CS 294-2 NP-complete Problems: lower bounds + Zeno Effect 2/21/07
Spring 2007 Lecture 10

0.0.1 NP-Completeness

Consider SAT, the prototypical example of an NP-complete problem. An instance of this problem consists of a Boolean function $f(x_1,\ldots,x_n) = c_1 \wedge \ldots \wedge c_m$; the SAT problem asks you to determine whether there exists a satisfying assignment—that is, an input $(a_1,\ldots,a_n)$ such that $f(a_1,\ldots,a_n) = 1$. UNIQUE-SAT is a variant of SAT that poses the same problem with the restriction that f must have zero or one satisfying assignments, but no more. As it turns out, there is a randomized reduction from SAT to UNIQUE-SAT; thus, the two problems are equally hard.

We'll use the black box model when considering this problem. In this model, we know that either $f \equiv 0$ or there exists exactly one a such that $f(a) = 1$, where a is chosen uniformly at random. That is, f is treated as a black box; we can make queries to f, but we have no access to the Boolean formula itself. Equivalently we can represent f by a table of $N = 2^n$ entries where either none or exactly one entry is 1. Ideally we want a quantum algorithm that solves this problem in time $O(\text{poly}(n)) = O(poly - log(n))$.

Can a quantum computer solve this problem by going into a superposition of all exponentially many possible truth assignments? To answer this question precisely, let us define the black box query model:

0.0.2 The quantum black box model

Here's the problem: You are given a boolean function $f : \{1,\ldots,N\} \to \{0,1\}$, and are promised that for exactly one $a \in \{1,\ldots,N\}$, $f(a) = 1$. Think of this as a table of size N, where exactly one element has value 1, and all the others are 0. Since we assume f can be computed classically in polynomial time, we can also compute it in superposition:

$$\sum_x \alpha_x |x\rangle\,|0\rangle \to \sum_x \alpha_x |x\rangle\,|f(x)\rangle$$

Another way we can implement f is put the answer register in superposition:

Now, we might as well assume f is a black box or oracle. All we need to do is design an algorithm that finds $a : f(a) = 1$.

0.0.3 The Hybrid Argument

For the purposes of this discussion, we want to separate the quantum algorithm itself from the function f. We assume that the quantum algorithm is infinitely powerful (i.e., it can do any computation in one step) and focus instead on the number of queries it must make to f. All queries to f occur in superposition; that is, a single query on $\sum_x \alpha_x |x\rangle\,|0\rangle$ yields the output $\sum_x \alpha_x |x\rangle\,|f(x)\rangle$.

Theorem 10.1: *In the black box model, any quantum algorithm for determining whether there exist $x_1,\ldots,x_n$ such that $f(x_1,\ldots,x_n) = 1$ must make $\Omega(\sqrt{N})$ queries to f.*

Proof:

Consider any quantum algorithm A for solving this search problem. First do a test run of A on the function $f \equiv 0$. Let T be the number of queries that A makes to f, and let $\alpha_{x,t}$ be the amplitude with which A queries x at time t (that is, the query at time t is $\sum_x \alpha_{x,t} \left|x\right\rangle$). Now, define the query magnitude of x to be $\sum_t |\alpha_{x,t}|^2$. The expectation value of the query magnitude of x is $E_x\left(\sum_t |\alpha_{x,t}|^2\right) = T/N$. Thus $\min_x\left(\sum_t |\alpha_{x,t}|^2\right) \leq T/N$. Let z be the input at which this minimum occurs; then by the Cauchy-Schwarz inequality, $\sum_t |\alpha_{z,t}| \leq T/\sqrt{N}$. [1]

Let $\left|\phi_t\right\rangle$ be the states of A_f after the t-th step. Now run the algorithm A on the function g such that $g(z) = 1$ and for all $y \neq z$, $g(y) = 0$. Suppose the final state of A_g is $\left|\psi_T\right\rangle$. By the claim that follows, $\left|\phi_T\right\rangle - \left|\psi_T\right\rangle = \left|E_0\right\rangle + \ldots + \left|E_{T-1}\right\rangle$ where $||\left|E_t\right\rangle|| \leq \sqrt{2}|\alpha_{z,t}|$. Using the triangle inequality and the inequality proved above, we have $||\left|\phi_T\right\rangle - \left|\psi_T\right\rangle|| \leq \sum_t ||\left|E_t\right\rangle|| \leq \sqrt{2}\sum_t |\alpha_{z,t}| \leq T\sqrt{2/N}$. This implies that the two states can be distinguished with probability at most $O(T/\sqrt{N})$ by any measurement. Thus any quantum algorithm that distinguishes f from g with constant probability of success must make $\Omega(\sqrt{N})$ queries.

