Quantum and Classical Tradeoffs[1]

Yaoyun Shi

Department of Electrical and Computer Engineering
The University of Michigan
Ann Arbor, MI 48109-2122, USA
E-mail: shiyy@eecs.umich.edu.

Abstract

We initiate the study of quantifying the quantumness of a quantum circuit by the number of gates that do not preserve the computational basis, as a means to understand the nature of quantum algorithmic speedups. Intuitively, a reduction in the quantumness requires an increase in the amount of classical computation, thus giving a "quantum and classical tradeoff".

In this paper we present two results on this measure of quantumness. The first gives almost matching upper and lower bounds on the question: "what is the minimum number of non-basis-preserving gates required to generate a good approximation to a given state". This question is the quantum analogy of the following classical question, "how many fair coins are needed to generate a given probability distribution", which was studied and resolved by Knuth and Yao in 1976. Our second result shows that any quantum algorithm that solves Grover's Problem of size n using k queries and ℓ levels of non-basis-preserving gates must have $k \cdot \ell = \Omega(n)$.

Keywords: Quantum computation, quantum and classical tradeoff, quantum state generation, quantum lower bound, Grover's Algorithm.

---

1 Introduction.

The importance of quantum computing lies in the possibility that quantum mechanical algorithms may be dramatically more efficient than the best classical algorithms. In order to understand the nature of quantum speedup, it is important to identify features of quantum computing that are uniquely quantum and investigate their roles in quantum speedups. One example of this kind of study was taken by Jozsa and Linden [8], which relates the amount of entanglement during the computation to the difficulty of simulating the computation. Our work is alone a similar line, but instead of entanglement, we study another feature of quantum computing: the number of gates that do not preserve the computational basis.

It is well known that any classical computation can be carried out, without much sacrifice in the efficiency, using classical reversible gates, such as the Toffoli gate. In order to have nontrivial quantum speedup, gates that do not preserve the computational basis must be used. Furthermore, the more such gates involved, the more difficult a straightforward classical simulation is.

Recall that the state space of a qubit has an orthonormal basis, denoted by $\{|0\rangle, |1\rangle\}$, that is fixed a priori and called the *computational basis*. The computational basis for the state space of n qubits is the tensor products of their computational bases. Each qubit of a quantum computer is assumed to start in the computational base state $|0\rangle$. We follow this convention throughout this paper.

Let us formally call a gate G *basis-changing* if there exist two computational base vectors $|\phi\rangle$ and $|\psi\rangle$, such that $|\langle\phi|G|\psi\rangle| < 1$. If G is not basis-changing, G is said to be *basis-preserving*. An important example of a basis-changing gate is the Hadamard gate

$$H \stackrel{\text{def}}{=} \frac{1}{\sqrt{2}} \left(|0\rangle\langle0| + |1\rangle\langle0| + |0\rangle\langle1| - |1\rangle\langle1|\right).$$

It is well known (e.g. [11]) that any quantum circuit can be efficiently simulated by Toffoli and Hadamard gates. It is also easy to observe that a quantum circuit that uses k Hadamard gates, together with some other basis-preserving gates, can be simulated straightforwardly by a deterministic algorithm with a 2^k factor of slow-down.

Hence, it appears natural to quantify the amount of "quantumness" of a quantum circuit by the number of basis-changing gates, and to investigate the tradeoffs between this amount of quantumness with the best possible quantum speedup. This is precisely the theme of our investigation.

Many interesting questions can be asked in this diction. In particular, we present two results in this paper. The first is on the following question: what is the minimum number of basis-changing gates required to generate a good approximation of a given quantum state? This is in analogy to the following classical question: what is the least number of fair coins required to produce a given probability distribution? In 1976, Knuth and Yao [10] solved this problem completely: the minimum expected number of fair coins needed is equal to the Shannon entropy of the distribution plus a small constant. We find that the answer to the quantum problem is similar.

The second result investigates the quantum-classical tradeoffs in solving Grover's Problem [7], also called the Unstructured Search Problem, an important and well studied problem in quantum computing. We prove that any quantum algorithm that solves Grovers' Problem of size n using k queries and ℓ levels of basis-changing gates must have $k \cdot \ell = \Omega(n)$. This tradeoff relation is tight.

We shall present these two results in the following two sections, followed by a discussion of open problems.

