Homework 7

Introduction to Quantum Information Science
Due Monday, April 3rd at 11:59 PM

1. Universal gate sets. Identify the following gate sets as either universal or not universal in the sense of able to approximate any target unitary to any desired precision. If it is not, argue why.

a) {All single qubit gates, CNOT}

b) {Toffoli, Hadamard}

c) {Toffoli, Phase}

d) {Toffoli, Phase, Hadamard}

e) {Hadamard, Phase, CPHASE}

f) {Controlled Hadamard, Controlled Phase, NOT}

2. Quantum computation with real amplitudes.

'Real quantum mechanics' is a hypothetical theory that's identical to standard quantum mechanics, except that the amplitudes always need to be real – and instead of unitary matrices, we're restricted to applying real orthogonal matrices.

Prove that any standard quantum circuit acting on n qubits, can be perfectly simulated by a real quantum circuit acting on n+1 qubits – and moreover, by a circuit containing exactly as many gates as the original circuit (although the gates might act on slightly more qubits than the gates of the original circuit).

To illustrate, show how the Phase gate gets converted into a purely real gate in your simulation. Conclude that complex amplitudes are never actually needed for quantum computing speedups – positive and negative real amplitudes suffice. [Hint: Observe that, with n+1 qubits rather than n, you have 2^{n+1} amplitudes rather than just 2^n.]

3. Simon's problem. In Simon's problem, recall that we're given oracle access to a function $f : \{0,1\}^n \rightarrow \{0,1\}^n$, and are promised that there exists a secret string $s \neq 0^n$ such that $f(x) = f(y)$ if and only if $y = x \oplus s$ for all different $x, y \in \{0,1\}^n$ (where $\oplus$ denotes bitwise XOR).

a) Simon's algorithm proceeds by repeatedly preparing a superposition over all 2^n possible inputs to f, then querying f, and finally measuring all n qubits of the question register in the Hadamard basis. This yields a collection of output strings, $z_1, z_2, ...$ In class, we asserted without proof that, after $k = O(n)$ of these output strings, one has a collection of random mod-2 linear equations that one can solve, in classical polynomial time, to uniquely

recover s with overwhelming probability—meaning that the probability that the algorithm fails decreases exponentially with n. Rigorously prove that $O(n)$ repetitions of Simon's algorithm are enough, if we want to succeed with $1 - e^{-n}$ probability.

b) Suppose instead that there are two nonzero secret strings, $s \neq t$, such that $f(x) = f(x \oplus s) = f(x \oplus t) = f(x \oplus s \oplus t)$ for all x. Describe a variation of Simon's algorithm that finds the entire set $s, t, s \oplus t$ in time polynomial in n. When you measure a state in your algorithm, what are the possible results of the measurement? How do you use those measurement results to reconstruct the set $s, t, s \oplus t$?

4. Quantum Fourier Transform.

The Quantum Fourier Transform QFT_d is a quantum gate acting on *qudits*, i.e. quantum systems with d levels. It is defined below for $x, y \in 0, 1, ..., d-1$.

$$QFT_d |x\rangle = \frac{1}{\sqrt{d}} \sum_{y=0}^{d-1} e^{2\pi i x y/d} |y\rangle$$

a) Calculate QFT_2, QFT_3, and QFT_4 explicitly. By what other name is QFT_2 known?

b) Prove that QFT_d is unitary for all d.

c) For which values of d is QFT_d its own inverse?

d) Suppose $d = pq$ is a composite number, and suppose we feed QFT_d a uniform superposition over all the integer multiples of p, from 0 up to $d - p$. What state does this produce as output?

5. In the Bernstein-Vazirani problem, recall that we're given oracle access to a Boolean function $f : \{0,1\}^n \to \{0,1\}$. We're promised that there exists a "secret string," s in $\{0,1\}^n$, such that $f(x) = sx (\text{mod } 2)$ for all x. The problem is to recover x. The Bernstein-Vazirani algorithm solves this problem with just a single quantum query to f.

Now suppose that the oracle is noisy: that is, for some error parameter $\epsilon > 0$, the equation $f(x) = sx (\text{mod } 2)$ holds only for a $1 - \epsilon$ fraction of inputs x.

a) Calculate a lower bound on the probability that a single run of the Bernstein-Vazirani algorithm nevertheless succeeds in recovering s.

b) Explain what happens when $\epsilon = 1/2$. Is there a reason why, in some sense, the Bernstein-Vazirani algorithm can't possibly succeed at recovering s in that case?