QUANTUM COMPUTATION

EXERCISE SHEET 3 (v1.1)

Ashley Montanaro, DAMTP Cambridge
am994@cam.ac.uk

1. **The polynomial method.** This question aims to build expertise in working with polynomials for boolean functions.

   (a) Prove that any function $f : \{0,1\}^n \to \mathbb{R}$ has a unique representation as a multilinear polynomial.

   (b) Write down the polynomials representing the AND_n, OR_n and PARITY_n functions and hence verify that $\deg(\mathrm{AND}_n) = \deg(\mathrm{OR}_n) = \deg(\mathrm{PARITY}_n) = n$.

   (c) Show that $\widetilde{\deg}(\mathrm{PARITY}_n) = n$, and hence that any quantum query algorithm computing PARITY_n with success probability $2/3$ on every input requires $\Omega(n)$ queries to the input. (Hint: reduce PARITY_n to a univariate function and consider the behaviour of any polynomial approximating this function.)

   (d) Show that any quantum algorithm computing the OR_n function exactly must make at least n queries to the input, and hence can achieve no speed-up over classical algorithms. (Hint: consider the state of the computer just before the final measurement.)

2. **Factoring via phase estimation.** Fix two coprime positive integers x and N such that $x < N$, and let U_x be the unitary operator defined by $U_x|y\rangle = |xy \pmod N\rangle$. Let r be the order of x mod N (the minimal t such that $x^t \equiv 1$). For $0 \leq s \leq r-1$, define the states

$$|\psi_s\rangle := \frac{1}{\sqrt{r}} \sum_{k=0}^{r-1} e^{-2\pi i s k/r} \, |x^k \pmod N\rangle.$$

   (a) Verify that U_x is indeed unitary.

   (b) Show that, for arbitrary integer $n \geq 0$, $U_x^{2^n}$ can be implemented in time $\mathrm{poly}(n)$ (not $\mathrm{poly}(2^n)$!).

   (c) Show that each state $|\psi_s\rangle$ is an eigenvector of U_x with eigenvalue $e^{2\pi i s/r}$.

   (d) Show that

$$\frac{1}{\sqrt{r}} \sum_{s=0}^{r-1} |\psi_s\rangle = |1\rangle.$$

   (e) Thus show that, if the phase estimation algorithm with n qubits is applied to U_x using $|1\rangle$ as an "eigenvector", the algorithm outputs an estimate of s/r accurate up to n bits, for $s \in \{0, \dots, r-1\}$ picked uniformly at random, with constant probability.

   (f) Argue, following Section 6 of the first set of lecture notes, that this implies that the phase estimation algorithm can be used to factorise an integer N in $\mathrm{poly}(\log N)$ time.

3. **More efficient quantum simulation.**

   (a) Let A and B be Hermitian operators with $\|A\| \leq K$, $\|B\| \leq K$ for some $K \leq 1$. Show that
   $$e^{-iA/2}e^{-iB}e^{-iA/2} = e^{-i(A+B)} + O(K^3)$$
   (this is the so-called *Strang splitting*). Use this to give a more efficient approximation of k-local Hamiltonians by quantum circuits than the algorithm given in the notes, and calculate its complexity.

   (b) Let H be a Hamiltonian which can be written as $H = UDU^\dagger$, where U is a unitary matrix that can be implemented by a quantum circuit running in time poly(n), and $D = \sum_x d(x)|x\rangle\langle x|$ is a diagonal matrix such that the map $|x\rangle \mapsto e^{-id(x)t}|x\rangle$ can be implemented in time poly(n) for all x. Show that e^{-iHt} can be implemented in time poly(n).

4. **Other definitions of quantum walks.** In some sense, random walks require less space than quantum walks. A random walk on a graph for t steps can be concisely expressed as applying the t'th power of a matrix M to a vector. However, quantum walks as defined in this course use an additional coin. A simpler way to define a quantum walk in such a way that it respects the structure of a graph G with n vertices would be as repeated application of an n-dimensional unitary matrix U such that $U_{xy} = 0$ if and only if x and y are not connected. In other words, if A is the adjacency matrix of G ($A_{xy} = 1$ if x and y are connected, $A_{xy} = 0$ otherwise), $U_{xy} \neq 0 \Leftrightarrow A_{xy} = 1$. Call such quantum walks *concise*.

   (a) Consider the line with n vertices (i.e. vertices are numbered between 1 and n; vertices x and y are connected if $|x - y| = 1$). Show that no concise quantum walk can exist on this graph when n is odd, and that when n is even, any concise quantum walk only involves interactions between positions $(2k-1, 2k)$ for integer $k \geq 1$.

   (b) However, show that the hypercube does admit a concise quantum walk with non-trivial behaviour. (Hint: the adjacency matrix A_n of the dimension n hypercube can be written as
   $$A_n = \begin{pmatrix} A_{n-1} & I_{2^{n-1}} \\ I_{2^{n-1}} & A_{n-1} \end{pmatrix},$$
   where I_d is the d-dimensional identity matrix.)

   An alternative way to define a "concise" quantum walk on a graph, which is closer in spirit to classical *continuous-time* random walks, is as follows. For a graph with adjacency matrix A, and an arbitrary real time t, simply define the unitary matrix $U(t) = e^{-iAt}$, and define the amplitude of being at vertex y, given that the walk started at x and proceeded for time t, as $\langle y|U(t)|x\rangle$.

   (c) Show that the adjacency matrix of the n-dimensional hypercube can be written as $A_n = \sum_{j=1}^n X^{(j)}$, where $X^{(j)}$ denotes the operator which is a tensor product of $X = \left(\begin{smallmatrix} 0 & 1 \\ 1 & 0 \end{smallmatrix}\right)$ acting on the j'th qubit, and the identity elsewhere.

   (d) Hence show that $U(t) = e^{-iA_n t}$ factorises into a tensor product of 2×2 unitary matrices.

   (e) Hence show that there is a constant time t at which $\langle 1^n|U(t)|0^n\rangle = 1$, up to an overall phase, implying that this notion of quantum walk also admits fast hitting from vertices 0^n to 1^n on the hypercube.

5. **Optional (but fun): quantum oracle interrogation.** In this question, you will prove the following result of Wim van Dam.

Theorem 1. *Given oracle access to bits of an unknown n-bit string x, there is a quantum algorithm that learns x completely with success probability at least 0.999 using $n/2 + O(\sqrt{n})$ queries, for any x.*

This success probability can in fact be taken to be any constant strictly less than 1. Of course, classically we need precisely n queries to learn x with this worst-case success probability.

(a) Show that, for any $x \in \{0,1\}^n$, given the n qubit state

$$|\psi_x\rangle := \frac{1}{2^{n/2}} \sum_{y \in \{0,1\}^n} (-1)^{x \cdot y} |y\rangle,$$

there is a quantum algorithm that determines x with certainty using no additional queries to the bits of x. (Here $x \cdot y = \sum_i x_i y_i$ is the inner product of x and y modulo 2.)

(b) For any $0 \leq r \leq n$, consider the state

$$|\psi_x^r\rangle := \frac{1}{\sqrt{R}} \sum_{y \in \{0,1\}^n, |y| \leq r} (-1)^{x \cdot y} |y\rangle,$$

where $R = \sum_{i=0}^{r} \binom{n}{i}$. Show that, for some $r = n/2 + O(\sqrt{n})$, $|\langle \psi_x | \psi_x^r \rangle|^2 \geq 0.999$.

(c) Show that the state $|\psi_x^r\rangle$ can be produced using r queries to bits of x.

(d) Use parts (a)-(c) to prove Theorem 1.