2 Quantum State Generation.

A classical problem studied by Knuth and Yao in [10] is the following: how many independent 0/1 variables are needed in order to generate a given probability distribution? They prove that the minimum expected number of coins is precisely $H(\mathcal{D})$, the Shannon entropy of $\mathcal{D}$, plus some small constant. In this section we

study the quantum analog of the question: how many basis-changing gates are needed in order to generate a good approximation of a given quantum state?

For a quantum state $|\phi\rangle$, denote by $H(\phi)$ the Shannon entropy of the probability distribution obtained from measuring $|\phi\rangle$ in the computational basis. We prove both upper and lower bounds to the quantum problem in terms of $H(\phi)$.

2.1 Upper bound.

We first consider a special case, and then reduce the general case to it.

Lemma 2.1. *Let $|\phi\rangle$ be a state over n qubits with nonnegative amplitudes, and $\epsilon > 0$ be a real constant. Then there is a quantum algorithm that uses $O(n \log \frac{n}{\epsilon})$ basis-changing gates and maps $|0\rangle^{\otimes n}$ to a state $|\phi'\rangle$, such that $\||\phi'\rangle - |\phi\rangle\| \leq \epsilon$.*

The algorithm is along the lines of the algorithm in [9] for approximating an operator that maps $|q\rangle \otimes |0\rangle^{\otimes n}$ to $|q\rangle \otimes \left(\sum_{j=0..q-1} \frac{1}{\sqrt{q}} |j\rangle \right)$.

Proof. Suppose

$$|\phi\rangle = \sum_{y \in \{0,1\}^n} \sqrt{p_y} |y\rangle.$$

For $0 \leq t \leq n$, and $y \in \{0,1\}^t$, let

$$q_y \stackrel{\text{def}}{=} \sum_{z \in \{0,1\}^{n-t}} p_{yz}, \quad \text{and,} \quad |\phi_t\rangle \stackrel{\text{def}}{=} \sum_{y \in \{0,1\}^t} \sqrt{q_y} |y\rangle \otimes |0\rangle^{\otimes n-t}.$$

Then $|\phi_0\rangle = |0\rangle^{\otimes n}$, and $|\phi_n\rangle = |\phi\rangle$.

The algorithm has n stages. At the ith stage, the algorithm transforms $|\phi_i\rangle$ to a state $|\phi'_{i+1}\rangle$ such that $\||\phi'_{i+1}\rangle - \phi_{i+1}\rangle\| \leq \epsilon/n$, and uses $\ell \stackrel{\text{def}}{=} \lceil \log \frac{n}{\epsilon} \rceil$ basis-changing gates. This can be done by the following.

1. For each $y \in \{0,1\}^t$, let $\theta_y \stackrel{\text{def}}{=} \arccos(\sqrt{q_{y0}/q_y})$. Compute on the ancilla and using Toffoli gates the first ℓ bits of θ_y/π, $a_{y,1}, a_{y,2}, \cdots, a_{y,\ell}$. This maps $|\phi_t\rangle \otimes |0\rangle^{\otimes \ell}$ to

$$\sum_{y \in \{0,1\}^t} \sqrt{q_y} |y\rangle \otimes |0\rangle^{\otimes n-t} \otimes |a_{y,1}, a_{y,2}, \cdots, a_{y,\ell}\rangle.$$

2. Denote by $R(\theta)$ the single qubit rotation operator of an angle θ. Let $\theta'_y = \sum_{s=1..\ell} a_{y,s} \pi/2^s$. For $s = 1..\ell$, apply the Controlled-$R(a_{y,s}\pi/2^s)$ gate with the sth qubit in the ancilla as the control qubit and the $(t+1)^{\text{th}}$ qubit in the output state as the destination qubit. This results in mapping

$$\sum_{y \in \{0,1\}^t} \sqrt{q_y} \, |y\rangle \otimes |0\rangle \ \rightarrow \ \sum_{y \in \{0,1\}^t} \sqrt{q_y} \, |y\rangle \otimes R(\theta'_y)|0\rangle.$$

Since $\|R(\theta'_y) - R(\theta_y)\| \leq \pi/2^\ell$, and

$$|\phi_{t+1}\rangle = \sum_{y \in \{0,1\}^t} \sqrt{q_y} |y\rangle \otimes R(\theta_y)|0\rangle \otimes |0\rangle^{\otimes n-t-1},$$

the resulted vector $|\phi'_{t+1}\rangle$ satisfies

$$\||\phi'_{t+1}\rangle - |\phi_{t+1}\rangle\| \leq \pi/2^\ell.$$

3

Hence, setting $\ell = \lceil \log_2(\pi n/\epsilon) \rceil$, the algorithm outputs a state $|\phi'\rangle \stackrel{\text{def}}{=} |\phi'_{n+1}\rangle$ that satisfies

$$\| |\phi'\rangle - |\phi\rangle \| \leq \sum_{t=1}^{n} \| |\phi'_t\rangle - |\phi_t\rangle \| \leq \epsilon.$$

The total number of basis-changing gates used is $n \cdot \ell = O(n \log \frac{n}{\epsilon})$. □

We now consider the the general case.

