CS 6846: Quantum Computing 13th September, 2024

Homework 2

Instructor: Shweta Agrawal *Due: 23rd Sept, 5 PM*

Name	Roll No.

Instructions:

1. Please type up your solutions using latex.
2. Please email TA and instructor.
3. You may collaborate with other students. Please mention the names of your collaborators or any other source that you use for the solution.
4. Please type up your solutions *individually* without any help.

Problem 1: Superdense Coding (5 points)

Alice and Bob prepare an EPR pair, that is, two qubits in the state $\frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$. They each take one qubit home. Suddenly, Alice decides she wishes to convey one of 4 messages to Bob; in other words, she wants to convey a classical string $uv \in \{0,1\}^2$ to Bob. Alice does the following in the privacy of her own home: First, if $u = 1$, she applies a NOT gate to her qubit (else if $u = 0$ she does nothing here). Next, if $v = 1$, she applies a Z gate to her qubit (else if $v = 0$, she does nothing here). Finally, she walks to Bob's house and silently hands him her qubit. Show that by measuring in an appropriate basis, Bob can exactly determine Alice's message $uv \in \{0,1\}^2$.

Problem 2: Deferred Measurements (3 points)

For $0 \leq p \leq 1$ let COIN_p denote a gate that has no input and one output, the output being a random bit which is 1 with probability p and 0 with probability $1 - p$. The standard way to augment the basic circuit model with randomness is to allow the use of $\text{COIN}_{1/2}$ gates. In our definition of quantum circuits, we allowed quantum gates, plus measurement at the very end. We saw in class that CCNOT can simulate the AND, OR, and NOT gates. To simulate $\text{COIN}_{1/2}$ gates we suggested to pass a $|0\rangle$ qubit through a Hadamard gate and then measure it. However, if we want to use this random bit within our circuit, we need to augment the quantum circuit model by

allowing "intermediate measurements" (i.e., measuring some qubits prior to the end of the computation). While this is okay both theoretically and physically, it makes the model somewhat more complicated. Luckily, we can show that any computation done by a quantum circuit using intermediate measurements can be equivalently and nearly as efficiently done by a quantum circuit that only has a single measurement at the end. In this problem you won't quite prove this in full, but you'll get the essential idea. Precisely, suppose C is a randomized circuit with n input bits, a ancilla bits, r $\text{COIN}_{1/2}$ gates, s CCNOT gates, and m output bits (possibly including garbage). Describe a straightforward transformation to a quantum circuit C' with n input bits, $a + 2r$ ancilla bits, $s + 2r$ CCNOT/CNOT/Hadamard gates, and $m + r$ output bits, such that when the output bits are measured at the end of $C'(x)$, the probability distribution on the first m of them is exactly the same as the probability distribution on the output bits of $C(x)$. (Prove that your circuit indeed produces random bits).

Problem 3: Classical Comparison (3 points)

Give a randomized classical algorithm that makes only two queries to f, and decides the Deutsch-Jozsa problem with success probability at least 2/3 on every possible input.

Problem 4: Fun with Deutsch-Jozsa (4 points)

Let $N = 2^n$. Suppose our function $f\colon \{0,1\}^n \to \{0,1\}$ satisfies the following premise: either (1) the first $N/2$ bits of the truth table of f are all 0 and the second $N/2$ bits are all 1; or (2) the number of $1s$ in the first half of the truth table of f plus the number of $0s$ in the second half, equals $N/2$. Modify the Deutsch-Jozsa algorithm to efficiently distinguish these two cases (1) and (2).

Problem 5: Simon's Algorithm

Suppose we run Simon's algorithm for the following function $f\colon \{0,1\}^3 \to \{0,1\}^3$:
$f(000) = f(111) = 000$
$f(001) = f(110) = 001$
$f(010) = f(101) = 010$
$f(011) = f(100) = 011$

Note that f is 2-to-1 and $f(x) = f(x \oplus 111)$ for all $x \in \{0,1\}^3$, so $s = 111$.

(a) (1 point) Give the starting state of Simon's algorithm.

(b) (1 point) Give that state after the first Hadamard transforms on the first 3 qubits.

(c) (2 points) Give the state after applying the oracle.

(d) (1 points) Give the state after measuring the second register (suppose the measurement gave $|001\rangle$).

(e) (2 points) Using $H^{\otimes n}|x\rangle = \frac{1}{\sqrt{2^n}} \sum_{z\in\{0,1\}^n} (-1)^{\langle x,z\rangle} |z\rangle$, give the state after the final Hadamards.