■

$\left|\psi_T\right\rangle = \left|\phi_T\right\rangle + \left|E_0\right\rangle + \left|E_1\right\rangle + \ldots + \left|E_{T-1}\right\rangle$, where $||\left|E_t\right\rangle|| \leq \sqrt{2}|\alpha_{z,t}|$.

Proof:

Consider two runs of the algorithm A, which differ only on the t-th step. The first run queries the function f on the first t steps and queries g for the remaining $T-t$ steps; the second run queries f on the first $t-1$ steps and g for the remaining $T-t+1$ steps. After the first $t-1$ steps, both runs are in state $\left|\phi_t\right\rangle$. On the t-th step, one run queries f and the other queries g. The outputs of these queries differ only on the amplitude of the two basis vectors $\left|z\right\rangle\left|0\right\rangle$ and $\left|z\right\rangle\left|1\right\rangle$, so overall the output vectors differ by at most $\sqrt{2}|\alpha_{z,t}|$. Thus, at the end of the t-th step, the state of the first run is $\left|\phi_t\right\rangle$, whereas the state of the second run is $\left|\phi_t\right\rangle + \left|F_t\right\rangle$, where $||\left|F_t\right\rangle|| \leq \sqrt{2}|\alpha_{z,t}|$. Now if U is the unitary transform describing the remaining $T-t$ steps (of both runs), then the final state after T steps for the two runs are $U\left|\phi_t\right\rangle$ and $U(\left|\phi_t\right\rangle + \left|F_t\right\rangle)$, respectively. The latter state can be written as $U\left|\phi_t\right\rangle + \left|E_t\right\rangle$, where $\left|E_t\right\rangle = U\left|F_t\right\rangle$. Since unitary transformations preserve length, we know that $||\left|E_t\right\rangle|| \leq \sqrt{2}|\alpha_{z,t}|$. Thus, the effect of switching the queried function only on the t-th step can be described by an "error" $\left|E_t\right\rangle$ in the final state of the algorithm, where $||\left|E_t\right\rangle|| \leq \sqrt{2}|\alpha_{z,t}|$.

We can transform the run A_f to A_g by a succession of T changes of the kind described above. Overall, the difference between the final states of A_f and A_g is $\left|E_0\right\rangle + \left|E_1\right\rangle + \ldots + \left|E_{T-1}\right\rangle$, where $\left\|\left|E_t\right\rangle\right\| \leq \sqrt{2}|\alpha_{z,t}|$.

■

Finally, it is useful to consider where this factor of $\sqrt{N}$ comes from. In the worst case, we query z with amplitude $1/\sqrt{N}$ at each time step. The vectors that indicate the differences at each step could all be orthogonal, in which case the total distance is the sum of the squares of each vector's length, which is about N. However, if all vectors are in the same direction, the total distance is the sum of the length of each vector, which is approximately $\sqrt{N}$. Grover's algorithm, which we will describe next, demonstrates that it is possible to align all of these vectors and achieve the factor of $\sqrt{N}$.

Something about relativization, as well as about inverting permutations.

Vaidman's Bomb

To illustrate some of the concepts behind Grover's algorithm, we'll briefly consider a problem known as Vaidman's bomb. In this problem, we have a package that may or may not contain a bomb. However, the bomb is so sensitive that simply looking to see if the bomb exists will cause it to explode. So, can we determine whether the package contains a bomb without setting it off? Paradoxically, quantum mechanics says that we can. In particular, we will demonstrate that there is a sequence of N measurements such that if the package contains a bomb, we will look with probability $1/N$, and if the package does not contain a bomb, we will look with certainty.

The Quantum Zeno Effect

[1] The Cauchy-Schwarz inequality says that for two vectors a and b of length T, $(\sum_t a_t b_t)^2 \leq \left(\sum_t a_t^2\right)\left(\sum_t b_t^2\right)$. If we let $b_t = 1$ for all t, then we have $(\sum_t a_t)^2 \leq T\sum_t a_t^2$. Thus, if $\sum_t |\alpha_{z,t}|^2 \leq T/N$, then $(\sum_t |\alpha_{z,t}|)^2 \leq T^2/N$.