Theorem 2.2. *Let $|\phi\rangle$ be a quantum state over n qubits and $\epsilon > 0$ be a constant. Then there exists a quantum algorithm that uses $O(\frac{1}{\epsilon} H(\phi) \log \frac{H(\phi)}{\epsilon}) = O(H(\phi) \log H(\phi))$ number of basis-changing gates, and maps $|0\rangle^{\otimes n}$ to a state $|\phi'\rangle$ such that*

$$\| |\phi'\rangle - |\phi\rangle \| \leq \epsilon. \tag{1}$$

Proof. Suppose for some $N > 0$ and $\alpha_i \in [0, 2\pi)$, $p_i \geq 0$, $0 \leq i \leq N - 1$,

$$|\phi\rangle = \sum_{i=0}^{N-1} e^{i\alpha_i} \sqrt{p_i} |i\rangle,$$

where $\sum_{i=0}^{N-1} p_i = 1$.

We first observe that the basis-preserving gate

$$G_\phi \stackrel{\text{def}}{=} \sum_i e^{i\alpha_i} |i\rangle\langle i|$$

maps

$$|\phi\rangle \rightarrow \sum_{i=0..N-1} \sqrt{p_i} |i\rangle,$$

and vice verser. Therefore we can assume that $\alpha_i = 0$, for all i.

For a real $\lambda > 1$ to be determined later, define

$$W_\lambda \stackrel{\text{def}}{=} \{i : p_i \geq 2^{-\lambda H(\phi)}\}, \qquad p \stackrel{\text{def}}{=} \sum_{i \notin W_\lambda} p_i, \qquad \text{and,}$$

$$|\phi_\lambda\rangle \stackrel{\text{def}}{=} \sum_{i \in W_\lambda} \sqrt{\frac{p_i}{1-p}} |i\rangle.$$

Then we have $p \leq \frac{1}{\lambda}$. Hence, $|W_\lambda| \leq 2^{\lambda H(\phi)}$, and

$$\| |\phi\rangle - |\phi_\lambda\rangle \| \leq p \leq \frac{1}{\lambda}.$$

Now set $k \stackrel{\text{def}}{=} \lambda H(\phi)$. After an appropriate permutation σ on $\{0,1\}^k$, $|\phi_\lambda\rangle$ can be written as

$$|\psi_\lambda\rangle \stackrel{\text{def}}{=} \sigma |\phi_\lambda\rangle = \sum_{x \in \{0,1\}^k} \sqrt{q_x} |x\rangle.$$

By Lemma 2.1, we can generate a state $|\psi'_\lambda\rangle$ using $O(k \log(\lambda k))$ basis-changing gates and

$$\| |\psi'_\lambda\rangle - |\psi_\lambda\rangle \| \leq \frac{1}{\lambda}.$$

4

The output state is $|\phi'_\lambda\rangle \overset{\text{def}}{=} \sigma^{-1}|\psi'_\lambda\rangle$, which satisfies

$$\||\phi'_\lambda\rangle - |\phi\rangle\| \le \||\phi'_\lambda\rangle - |\phi_\lambda\rangle\| + \||\phi_\lambda\rangle - |\phi\rangle\| \le \||\psi'_\lambda\rangle - |\psi_\lambda\rangle\| + 1/\lambda \le 2/\lambda.$$

Setting $\lambda = 2/\epsilon$, this gives the required precision. The total number of basis-changing gates used is

$$O(k \log(\lambda k)) = O(\frac{1}{\epsilon} H(\phi) \log \frac{H(\phi)}{\epsilon}) = O(H(\phi) \log H(\phi)).$$

$\square$

Remark 2.3 (Improving the upper bound). Since for some quantum states $|\phi\rangle$, a small perturbation may reduce $H(\phi)$ dramatically, the upper bound in Theorem 2.2 may be improved by approximating such a lower entropy approximation state. For example, consider

$$|\phi_\delta\rangle = (1 - \delta)|0\rangle + \sum_{i \in [K]} \frac{\sqrt{2\delta - \delta^2}}{\sqrt{K}} |i\rangle.$$

Then $H_{\phi_\delta} = \Theta(\log K)$. On the other hand, if $\epsilon \ge 2\delta$, the constant state $|0\rangle$ is an ϵ-approximation of $|\phi_\delta\rangle$. Hence no basis-changing gate is needed at all.

