Quantum Computing Exercises, October 13, 2000

Recall that Grover's algorithm searches for an i such that $f(i) = 1$, where $f : \{0,1\}^n \to \{0,1\}$. We will call such an i a "solution". The "search space" of this problem has size $N = 2^n$. The algorithm is:

$$G^k H^{\otimes n}|0^n\rangle.$$

Here $H^{\otimes n}$ is the Hadamard operation on n qubits, and the *Grover iterate* G is

$$G = -H^{\otimes n} O_0 H^{\otimes n} O_f,$$

where O_0 puts a -1 in front of $|0^n\rangle$ and does nothing to other basis states, and O_f corresponds to making a query: $O_f : |i\rangle \to (-1)^{f(i)}|i\rangle$. The output of the algorithm is obtained by measuring the final state.

The algorithm makes k queries, one in each Grover iterate. If there are t solutions, then setting $k \approx \frac{\pi}{4}\sqrt{\frac{N}{t}}$ will with high probability give a solution to the search problem. If the number t is unknown, then by trying exponentially decreasing values for t, we can still find a solution with an expected number of $O(\sqrt{N/t})$ queries (see [1] for the appropriate values of t that should be tried). In contrast, a classical bounded-error algorithm will need roughly N/t queries for this.

Exercises:

1. Show that $-H^{\otimes n} O_0 H^{\otimes n} = [2/N] - I$, where $[2/N]$ is the $N \times N$ matrix with all entries equal to $2/N$, and I is the $N \times N$ identity matrix.
2. Suppose $n = 2$, and $f(00) = f(01) = f(10) = 0$ and $f(11) = 1$. Give the intermediate and final superpositions in Grover's algorithm after each unitary transformation, for $k = 1$. What is the success probability?
3. Give the final superposition for the above f after $k = 2$ iterations. What is the success probability?
4. Show that if the number of solutions is $t = N/4$, then Grover's algorithm always finds a solution after just one query. How many queries would a classical algorithm need to find a solution with certainty if $t = N/4$? And if we allow some small error probability?

In the second hour of the Friday the 13th lecture, we will explain two applications of Grover's algorithm: finding the *minimum* in a list of numbers quickly [3], and finding a *collision* in a 2-to-1 function quickly [2]. Other applications, such as *quantum counting* the number of solutions, can be found in Chapter 6 of the book [4].

References

[1] M. Boyer, G. Brassard, P. Høyer, and A. Tapp. Tight bounds on quantum searching. *Fortschritte der Physik*, 46(4–5):493–505, 1998. Earlier version in Physcomp'96. quant-ph/9605034.

[2] G. Brassard, P. Høyer, and A. Tapp. Quantum algorithm for the collision problem. *ACM SIGACT News (Cryptology Column)*, 28:14–19, 1997. quant-ph/9705002.

[3] C. Dürr and P. Høyer. A quantum algorithm for finding the minimum. quant-ph/9607014, 18 Jul 1996.

[4] M. A. Nielsen and I. L. Chuang. *Quantum Computation and Quantum Information*. Cambridge University Press, 2000.