(f) (1 points) Why does a measurement of the first 3 qubits of the final stage give information about s?

(g) (2 points) Suppose the first run of the algorithm gives $z = 011$ and a second run gives $z = 101$. Show that, assuming $s \neq 000$, those two runs of the algorithm already determine s.

Problem 6: Amplifying Success (4 points)

Let $f : \{0,1\}^n \to \{0,1\}$ be a function and C be a randomized (or quantum) circuit that computes the function f in the sense that for every input x, it holds that

$$\Pr\left(C(x) = f(x)\right) \geq \frac{2}{3}.$$

In class we discussed how by repeating this computation several times, we can make the probability of successful computation arbitrarily large. Formalize this. In more detail, design a circuit C' such that

$$\Pr\left(C'(x) = f(x)\right) \geq 1 - 2^{-n}.$$

Problem 7: Leveraging Parity (3 points).

Let $N = 2^n$. A *parity query* to input $x \in \{0,1\}^N$ corresponds to the $(N+1)$-qubit unitary map $Q_x \colon |y, b\rangle \to |y, b \oplus (x \cdot y)\rangle$ where $x \cdot y = \sum_{i=0}^{N-1} x_i y_i \mod 2$. For a fixed function $f \colon \{0,1\}^N \to \{0,1\}$, give a quantum algorithm that computes $f(x)$ using only one such query (i.e., one application of Q_x), and as many elementary gates as you want.

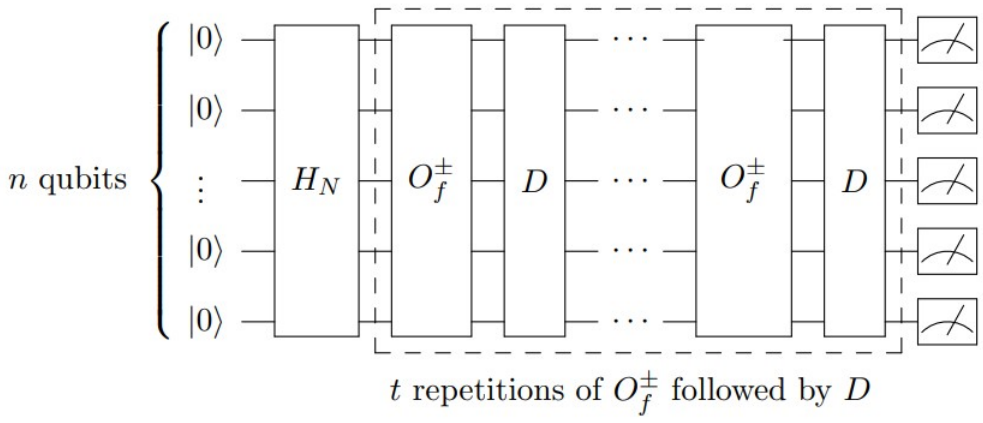
Problem 8: Grover search with multiple satisfying inputs

In this problem, we will generalize Grover's search to the case when f has more than one satisfying input.

So suppose we are given an oracle O_f for $f : \{0,1\}^n \to \{0,1\}$. Write $A = \{x : f(x) = 1\}$ and $k = |A|$. We will assume $k \geq 1$. Also write $B = \{x : f(x) = 0\}$, so $|B| = N - k$ where $N = 2^n$.

(a) (2 points) Explain how we can use the oracle to check if a given string $y \in \{0,1\}^n$ has $f(y) = 1$. Explain how we can use this to find an $x \in A$ with high probability in $O(1)$ queries whenever $k \geq N/2$. (We henceforth assume $k < N/2$.)

(b) (2 points) Recall Grover's algorithm, with t repetitions:



Letting $|\psi^{(t)}\rangle$ denotes the state of the circuit after t repetitions, show that we can write it as

$$|\psi^{(t)}\rangle = \alpha_t \frac{1}{\sqrt{k}} \sum_{x \in A} |x\rangle + \beta_t \frac{1}{\sqrt{N-k}} \sum_{x \in B} |x\rangle$$

where $\alpha_t, \beta_t \in \mathbb{R}$ satisfy $\alpha_t^2 + \beta_t^2 = 1$. What are α_0 and β_0?