2.2   Lower bound.

We can generalize the definition of $H(\phi)$ to $H(\rho)$ for a mixed state ρ in the obvious way. Denote the trace norm of a matrix M by $\|M\|_{tr}$. Given a state $|\phi\rangle$ and a real $\epsilon > 0$ let

$$H_\epsilon(\phi) \overset{\text{def}}{=} \inf \{H(\rho) : \|\rho - |\phi\rangle\langle\phi|\|_{tr} \le \epsilon\}.$$

Note that $H_\epsilon(\phi)$ could be substantially smaller than $H(\phi)$, as demonstrated by the example in Remark 2.3. On the other hand, for some family of states, such as the uniform superpositions $\{\frac{1}{\sqrt{N}} \sum_{i=0..N-1} |i\rangle : N > 0\}$, $H_\epsilon(\phi) = \Theta(H(\phi))$, for small ϵ.

Theorem 2.4. *Let $|\phi\rangle$ be a quantum state and $\epsilon > 0$ be a constant. Then any quantum algorithm that generates a mixed state ρ that satisfies $\|\rho - |\phi\rangle\langle\phi|\|_{tr} \le \epsilon$ must use $\Omega(H_\epsilon(\phi))$ number of basis-changing gates.*

Notice that if $\||\phi\rangle - |\phi'\rangle\| \le \epsilon$, then

$$\||\phi\rangle\langle\phi| - |\phi'\rangle\langle\phi'|\|_{tr} \le 2\epsilon.$$

Therefore, in general, the algorithm in Theorem 2.2 is almost tight (up to a logarithmic factor) for sufficiently small ϵ and family of states that have $H_\epsilon(\phi) = \Theta(H(\phi))$.

Proof of Theorem 2.4.   Suppose k number of basis-changing gates are used to generate ρ. We will prove that $k = \Omega(H_\epsilon(\phi))$.

Denote the state after the ith basis-changing gate by $|\phi_i\rangle$, $0 \le i \le k$. Note that $\rho = F(|\phi_k\rangle\langle\phi_k|)$, for some physically realizable operator F, which is a composition of a permutation (with some phase) of the computational basis followed a partial trace. Hence

$$H(\phi_k) \ge H(\rho) \ge H_\epsilon(\phi). \tag{2}$$

Since $H_0 = 0$, it suffices to prove that $H_{i+1} \le H_i + C$, for all $0 \le i \le k - 1$, and some constant C.

Fix a t, $0 \le t \le k - 1$. Let U_t be the tth basis-changing gate, which is applied to a set of qubits A. The other qubits are denoted by B. Note that the number of qubits in A, denoted by C, is a constant. Denote by A' and B' two new systems that have the same number of qubits as in A and B, respectively. Let $Copy[A; A']$

be the product of Controlled-Not gates that use qubits in A as the control and the corresponding qubits in A' as the destination. Similarly define $Copy[B; B]$. Let

$$|\psi_t\rangle \overset{\text{def}}{=} (Copy[A; A] \cdot Copy[B; B]) \; |\phi_t\rangle_{AB} \otimes |00\cdots0\rangle_{A'B'}.$$

Define $|\psi_{t+1}\rangle$ similarly. Denote the von Neumann entropy of a mixed state by $E(\cdot)$. Then

$$H_t = E\left((|\psi_t\rangle\langle\psi_t|)_{AB}\right) = E\left((|\psi_t\rangle\langle\psi_t|)_{A'B'}\right).$$

The second equality follows from that $|\psi_t\rangle$ is a pure state. Similarly,

$$H_{t+1} = E\left((|\psi_{t+1}\rangle\langle\psi_{t+1}|)_{AB}\right) = E\left((|\psi_{t+1}\rangle\langle\psi_{t+1}|)_{A'B'}\right).$$

By the subadditivity of von Neumann entropy,

$$E((|\psi_{t+1}\rangle\langle\psi_{t+1}|)_{A'B'}) \leq E((|\psi_{t+1}\rangle\langle\psi_{t+1}|)_{A'}) + E((|\psi_{t+1}\rangle\langle\psi_{t+1}|_{B'}).$$

