Lecture 14

Comment: I made two mistakes today, sorry! (1) my first definition of D_1 was wrong, which was pointed out by a student, Joey; my second "alternative" definition is the correct one (2) my claim that the deterministic query complexity of Simon's problem isequal to $2^{n-1}+1$ is wrong: while this is an upper bound, the tight bound is $\Theta(2^{n/2})$ like in the randomized case. For more details, see below.

Even within the query model, it is unsatisfactory that the DJ speedup only holds when we demand certain correctness. *Question: can we have an exponential speedup in the query model if we don't demand certain correctness, but say 99.99% correctness?* It turns out the answer is yes, as can be witnessed by Simon's problem. This problem inspired Shor's algorithm, which in some sense instantiates the quantum oracle in Simon's problem as a specific circuit yet the exponential speedup persists as far as we know.

Simon's problem

Definition 14 (Simon's problem)**.** For $n \in \mathbb{N}$, define the set of functions:

$$D_0 = \left\{ f : \{0,1\}^n \to \{0,1\}^n \;\middle|\; f \text{ is a bijection} \right\},$$
$$D_1 = \left\{ f : \{0,1\}^n \to \{0,1\}^n \;\middle|\; \text{there exists unique } s \neq 0^n \text{ such that for all } x, y \in \{0,1\}^n \colon f(x) = f(y) \iff x \in \{y, y \oplus s\} \right\}.$$

Problem: given query access to $f \in D_0 \cup D_1$, determine whether $f \in D_0$ or $f \in D_1$.

Functions $f \in D_1$ are 2-to-1, i.e., every image of f has exactly two preimages. Comment: illustrate by cube with 4 colors; the preceding sentence means that each color appears exactly twice (as we saw in our example).

Warning: the definition of D_1 aboveis the alternative definition I gave in class. In fact, the first definition I gave of D_1, namely

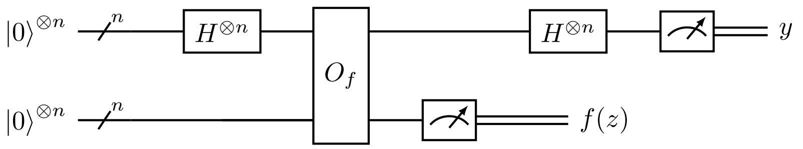
$$\left\{ f : \{0,1\}^n \to \{0,1\}^n \;\middle|\; \text{there exists unique } s \neq 0^n \text{ such that for all } x \in \{0,1\}^n \colon f(x) = f(x \oplus s) \right\},$$

is wrong. Functions in this set might not be 2-to-1. (It may be fun to see why.)

Classical query complexity. For deterministic computation, the query complexity is $\leq 2^{n-1}+1$. But, unlike what I said in class, this is *not* the tight bound. In fact, the tight bound is $\Theta(2^{n/2})$ like in the randomized case. For the upper bound, the idea is sort of like a "derandomized" version of the randomized algorithm below. For more, see John Watrous's answer to this [StackExchange post]. For randomized computation, the query complexity is $\Theta(2^{n/2})$.

1. Upper bound. Consider querying the value of f on a random subset of M points. The probability that a pair of (distinct) points map to the same value under f is $1/(2^n-1) \approx 1/2^n$. So if we know the value of f on $\approx 2^n$ pairs then we can get the probability close to $2^n \times 1/2^n = 1$.[5] But to get the value of f on $\approx 2^n$ pairs, only need to query f on $M \approx 2^{n/2}$ points. (Related to birthday paradox.)
2. Lower bound. The intuition is that any pair of inputs mapping to distinct values *only* rules out one s so need to query f on at least $\Omega(2^{n/2})$ inputs to rule out all possible s.

Quantum query complexity. The quantum algorithm solves Simon's problem with $O(n)$ queries:



Analysis:

1. Initialize with $|0^n\rangle\,|0^n\rangle$.
2. Apply $H^{\otimes n}$ to the first register.
$$\frac{1}{\sqrt{2^n}} \sum_{x \in \{0,1\}^n} |x\rangle\,|0^n\rangle \tag{79}$$
3. Apply O_f: $|x\rangle\,|y\rangle \mapsto |x\rangle\,|y \oplus f(x)\rangle$ to obtain
$$\frac{1}{\sqrt{2^n}} \sum_{x} |x\rangle\,|f(x)\rangle \tag{80}$$

[5] This is a hand wave as probability is not additive like this, more precisely $\Pr[A \cup B] \neq \Pr[A] + \Pr[B]$ in general.