\}$ with $t \leq N/2$*

$$Q_2(f) = \Theta\left(\max_\mu Q^\mu_{WE}(f)\right) = \Theta\left(\sqrt{tN}\right).$$

5.2 Equality up to a Constant Factor for Symmetric Functions.

With the result about threshold functions in mind, we can easily prove that the quantum Yao principle holds for all symmetric functions as well.

Theorem 3. *For all symmetric functions $f : \{0,1\}^N \to \{0,1\}$*

$$Q_2(f) = \Theta\left(\max_\mu Q^\mu_{WE}(f)\right) = \Theta\left(\sqrt{N(N - \Gamma(f))}\right).$$

We give an informal sketch of the proof whose details are straightforward. Firstly, note that $\Gamma(f)$ measures the length of the interval around Hamming weight $\frac{N}{2}$ where f is constant, so in order to compute $f(x)$ it suffices to know $|x|$ exactly if $|x| \in [0, \frac{N-\Gamma(f)}{2})$ or $|x| \in (\frac{N+\Gamma(f)-2}{2}, N]$, or to know that $|x| \in [\frac{N-\Gamma(f)}{2}, \frac{N+\Gamma(f)-2}{2}]$ otherwise. Using the threshold algorithm from Section 5.1 twice, we can, at a cost of $O(\sqrt{N(N-\Gamma(f))})$ queries, compute which of three intervals $|x|$ is in. If $|x|$ is in the interval of length $\Gamma(f)$ around $\frac{N}{2}$ where f is constant we are done. In both other cases we now in effect have an upper bound on the number of 1's in the input, and we can use Algorithm 1 to exactly count the number of 1's, again using $O(\sqrt{N(N-\Gamma(f))})$ queries.

5.3 A Result for the AND-OR Tree

Above we proved that the weak quantum Yao principle holds (up to a constant factor) for all *symmetric* functions. A similar result might be provable for all *monotone* functions. In this section we state a preliminary result in this direction, namely that the known upper and lower bounds on the $Q_2(f)$-complexity of the 2-level *AND-OR tree* carry over to weakly $\frac{2}{3}$-exact quantum algorithms. This monotone but non-symmetric function is the AND of $\sqrt{N}$ independent ORs of $\sqrt{N}$ variables each. In the sequel, we use AO to denote this N-bit AND-OR tree.

No tight characterization of $Q_2(AO)$ is known, but Buhrman, Cleve, and Widgerson [5] proved $Q_2(AO) = O(\sqrt{N}\log N)$ via a recursive application of Grover's algorithm. Using a result about efficient error-reduction in quantum search from [6], this can be improved to $Q_2(AO) = O(\sqrt{N\log N})$. This nearly matches Ambainis' lower bound of $\Omega(\sqrt{N})$ [2]. Note that Ambainis' bound together with our Theorem 1 immediately gives the lower bound $\max_\mu Q^\mu_{WE}(AO) = \Omega(\sqrt{N})$. Using the same techniques as in the previous section one can show that the best known *upper* bound carries over to weakly $\frac{2}{3}$-exact algorithms. Due to space constraints we omit the proof, which may be found at the Los Alamos preprint server at `http://xxx.lanl.gov/abs/quant-ph/0109070`.

Theorem 4. *For every distribution μ we have $Q^\mu_{WE}(AO) = O(\sqrt{N\log N})$.*

6 Summary and Open Problems

In this paper we investigated to what extent quantum versions of the classical Yao principle hold. We formulated a strong and a weak version of the quantum Yao principle, showed that both hold in one direction, falsified the other direction for the strong version, and proved the weak version for the query complexity of all symmetric functions.

