Homework 10

Introduction to Quantum Information Science
Due Monday, April 24th at 11:59 PM

1. If you intend to attend the IBM quantum experience recitation session on Friday, April 28th at 3:00 PM in GDC 2.506, sign up on IBM's website here:
`https://quantumexperience.ng.bluemix.net/qstage/`
What email address did you use? (Note: it looks like the {eid}@eid.utexas.edu do not work for everybody.)

2. Recall the $N \times N$ Grover diffusion matrix D, whose diagonal entries are all $\frac{2}{N} - 1$ and whose off-diagonal entries are all $\frac{2}{N}$.

a) Prove that D is indeed a unitary matrix.

b) Let $N = 2^n$ be a power of 2. Prove that D can indeed by generated by Hadamarding n qubits, then applying the $N \times N$ diagonal matrix $A = \text{diag}(1, -1, -1, ..., -1)$, then Hadamarding n qubits a second time.

c) Prove that, up to scalar multiplication, D is actually the *only* real orthogonal matrix whose diagonal entries are all the same and whose off-diagonal entries are all the same, other than the identity matrix. This provides a big hint about how one might have discovered D, if one didn't already know Grover's algorithm.

d) Show an actual circuit, made of Toffoli gates and Phase gates for applying A. Your circuit should have $O(n) = O(\log N)$ gates, and is allowed to use ancilla qubits initialized to any state you like, provided you return them to that state afterwards.

3. Someone claims they have a quantum algorithm to compute the parity of N bits in only 1 query, for arbitrarily large N. Their algorithm works as follows: assume without loss of generality that N is a power of 2. Split the input string X into two substrings X_0 and X_1 with $N/2$ bits each. In equal superposition, compute the parity p_0 of X_0 and the parity p_1 of X_1. We can then read out $p_0 + p_1 \pmod 2$, the parity of X itself, using Deutsch's algorithm (i.e., by measuring the superposition in the Hadamard basis). But how do we compute p_0 and p_1 themselves? Aha, by running the same algorithm recursively! In other words, each branch of the superposition further subdivides its string, X_0 or X_1, into two strings of $N/4$ bits, computes their parities in superposition, and combines the results using Deutsch's algorithm ... and so on, until we get down to the individual bits. Since only 1 query is needed at each level of the recursion, the total number of queries made to underlying string X is just $1 \cdot 1 ... \cdot 1 = 1$.

What's the error in this person's argument?

4. For the following problems a numerical answer is fine, while you could answer these questions by blindly plugging in to numerical solvers, for purposes of the final exam, we recommend you get some understanding of what's going on.

a) For each of the following Hamiltonians H, calculate e^{-iH}.

$$H_1 = \begin{bmatrix} 0 & \pi \\ \pi & 0 \end{bmatrix}; \qquad H_2 = \pi \begin{bmatrix} 1 & i \\ -i & 1 \end{bmatrix}; \qquad H_3 = \begin{bmatrix} 0 & 0 \\ 0 & 3 \end{bmatrix}$$

b) For each of the Hamiltonians above, calculate its ground state and its first excited state, and give the associated energies.

c) Describe all the possible Hamiltonians H such that e^{-iH} equals $\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$.

d) Give a 2×2 Hamiltonian H such that e^{-iH} equals the Hadamard matrix. [Hint: For this part and the next one, you'll make life easier for yourself if you start by diagonalizing.]

e) Give a 4×4 Hamiltonian H such that e^{-iH} equals the CNOT matrix.

f) For your Hamiltonian from part (e), what is $e^{-iH/2}$? (In other words, what's the associated "square root of the CNOT gate"?)

g) Give an example of a 2×2 matrix that has real eigenvalues but is not Hermitian – thereby showing that, while real eigenvalues are a necessary condition for a matrix to be Hermitian, they're not a sufficient one.

5. In the NP-complete 3-Coloring problem, we're given as input a description of a graph G, and the problem is to decide whether G is 3-colorable: that is, whether every vertex can be colored red, blue, or green, in such a way that no two adjacent vertices are colored the same. Given an n-vertex graph G, explain how to construct an n-qutrit Hamiltonian $H = H(G)$, as a sum of 2-qutrit terms, such that if G is 3-colorable, then H's ground states encode the 3-colorings.