Since

$$|\psi_t\rangle \; = \; \left(Copy[A; A'] \cdot U_t^\dagger \cdot Copy[A; A']\right) \; |\psi_{t+1}\rangle,$$

we have

$$E((|\psi_{t+1}\rangle\langle\psi_{t+1}|)_{B'}) = E((|\psi_t\rangle\langle\psi_t|)_{B'}).$$

The latter is exactly

$$E((|\psi_t\rangle\langle\psi_t|)_{ABA'} \leq E((|\psi_t\rangle\langle\psi_t|)_{AB}) + E((|\psi_t\rangle\langle\psi_t|)_{A'}),$$

by the subadditivity again. Putting the above together, we have

$$H_{k+1} \leq H_k + E((|\psi_t\rangle\langle\psi_t|)_A) + E((|\psi_{t+1}\rangle\langle\psi_{t+1}|)_{A'}) \leq H_k + 2C.$$

Together with (2), this implies $k = \Omega(H_\epsilon(\phi))$. □

3 Quantum and Classical Tradeoffs in solving Grover's Problem.

In this section, we prove a quantum and classical tradeoff relation for Grover's Problem [7], which is also called Unstructured Search Problem. We start with the framework in which Grover's Problem is formulated and then present the main result.

3.1 Grover's Problem.

The input to Grover's Problem (or, the Unstructured Search Problem) of size n is a binary string $x = x_0 x_2 \cdots x_{n-1}$, where $x_i \in \{0, 1\}$, $0 \leq i \leq n - 1$, with the promise that there exists one and only one index i such that $x_i = 1$. The task is to identify i. The complicacy is that x is known *only* to an oracle, which can only be accessed by applying the oracle gate O_x:

$$O_x|i, b\rangle = |i, b \oplus x_i\rangle, \quad 0 \leq i \leq n - 1, \; b \in \{0, 1\}.$$

Hence, in general, an algorithm would start with a constant vector $|\phi_0\rangle$ in its state space, apply a sequence of unitary transformations $U_0, O_x, U_1, O_x, \cdots, O_x, U_T$, which is followed by a measurement that would

output i with a high probability (say $\geq 2/3$). The complexity of the algorithm is T, the number of applications of O_x.

In one of the most important papers in quantum computing, Grover [7] discovered a surprising quantum algorithm that makes only $O(\sqrt{n})$ queries, a quadratic speedup over the best possible classical algorithm. Because Grover's Problem is formulated in such a general way, Grover's Algorithm can be used in solving many other problems with a quantum speedup. A recent example is Ambainis' quantum algorithm for the classical problem of Element Distinctness [3]. In fact, Grover's Problem is an example of problems formulated in the so-called "black-box model", which has been widely studied by many authors (see, e.g., the survey of Ambainis [2]).

3.2   Quantum and classical tradeoffs for Grover's Problem.

Much work has been done on proving lower bounds in the quantum black-box model (see, e.g., two representative papers by Beals, Buhrman, Cleve, Mosca, and de Wolf [4], and by Ambainis [1]). In fact, the tight lower bound for Grover's Problem was known before Grover's work due to Bennett, Bernstein, Brassard, and Vazirani [5], and was refined by Boyer, Brassard, Høyer, and Tapp [6], and by Zalka [12].

Theorem 3.1 ([5]). *Any quantum algorithm for solving Grover's Problem of size n must query $\Omega(\sqrt{n})$ times.*

A quantum black-box algorithm can viewed as a sequence of blocks of classical reversible computation that may include oracle queries and are separated by layers of basis-changing gates. For example, for some $T = \Theta(\sqrt{n})$, Grover's Algorithm uses $2T + O(1)$ Fourier transforms, and in between, T oracle queries together with other classical reversible computation. We are interested in the tradeoff of the number of basis-changing layers and the number of queries.

Theorem 3.2. *Any quantum algorithm solving Grover's problem of size n using T queries and ℓ Fourier transforms must satisfy $T \cdot \ell = \Omega(n)$.*

A special case where the algorithm is required to make s queries non-adaptively, for a fixed s, before making a local computation was studied by Zalka [12], which implies the same lower bound as the above for this case.

It is not hard to see that this tradeoff relation is optimal as long as $T = \Omega(\sqrt{n})$:

Proposition 3.3. *For any $T \geq \sqrt{n}$, there exists a quantum algorithm that solves Grover's Problem of size n using $\Theta(T)$ queries and $\Theta(n/T)$ layers of basis-changing gates.*

3.3   Proofs.

We shall prove Theorem 3.2 by a generalized form of the "quantum adversary" technique of Ambainis [1], which we now briefly review.