(c) (2 points) Thinking of (β_t, α_t) as a point on the unit circle in $\mathbb{R}^2$, let us write θ_t for its angle from the horizontal axis (so that $\alpha_t = \sin\theta_t, \beta_t = \cos\theta_t$). Show that the transformation $(\beta_t, \alpha_t) \mapsto (\beta_{t+1}, \alpha_{t+1})$ is precisely rotation around the circle by an angle of $2\theta_0$.

(d) (2 points) Assume that the algorithm knows k. Briefly, why would the algorithm like to choose $t = \frac{1}{2}\left(\frac{\pi}{2\theta_0} - 1\right)$, if it could? Show that if it takes t to be the closest integer to this value, the circuit has the property that it outputs an element of A with a probability of at least $1/2$.

(e) (2 points) Again, assuming the algorithm knows k, show that it can find an element of A with high probability using $O(\sqrt{N/k})$ queries to the oracle. (You may want to use that $\sin\theta \leq \theta$ for all $\theta \geq 0$.)

Problem 9: Application of Grover's Algorithm (4 pts)

Let $N = 2^n$ and $x_0, \cdots, x_{N-1}$ be a sequence of distinct integers (you can think of them as the outputs in the truth table of some function F). We can query this function in the usual way, i.e., we can apply unitary $O\colon\ |i, 0\rangle \to |i, x_i\rangle$, as well as its inverse. The minimum of F is defined as $\min\{x_i | i \in \{0, \cdots, N-1\}\}$ Give a quantum

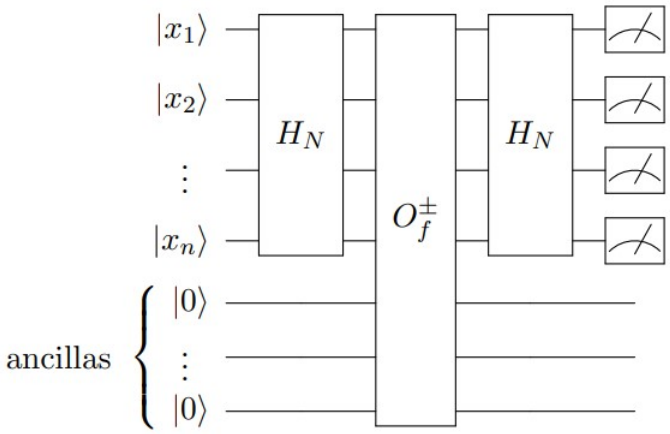
algorithm that finds (with probability $\geq 2/3$) an index achieving the minimum, using $O(\sqrt{N}\log N)$ queries.

Hint: start with $m = x_i$ for a random i, and repeatedly use Grover's algorithm to find an index j such that $x_j < m$ and update $m = x_j$. Continue this until you can find no element smaller than m, and analyze the number of queries of this algorithm. You are allowed to argue about this algorithm on a high level. Bonus: give a quantum algorithm that uses $O(\sqrt{N})$ queries.

Problem 10: The necessity of uncomputing (4 points).

Recall the convention that the oracle gate $O_f^{\pm}$ for a Boolean function $f : \{0,1\}^n \to \{0,1\}$ denotes the unitary transformation $|x\rangle \mapsto (-1)^{f(x)}|x\rangle$. When implementing $O_f^{\pm}$ in some applications, we might need additional ancilla/garbage bits, in which case $O_f^{\pm}$ actually denotes the unitary transformation $|x\rangle|0^m\rangle \mapsto (-1)^{f(x)}|x\rangle|g(x)\rangle$, where $g(x)$ is whichever m-bit garbage string produced on input x. In class, we have insisted that all oracle circuits uncompute their garbage, meaning that $g(x) = 0^m$ for all $x \in \{0,1\}^n$.

In this problem, we will justify why it is okay to "pretend" the ancilla bits don't exist. We will use as our example the Deutsch–Jozsa circuit, including ancilla/garbage bits; it is drawn as follows.



Suppose $O_f^{\pm}$ uncomputes its garbage, i.e. $g(x) = 0^m$ for all x. For a given Boolean function $f : \{0,1\}^n \to \{0,1\}$ (which need not be all-0's or balanced), compute the state of the system after the $O_f^{\pm}$ gate and after the second H_N gate. Show that each of these states can be written as $|\psi_1\rangle \otimes |\psi_2\rangle$, where $|\psi_1\rangle$ is the state of the first n qubits and $|\psi_2\rangle$ is the state of the last m qubits. Finally, show that the distribution on measurement outcomes is the same as it would have been if $O_f^{\pm}$ had no ancilla bits.