Quantum algorithms (CO 781/CS 867/QIC 823) Winter 2011

**ASSIGNMENT 3** due Tuesday 29 March (in class)

**Problem 1 (*The triangle problem*).**

In the *triangle problem*, you are asked to decide whether an $n$-vertex graph $G$ contains a triangle (a complete subgraph on 3 vertices). The graph is specified by a black box that, for any pair of vertices of $G$, returns a bit indicating whether those vertices are connected by an edge in $G$.

a. What is the classical query complexity of the triangle problem?

b. Say that an edge of $G$ is a *triangle edge* if it is part of a triangle in $G$. What is the quantum query complexity of deciding whether a particular edge of $G$ is a triangle edge?

c. Now suppose you know the vertices and edges of some $m$-vertex subgraph of $G$. Explain how you can decide whether this subgraph contains a triangle edge using $O(m^{2/3}\sqrt{n})$ quantum queries.

d. Consider a quantum walk algorithm for the triangle problem (or, equivalently, deciding whether a graph contains a triangle edge). The walk takes place on a graph $\mathcal{G}$ whose vertices correspond to subgraphs of $G$ on $m$ vertices, and whose edges correspond to subgraphs that differ by changing one vertex. A vertex of $\mathcal{G}$ is marked if it contains a triangle edge. How many queries does this algorithm use to decide whether $G$ contains a triangle? (Hint: Be sure to account for the queries used to initialize the walk, the queries used to move between neighboring vertices of $\mathcal{G}$, and the queries used to check whether a given vertex of $\mathcal{G}$ is marked. To get a nontrivial result, you should use the search framework mentioned in class that takes many steps according to the walk on $\mathcal{G}$ with no marked vertices before performing a phase flip at marked vertices.)

e. Choose a value of $m$ that minimizes the number of queries used by the algorithm. What is the resulting upper bound on the quantum query complexity of the triangle problem?

f. *Challenge problem:* Generalize this algorithm to decide whether $G$ contains a $k$-clique. How many queries does the algorithm use?

**Problem 2 (*Unstructured search by formula evaluation*).**

Grover's algorithm computes the OR of $n$ bits using $O(\sqrt{n})$ quantum queries to those bits. In this problem you will give an alternative algorithm for computing OR by evaluating a NAND formula.

Since $\text{OR}(x_1, \ldots, x_n) = \text{NAND}(\bar{x}_1, \ldots, \bar{x}_n)$, we can represent the OR formula by a NAND tree in which the root has $n$ children, and each of those children has one child, which is a leaf. Given an input $x_1, \ldots, x_n$, we modify the tree by deleting every leaf in the original tree corresponding to an index $i$ for which $x_i = 1$.

We will start our quantum algorithm from the root, so you can restrict your attention to the subspace $\mathcal{S} := \text{span}\{H^j|\text{root}\rangle : j = 0, 1, 2, \ldots\}$, where $H$ is a weighted adjacency matrix of the tree (with weights to be determined).

a. First consider the input $x_1 = \cdots = x_n = 0$, for which the formula evaluates to 0. Define the weighted adjacency matrix $H$ of the corresponding tree by assigning a weight of $\alpha$ to the edges connected to the root and a weight of 1 to the remaining edges. Compute the spectrum (both eigenvalues and eigenvectors) of $H$ within the subspace $\mathcal{S}$.

b. For what values of $\alpha$ does $H$ (as defined in part a) have an eigenstate of eigenvalue 0 with overlap $\Omega(1)$ on the root?

c. Now consider an input with $x_i = 1$ for precisely one index $i$. Compute the spectrum of $H$ within the subspace $\mathcal{S}$.

d. For what values of $\alpha$ does $H$ (as defined in part c) have a minimum eigenvalue of $\Omega(1/\sqrt{n})$ (in absolute value)? Choose a value of $\alpha$ so that this condition and the one from part b are satisfied simultaneously.

e. Compute the spectrum of $H$ for an arbitrary input, and show that the minimum eigenvalue of $H$ (again in absolute value) can only be larger than in part c if there is more than one index $i$ for which $x_i = 1$.

f. *Challenge problem:* Describe a simulation of the continuous-time quantum walk generated by $H$ that computes OR using $O(\sqrt{n})$ queries. (Notice that the root of the tree has high degree, so you cannot use results on the simulation of sparse Hamiltonians.)

**Problem 3 (*Original formulation of the adversary method*).**

For a Boolean function $f\colon \{0,1\}^n \to S$, the adversary method says that $Q_\epsilon(f) \geq \frac{1-2\sqrt{\epsilon(1-\epsilon)}}{2}\,\mathrm{Adv}(f)$, where $\mathrm{Adv}(f) := \max_\Gamma \frac{\|\Gamma\|}{\|\Gamma_i\|}$, with the maximization is over all adversary matrices $\Gamma$.

Ambainis originally formulated the adversary method differently, as follows. Let $X, Y \subset \{0,1\}^n$ such that $f(x) \neq f(y)$ for all $x \in X, y \in Y$. For any relation $R \subset X \times Y$, define

$$m := \min_{x\in X} |\{y \in Y\colon (x,y) \in R| \qquad \ell := \max_{\substack{x\in X\\ i\in\{1,\ldots,n\}}} |\{y \in Y\colon (x,y) \in R \text{ and } x_i \neq y_i\}|$$

$$m' := \min_{y\in Y} |\{x \in X\colon (x,y) \in R| \qquad \ell' := \max_{\substack{y\in Y\\ i\in\{1,\ldots,n\}}} |\{x \in X\colon (x,y) \in R \text{ and } x_i \neq y_i\}|.$$

Then define $\mathrm{Amb}(f) := \max_{X,Y,R} \sqrt{\frac{mm'}{\ell\ell'}}$.