Let f be a function defined on two disjoint sets X and Y, where X, $Y \subseteq \{0, 1\}^n$, and for any pair $x \in X$, and $y \in Y$, $f(x) \neq f(y)$. Let $R \in X \times Y$, and

$$ m \stackrel{\text{def}}{=} \min_{x \in X} |\{y : (x, y) \in R\}|, \qquad \ell \stackrel{\text{def}}{=} \max_{x \in X, i \in [n]} |\{y : (x, y) \in R \text{ and } x_i \neq y_i\}|, $$

and m' and ℓ' are defined similarly with X (x) and Y (y) switched. Then

Lemma 3.4 ([1]). *Any quantum algorithm that computes f with error probability $\leq \epsilon$, $0 \leq \epsilon < 1/2$, must make $\Omega(\sqrt{mm'/\ell\ell'})$ queries.*

This can be proved by considering the changes on a "progress indicator" after each query of the algorithm. Specifically, suppose we fix an algorithm that makes T queries. Let $|\phi_z^t\rangle$ be the state with oracle z and after the tth oracle query. Define the progress indicator

$$p_t \overset{\text{def}}{=} E_{x \in X, y \in Y} [\ \langle \phi_x^t | \phi_y^t \rangle\ ], \qquad t = 0, \cdots, T.$$

Notice that only the oracle gate may change the progress indicator. Clearly $p_0 = 1$. Furthermore, since the algorithm succeeds with a probability at least $1 - \epsilon > 1/2$,

Proposition 3.5 ([1]). *For some constant c, $0 \le c < 1$, $p_T \le c$.*

The lower bound is then established by proving

$$|p_t - p_{t-1}| = O(\sqrt{\ell\ell'/mm'}), \qquad \forall t \in [T].$$

In our context, we shall consider the change on the progress indicator p_t after a sequence of classical reversible computation with oracle queries.

Lemma 3.6. *Let f, X, Y, and p_t be as described above. Let $k \in [n]$, and*

$$\alpha_k \overset{\text{def}}{=} \max_{x \in X, s \subseteq [n], |s|=k} \frac{|\{y : (x,y) \in R,\ y\ \text{differs from } x\ \text{when restricted to } s\}|}{|\{y : (x,y) \in R\}|}.$$

Similarly define β_k with x switched with y and X switched with Y. Then for any t, after a sequence of classical reversible computation that uses k queries,

$$|p_t - p_{t+k}| = O(\sqrt{\alpha_k \cdot \beta_k}).$$

Proof. Denote the computational basis by $\mathcal{C}$. Denote the starting state (before the sequence of classical reversible computation) with oracle z by

$$|\phi_z\rangle = \sum_{c \in \mathcal{C}} \gamma_{z,c} |c\rangle.$$

For an input $z \in \{0,1\}^n$, denote by σ_z the permutation on the computational basis specified by the algorithm. Then after the classical reversible computation, $|\phi_z\rangle \to \sigma_z |\phi_z\rangle$. Hence the change of the progress indicator

$$|p_t - p_{t+k}| = \Big|\ E_{x,y}\ [\langle \phi_x | \phi_y \rangle] - E_{x,y}\ [\langle \phi_x | \sigma_x^\dagger \sigma_y | \phi_y \rangle]\ \Big|$$

is upper bounded by

$$E[|\langle \phi_x | \sigma_x^\dagger \sigma_y - I | \phi_y \rangle|] \tag{3}$$

$$\le\ \frac{1}{|R|} \sum_{(x,y) \in R} \sum_{c,c'} |\gamma_{x,c}| \cdot |\gamma_{y,c'}| \cdot |\langle c | \sigma_x^\dagger \sigma_y - I | c' \rangle| \tag{4}$$

$$\le\ \frac{1}{|R|} \sum_{\substack{(x,y) \in R,\ c,\ c' \\ c' \ne c,\ \sigma_y(c')=\sigma_x(c)}} |\gamma_{x,c}| \cdot |\gamma_{y,c'}| \tag{5}$$

$$+ \frac{1}{|R|} \sum_{\substack{(x,y) \in R,\ c \\ \sigma_x(c) \ne \sigma_y(c)}} |\gamma_{x,c}| \cdot |\gamma_{y,c}|. \tag{6}$$

Let us bound the second summation (6) first. After applying Cauchy-Schwartz, we have the upper bound

$$\frac{1}{|R|} \sqrt{\sum_{x,c} \sum_{\substack{y:(x,y) \in R \\ \sigma_x(c) \ne \sigma_y(c)}} |\gamma_{x,c}|^2} \cdot \sqrt{\sum_{y,c} \sum_{\substack{x:(x,y) \in R \\ \sigma_x(c) \ne \sigma_y(c)}} |\gamma_{y,c}|^2}. \tag{7}$$