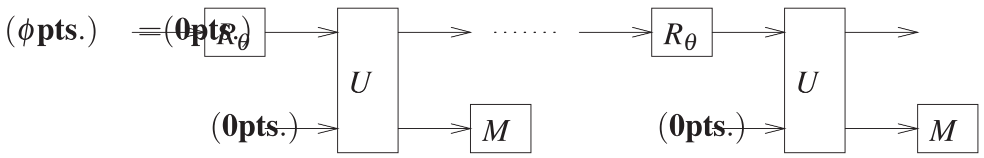

Figure 0.1: This figure shows the circuit used for finding a bomb. There are N steps, and in each step we rotate the control bit and run U.

To achieve this goal, we'll take advantage of a phenomenon known as the Quantum Zeno Effect (also referred to as the "watched pot" or the "watchdog" effect). Consider a quantum state consisting of a single qubit. This qubit starts at $|0\rangle$, and at every step we will rotate it toward $|1\rangle$ by $\theta = \pi/2N$. After one rotation, we have $|\phi\rangle = \alpha|0\rangle + \beta|1\rangle$, where $\beta = \sin\theta \approx 1/N$. After N steps, the state will be $|1\rangle$, so any measurement will return $|1\rangle$ with high probability.

Now what if we decide to measure the state after each rotation? After the first rotation, we will measure $|0\rangle$ with high probability, *but this measurement collapses the state back to* $|0\rangle$. Thus, each measurement has a high probability of yielding $|0\rangle$; the probability of getting $|1\rangle$ by the end is approximately $N\frac{1}{N^2} = \frac{1}{N}$, as opposed to the extremely high probability in the previous case.

Essentially, the Quantum Zeno Effect says that if we have a quantum state that is in transition toward a different state, making frequent measurements can delay that transition by repeatedly collapsing the qubit back to its original state.

Looking for the Bomb

To determine whether Vaidman's bomb exists without actually looking at it, we want to take advantage of the Quantum Zeno Effect. We'll have a control qubit that indicates whether or not we plan to look at the contents of the package, and we'll have a measurement that collapses this qubit back to $|0\rangle$ in the case that a bomb is present.

We'll assume that we have a device that can measure whether a bomb is present. We will model this device as a quantum circuit U that has one input (the control qubit $|\phi\rangle$) and one output (a qubit that is $|1\rangle$ if a bomb is definitely present). If there is no bomb, then U maps $|\phi\rangle|0\rangle \mapsto |\phi\rangle|0\rangle$; in other words, U behaves as the identity. If there is a bomb, then U maps $|0\rangle|0\rangle \mapsto |0\rangle|0\rangle$ (there's a bomb, but we didn't look) and $|1\rangle|0\rangle \mapsto |1\rangle|1\rangle$ (we looked at the bomb); that is, U behaves as a CNOT gate.

We want to figure out whether there is a bomb (i.e., we want to test U's behavior) without setting off the bomb very often. Figure 0.1 shows the circuit we will use. We initialize the control qubit $|\phi\rangle$ to $|0\rangle$. In each step of the algorithm, we rotate the control qubit toward $|1\rangle$ by θ and then run U; we'll execute the algorithm for N steps.

Consider the case where there is no bomb; our initial input is $|0\rangle|0\rangle$. If we rotate the control qubit by θ, the input to the first U gate is $(\alpha|0\rangle + \beta|1\rangle)|0\rangle$, and the output of U is the same state (since U is the identity). Measuring the output qubit always returns $|0\rangle$ and doesn't alter the state; thus, each step rotates the qubit further until $\beta = 1$ at the last measurement.

Now consider the case where there is a bomb. Once again, our initial input is $|00\rangle$. After the first rotation, the input to U is $(\alpha|0\rangle + \beta|1\rangle)|0\rangle$, and the output of U is $\alpha|0\rangle|0\rangle + \beta|1\rangle|1\rangle$. When we measure the last qubit, we have a $\beta^2 \approx 1/N^2$ probability of looking at the bomb and setting it off. Otherwise, we measure $|0\rangle$ for the output qubit, which means we didn't look at the bomb. However, *this measurement collapses the state back to* $|0\rangle|0\rangle$. Thus, subsequent steps in the algorithm will simply repeat this process. Overall, we only have a $N\beta^2 \approx 1/N$ chance of actually looking at the bomb.

Vaidman and Grover

To see the relationship to Grover's algorithm, consider a particularly unfortunate case where we have N packages, $N-1$ of which contain bombs. We want to find the one package that does not contain a bomb, though we don't mind setting off a few of the bombs in the process. Grover's algorithm has a property similar to Vaidman's method where the amplitude of one target basis vector is amplified while all others are constantly diminished or reset.

The important thing to note is that it's highly counterintuitive to be able to search in $\sqrt{N}$ steps. By querying in superposition, we manage to search using fewer steps than there are locations to search!