Prove that $\mathrm{Adv}(f) \geq \mathrm{Amb}(f)$, and hence that $Q_\epsilon(f) \geq \frac{1-2\sqrt{\epsilon(1-\epsilon)}}{2}\mathrm{Amb}(f)$.

**Problem 4 (*Applying the adversary method*).**

Use the adversary method to prove the following lower bounds. (You should apply the adversary method directly to the given function rather than giving a reduction from some other problem.)

a. (*Parity*) Define $\textsc{parity}\colon \{0,1\}^n \to \{0,1\}$ by $\textsc{parity}(x) = x_1 \oplus \cdots \oplus x_n$. Show that $Q(\textsc{parity}) = \Omega(n)$.

b. (*Two-level* NAND *tree*) Define $\textsc{nand}^2\colon \{0,1\}^{n^2} \to \{0,1\}$ by

$$\textsc{nand}^2(x) = \textsc{nand}(\textsc{nand}(x_1,\ldots,x_n), \textsc{nand}(x_{n+1},\ldots,x_{2n}),\ldots,\textsc{nand}(x_{n^2-n+1},\ldots,x_{n^2})).$$

Show that $Q(\textsc{nand}^2) = \Omega(n)$.

c. (*Graph connectivity*) With $x \in \{0,1\}^{\binom{n}{2}}$ specifying an $n$-vertex graph as in Problem 1, define $\textsc{con}\colon \{0,1\}^{\binom{n}{2}} \to \{0,1\}$ by

$$\textsc{con}(x) = \begin{cases} 1 & \text{if the graph described by } x \text{ is connected} \\ 0 & \text{otherwise.} \end{cases}$$

Show that $Q(\textsc{con}) = \Omega(n^{3/2})$.

**Problem 5 (*A limitation on quantum speedup for total functions*).**

In this problem, you will show that quantum computers can obtain at most a polynomial speedup for the query complexity of total functions.

a. Given a Boolean function $f\colon \{0,1\}^n \to \{0,1\}$, a *certificate* for $f$ on input $x \in \{0,1\}^n$ is a subset of the bits of $x$ such that the value of $f(x)$ is determined by those bits alone. Let $C_x(f)$ denote the size of the smallest certificate for $f$ on input $x$, and let $C(f) := \max_{x\in\{0,1\}^n} C_x(f)$ (this is called the *certificate complexity* of $f$). What is $C(\textsc{or})$?

b. Consider the following algorithm for computing $f(x)$:

   Let $c \leftarrow \emptyset$
   While $c$ does not certify that $f(x) = 0$
       Choose $x' \in \{0,1\}^n$ such that $f(x') = 1$ and $x_i = x'_i$ for all $i \in c$
       Let $c'$ be a minimal certificate for $x'$
       Query $x_i$ for $i \in c'$
       Let $c \leftarrow c \cup c'$
       If $c$ certifies that $f(x) = 1$ then return "1"
   End while
   Return "0"

   Show that this algorithm uses at most $C(f)^2$ queries.

c. For $x \in \{0,1\}^n$ and $S \subseteq \{1,\dots,n\}$, let $x^{(S)}$ denote $x$ with $x_i$ replaced by $\bar{x}_i$ for all $i \in S$. Call $S$ a *sensitive block* of $x$ if $f(x) \neq f(x^{(S)})$. Prove that if $S_1,\dots,S_k$ is a maximal set of disjoint sensitive blocks of $x$ (i.e., there is no other sensitive block that is disjoint from all of $S_1,\dots,S_k$), then $S_1 \cup \cdots \cup S_k$ is a certificate for $f(x)$.

d. Let $\mathrm{bs}_x(f)$ denote the largest possible number of disjoint sensitive blocks of $x$, and let $\mathrm{bs}(f) := \max_{x\in\{0,1\}^n} \mathrm{bs}_x(f)$ (this is called the *block sensitivity* of $f$). Call a sensitive block $S$ *minimal* if no subset of $S$ is sensitive. Show that if $S$ is a minimal sensitive block of some input, then $|S| \le \mathrm{bs}(f)$.

e. Prove that $C(f) \le \mathrm{bs}(f)^2$.

f. Let $x \in \{0,1\}^n$ have disjoint sensitive blocks $S_1,\dots,S_{\mathrm{bs}(f)}$. For any $y \in \{0,1\}^{\mathrm{bs}(f)}$, let $x^{[y]}$ denote $x$ with $x_i$ replaced by $\bar{x}_i$ if $i \in S_j$ and $y_j = 1$ for some $j \in \{1,\dots,\mathrm{bs}(f)\}$. Given a polynomial $p\colon \{0,1\}^n \to \mathbb{R}$, define a polynomial $p'\colon \{0,1\}^{\mathrm{bs}(f)} \to \mathbb{R}$ by $p'(y) := p(x^{[y]})$. Explain why $\deg(p') \le \deg(p)$.

g. Prove that $\widetilde{\deg}(f) = \Omega(\sqrt{\mathrm{bs}(f)})$. (*Hint:* Generalize the proof that $\widetilde{\deg}(\textsc{or}) = \Omega(\sqrt{n})$, using $p'$ in place of the original approximating polynomial $p$.)

h. Conclude that $Q(f) = \Omega(D(f)^{1/8})$, where $D(f)$ denotes the deterministic classical query complexity of $f$.