8

A fixed combination of x and c determines a set of k coordinates being queried. If y is identical to x in these coordinates then $\sigma_y(c) = \sigma_x(c)$. Therefore, with

$$m_x = \max_{s \subseteq [n], |s|=k} |\{y : (x,y) \in R \text{ and } y \text{ differs from } x \text{ when restricted to } s\}|,$$

and m_y similarly defined, the above equation is further upper bounded by

$$\frac{1}{|R|} \sqrt{\sum_x m_x \cdot \sum_y m_y} \le \sqrt{\alpha_k \beta_k},$$

since $\frac{\sum_x m_x}{|R|} \le \alpha_k$ and $\frac{\sum_y m_y}{|R|} \le \beta_k$.

Now we bound the first summation (5). By Cauchy-Schwartz, it is upper-bounded by

$$\frac{1}{|R|} \sqrt{\sum_{x,c} \sum_{\substack{c', \ y:(x,y)\in R \\ c' \neq c, \ \sigma_x(c)=\sigma_y(c')}} |\gamma_{x,c}|^2} \cdot \sqrt{\sum_{y,c'} \sum_{\substack{c, \ x:(x,y)\in R \\ c\neq c', \ \sigma_x(c)=\sigma_y(c')}} |\gamma_{y,c'}|^2}.$$

The constraints on y and c' in the first summation are equivalent to that $\sigma_x(c) \neq \sigma_y(c)$ and $c' = \sigma_y^\dagger \sigma_x(c)$, therefore the above is upper bounded by Equation 7, hence by $\sqrt{\alpha_k \beta_k}$ as well. Therefore, the change on the progress indicator is at most $2\sqrt{\alpha_k \beta_k}$. □

Remark 3.7. A lower bound better than $2\sqrt{\alpha_k \beta_k}$ is $\frac{2}{|R|}\sqrt{(\sum_x m_x) \cdot (\sum_y m_y)}$, where m_x and m_y are defined in the above proof. However, for our purpose of proving Theorem 3.2, both bounds are the same.

We are now ready to prove Theorem 3.2

Proof of Theorem 3.2. For the purpose of proving lower bound, it suffices to consider the following decision version of Grover's problem: determine whether or not the oracle is $e_0 = 0^n$ or e_i, the n bit binary string that has the single 1 at the i-th position, for some $i \in [n]$. Let f in Lemma 3.6 be this decision problem and set

$$X \stackrel{\text{def}}{=} \{ e_0 \}, \qquad Y \stackrel{\text{def}}{=} \{ e_i : 1 \le i \le n \}, \qquad \text{and,} \qquad R \stackrel{\text{def}}{=} X \times Y.$$

Fix an algorithm that makes T queries and ℓ levels of basis-changing gates. Then the algorithm can be divided into $\ell + 1$ blocks of classical reversible computation with the ℓ basis-changing layers separating them. Number the blocks by $1, 2, \cdots, \ell + 1$. For each block s, let k_s be the number of queries in this block, and p_{s-1} be the progress indicator at the beginning of the block. The progress indicator at the end of the last block is denoted by $p_{\ell+1}$. We have $\sum_{s=1}^{\ell+1} k_s = T$, $p_0 = 1$, and $p_{\ell+1} \le c$ for some constant c with $0 \le c < 1$.

Furthermore, for each k_s,

$$\alpha_{k_s} = \frac{k_s}{n}, \qquad \text{and,} \qquad \beta_{k_s} = 1.$$

Then, by Lemma 3.6,

$$|p_s - p_{s-1}| = O(\sqrt{k_s/n}), \qquad \forall s \in [\ell+1].$$

Hence

$$\sum_{s \in [\ell+1]} \sqrt{k_s/n} = \Omega(1).$$

By the Cauchy-Swartz Inequality, the left hand side is upper-bounded by

$$\sqrt{(\ell+1) \cdot (\sum_{s=1..\ell+1} k_s)/n} = \Theta(\sqrt{\ell \cdot T/n}).$$

Hence $\ell \cdot T = \Omega(n)$. □

Proposition 3.3 can be proved by using a mixture of classical exhaustive algorithm and Grover's algorithm.