The main question left open by this research is the general validity of the weak quantum Yao principle. On the one hand, we may be able to find a counterexample to the weak principle as well, perhaps based on the query complexity of the *order-finding problem*. Shor showed that the order-finding problem can be solved by a bounded-error quantum algorithm using $O(\log N)$ queries [15]. Using Cleve's $\Omega(N^{1/3}/\log N)$ lower bound on classical algorithms for order-finding [7], we might be able to exhibit a μ such that any weakly $\frac{2}{3}$-exact quantum algorithm for f with respect to μ requires $N^{\Omega(1)}$ queries, as it seems hard to construct weakly $\frac{2}{3}$-exact quantum algorithms for this problem.

On the other hand, we may try to extend the class of functions for which we know the weak quantum Yao principle *does* hold. A good starting point here might be the class of all *monotone* functions. We discussed one such function, the 2-level AND-OR tree, in Section 5.3. Unfortunately, at the time of writing no general characterization of the $Q_2(f)$ complexity of monotone functions is known.

Acknowledgments

We thank Harry Buhrman for initiating this research, for coming up with the counterexample of Theorem 2, and for useful comments on a preliminary version of this paper. We thank him and Peter Høyer for their contributions to an initial proof of the weak quantum Yao principle for the OR function, which forms the basis for the current proof of Theorem 3. We also thank Leen Torenvliet and Chris Klaassen for useful discussions.

References

1. A. Ambainis. A note on quantum black box complexity of almost all Boolean functions. In *Information Processing Letters 71*, pages 5–7, 1999. quant-ph/9811080.
2. A. Ambainis. Quantum lower bounds by quantum arguments. In *Proceedings of 32nd ACM STOC*, pages 636–643, 2000. quant-ph/0002066.
3. R. Beals, H. Buhrman, R. Cleve, M. Mosca, and R. de Wolf. Quantum lower bounds by polynomials. In *Proceedings of 39th IEEE FOCS*, pages 352–361, 1998. quant-ph/9802049.
4. G. Brassard, P. Høyer, M. Mosca, and A. Tapp. Quantum amplitude amplification and estimation. quant-ph/0005055. To appear in Quantum Computation and Quantum Information: A Millennium Volume, AMS Contemporary Mathematics Series, 15 May 2000.
5. H. Buhrman, R. Cleve, and A. Wigderson. Quantum vs. classical communication and computation. In *Proceedings of 30th ACM STOC*, pages 63–68, 1998. quant-ph/9802040.
6. H. Buhrman, R. Cleve, R. de Wolf, and Ch. Zalka. Bounds for small-error and zero-error quantum algorithms. In *Proceedings of 40th IEEE FOCS*, pages 358–368, 1999. cs.CC/9904019.
7. R. Cleve. The query complexity of order-finding. In *Proceedings of 15th IEEE Conference on Computational Complexity*, pages 54–59, 2000. quant-ph/9911124.
8. L. K. Grover. A fast quantum mechanical algorithm for database search. In *Proceedings of 28th ACM STOC*, pages 212–219, 1996. quant-ph/9605043.
9. R. Motwani and P. Raghavan. *Randomized Algorithms*. Cambridge University Press, 1995.
10. J. von Neumann and O. Morgenstern. *Theory of Games and Economic Behavior*. Princeton University Press, 1947.
11. W. L. Nicholson. On the normal approximation to the hypergeometric distribution. *Annals of Mathematical Statistics*, 27:471–483, 1956.
12. M. A. Nielsen and I. L. Chuang. *Quantum Computation and Quantum Information*. Cambridge University Press, 2000.
13. G. Owen. *Game Theory*. Academic Press, second edition, 1982.
14. R. Paturi. On the degree of polynomials that approximate symmetric Boolean functions. In *Proceedings of 24th STOC*, pages 468–474, 1992.
15. P. W. Shor. Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. *SIAM Journal on Computing*, 26(5):1484–1509, 1997. Earlier version in FOCS'94. quant-ph/9508027.
16. D. Simon. On the power of quantum computation. *SIAM Journal on Computing*, 26(5):1474–1483, 1997. Earlier version in FOCS'94.
17. A. C-C. Yao. Probabilistic computations: Toward a unified measure of complexity. In *Proceedings of 18th IEEE FOCS*, pages 222–227, 1977.