Proof of Proposition 3.3. Consider the following algorithm. Divide the n bits binary string into $h \stackrel{\text{def}}{=} \lceil (n/t)^2 \rceil$ blocks. Apply Grover's algorithm to search for a block that contains the 1, and within each block, query all the bits. The total number of queries is $\Theta(\sqrt{h} \cdot \frac{n}{h}) = \Theta(T)$, and the total number of layers of basis-changing gates is $\Theta(\sqrt{h}) = \Theta(n/T)$. $\square$

4 Discussion.

We initiate the study of what we called "quantum and classical tradeoffs", which in essence is the relation of the number of basis-changing gates in a quantum circuit with the computation power of the quantum circuit. Specifically, we prove lower and upper bounds on the number of basis-changing gates for generating a given quantum state, and prove an optimal tradeoff relation between the number of a layers of basis-changing gates and the number of queries for algorithms that solve Grover's Problem. We shall conclude this paper by formulating a class of open problems in this direction.

Since Toffoli and Hadamard are universal for quantum computing (see, e.g., Shi [11]), we can assume that any quantum circuit involves only these two gates. Notice that the composition of a set of Hadamard gates is just a Fourier transform over a tensor product of $\mathbb{Z}_2$.

For each integer $k \geq 0$, define the complexity class FH_k (FH meant to stand for "Fourier Hierarchy) to be languages that can be decided with a bounded error probability by a quantum circuit of polynomial size and $\leq k$ Fourier transforms. Notice that if only uniform families of quantum circuits are considered, $FH_0 = P$, and $FH_1 = BPP$. When $k = 2$, FH_2 starts to have nontrivial quantum computation power. For example, the oracle version of FH_2 includes Simon's problem, and Factoring can be done in FH_2 via Kitaev's Phase Estimation Algorithm.

It appears a reasonable conjecture that in general, the number of Fourier transforms can not be reduced without substantial increase of the circuit size.

Conjecture 4.1. *For any $k \geq 0$, $FH_k \subsetneq FH_{k+1}$.*

Since we do not know how to prove strong lower bounds in a general model, one may have to consider first oracle versions of the problem, that is, show an exponential separation between FH_k and FH_{k+1} relative to an oracle for any k. Simon's Problem provides an oracle separation for FH_1 and FH_2. The iterated version of it, as well as the Recursive Fourier Sampling problem in [5] appear to be good candidates for an oracle separation for a general k.

5 Acknowledgment

References

[1] Andris Ambainis. Quantum lower bounds by quantum arguments. In *Proceedings of the Thirty-second Annual ACM Symposium on the Theory of Computing*, pages 636–643, Portland, Oregon, May 2000.

[2] Andris Ambainis. Quantum query algorithms and lower bounds. In *Proceedings of Foundations of the Formal Sciences III*, September 2001.

[3] Andris Ambainis. Quantum walk algorithm for element distinctness. Pre-print: quant-ph/0311001, 2003.

[4] Robert Beals, Harry Buhrman, Richard Cleve, Michele Mosca, and Ronald de Wolf. Quantum lower bounds by polynomials. *Journal of the ACM*, 48(4):778–797, July 2001.

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

[6] Michel Boyer, Gilles Brassard, Peter Hyer, and Alain Tapp. Tight bounds on quantum searching. Technical Report PP-1996-11, Department of Mathematics and Computer Science, Odense University, May 30 1996. Mon, 7 Oct 1996 11:36:47 GMT.

[7] Lov K. Grover. A fast quantum mechanical algorithm for database search. In *Proceedings of the Twenty-Eighth Annual ACM Symposium on the Theory of Computing*, pages 212–219, Philadelphia, Pennsylvania, May 1996.

[8] Richard Jozsa and Noah Linden. On the role of entanglement in quantum computational speed-up. Pre-print: quant-ph/0201143, 2002.

[9] A. Yu. Kitaev, A. H. Shen, and M. N. Vyalyi. *Classical and quantum computation*, volume 47 of *Graduate Studies in Mathematics*. American Mathematical Society, Providence, RI, 2002. Translated from the 1999 Russian original by Lester J. Senechal.

[10] Knuth and Yao. The complexity of nonuniform random number generation. In J. F. Traub, editor, *Algorithms and Complexity: New Directions and Recent Results*, pages 357–428. Academic Press, 1976.

[11] Yaoyun Shi. Both Toffoli and Hadamard need little help to do universal quantum computation. *Quantum Information and Computation*, 3(1):84–92, 2003.

[12] Christof Zalka. Grover's quantum searching algorithm is optimal. *Physical Revew A*, 60:2746–2751, October 1999.