Lecture 13: Grover's Algorithm (continued)

March 9, 2006

In the previous lecture we stated Grover's Algorithm and began analyzing it. Today we will complete this analysis and discuss how the algorithm can be used to solve various searching problems. For convenience, here is the algorithm again:

Grover's Algorithm

1. Let X be an n-qubit quantum register with initial state $|0^n\rangle$. Perform $H^{\otimes n}$ on X.
2. Apply to the register X the transformation

   $$G = -H^{\otimes n} Z_0 H^{\otimes n} Z_f$$

   k times (where k will be specified later).
3. Measure X and output the result.

The operations Z_f and Z_0 are defined as

$$Z_f |x\rangle = (-1)^{f(x)} |x\rangle ,$$

where $f : \{0,1\}^n \to \{0,1\}$ is the function being considered and for which we have a black box B_f, and

$$Z_0 |x\rangle = \begin{cases} -|x\rangle & \text{if } x = 0^n \\ |x\rangle & \text{if } x \neq 0^n. \end{cases}$$

The algorithm requires k queries to the black box B_f for whatever value of k we choose in step 2.

For the analysis, we had defined sets A and B as follows:

$$A = \{x \in \{0,1\}^n \,:\, f(x) = 1\} ,$$

$$B = \{x \in \{0,1\}^n \,:\, f(x) = 0\} ,$$

and let $a = |A|$ and $b = |B|$. The case where either $a = 0$ or $b = 0$ turns out to be an easy special case, so assumed for the time being that $a \neq 0$ and $b \neq 0$. We also defined states

$$|A\rangle = \frac{1}{\sqrt{a}} \sum_{x \in A} |x\rangle \quad \text{and} \quad |B\rangle = \frac{1}{\sqrt{b}} \sum_{x \in B} |x\rangle ,$$

which are both unit vectors and are orthogonal to one another. Because the state of register X immediately after step 1 in the algorithm is given by

$$|h\rangle \stackrel{\text{def}}{=} H^{\otimes n} |0^n\rangle = \frac{1}{\sqrt{2^n}} \sum_{x \in \{0,1\}^n} |x\rangle ,$$

we can write

$$|h\rangle = \sqrt{\frac{a}{N}}\,|A\rangle + \sqrt{\frac{b}{N}}\,|B\rangle\,.$$

Thus, the state of X before step 2 is a vector in the subspace spanned by $\{|A\rangle\,, |B\rangle\}$. The last thing we did was to determine the effect of the operation G on $|A\rangle$ and $|B\rangle$:

$$G\,|A\rangle = \left(1 - \frac{2a}{N}\right)|A\rangle - \frac{2\sqrt{ab}}{N}\,|B\rangle$$
$$G\,|B\rangle = \frac{2\sqrt{ab}}{N}\,|A\rangle - \left(1 - \frac{2b}{N}\right)|B\rangle\,.$$

This implies that the state of X after each application of G will remain in the subspace spanned by $|A\rangle$ and $|B\rangle$.

Now, we can express the action of G on the space spanned by $\{|A\rangle\,, |B\rangle\}$ as a matrix:

$$M = \begin{bmatrix} -\left(1 - \frac{2b}{N}\right) & -\frac{2\sqrt{ab}}{N} \\ \frac{2\sqrt{ab}}{N} & \left(1 - \frac{2a}{N}\right) \end{bmatrix} = \begin{bmatrix} \frac{b-a}{N} & -\frac{2\sqrt{ab}}{N} \\ \frac{2\sqrt{ab}}{N} & \frac{b-a}{N} \end{bmatrix}.$$

Here, the first row and column correspond to $|B\rangle$ and the second to $|A\rangle$. Notice that

$$\begin{bmatrix} \sqrt{\frac{b}{N}} & -\sqrt{\frac{a}{N}} \\ \sqrt{\frac{a}{N}} & \sqrt{\frac{b}{N}} \end{bmatrix}^2 = \begin{bmatrix} \frac{b-a}{N} & -\frac{2\sqrt{ab}}{N} \\ \frac{2\sqrt{ab}}{N} & \frac{b-a}{N} \end{bmatrix} = M,$$

which shows that M is a rotation as we originally hypothesized. Specifically, if $\theta \in (0, \pi/2)$ is the angle that satisfies

$$\sin\theta = \sqrt{\frac{a}{N}} \quad \text{and} \quad \cos\theta = \sqrt{\frac{b}{N}}$$

then

$$R_{2\theta} = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix}^2 = \begin{bmatrix} \sqrt{\frac{b}{N}} & -\sqrt{\frac{a}{N}} \\ \sqrt{\frac{a}{N}} & \sqrt{\frac{b}{N}} \end{bmatrix}^2 = M.$$

In other words, G causes a rotation by an angle 2θ in the space spanned by $\{|A\rangle\,, |B\rangle\}$ for

$$\theta = \sin^{-1}\sqrt{\frac{a}{N}}.$$

As stated above, the register X is in the state

$$|h\rangle = \sqrt{\frac{b}{N}}\,|B\rangle + \sqrt{\frac{a}{N}}\,|A\rangle = \cos\theta\,|B\rangle + \sin\theta\,|A\rangle$$

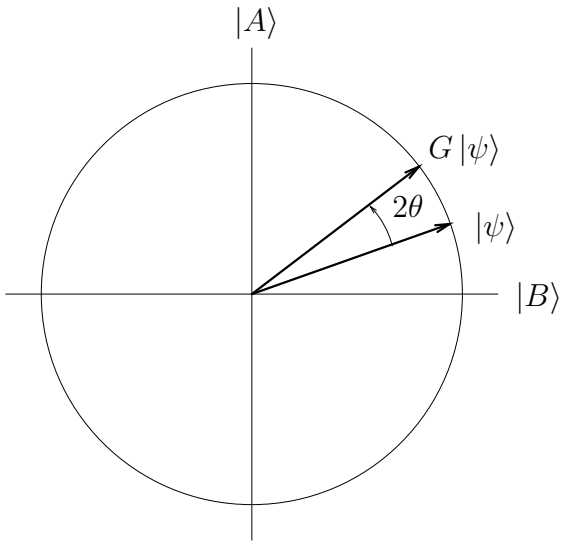

Figure 1: One iteration of G induces a rotation by an angle 2θ.

immediately after step 1 is performed. After k iterations of G, the state will be

$$\cos((2k+1)\theta)\,|B\rangle + \sin((2k+1)\theta)\,|A\rangle\,.$$

The goal is to measure some element $x \in A$, so we would like the state of X to be as close to $|A\rangle$ as possible. If we want

$$\sin((2k+1)\theta) \approx 1$$

then

$$(2k+1)\theta \approx \frac{\pi}{2}$$

will suffice, so we should choose

$$k \approx \frac{\pi}{4\theta} - \frac{1}{2}.$$

Of course k must be an integer, which is why we can only hope to approximate this quantity.

Suppose $a = 1$. Then

$$\theta = \sin^{-1}\sqrt{\frac{1}{N}} \approx \frac{1}{\sqrt{N}}$$

so

$$k = \left\lfloor \frac{\pi}{4}\sqrt{N} \right\rfloor$$

is a reasonable choice for our algorithm. Remember that k is also the number of queries to B_f needed by the algorithm, so we are apparently on the right track to proving that the algorithm needs $O(\sqrt{N})$ queries. Although the case $a = 1$ only represents a special case, it is clearly an interesting special case. With the above choice of k, the probability of finding the single x such that $f(x) = 1$ is

$$\sin^2\left(\left(2\left\lfloor \pi\sqrt{N}/4 \right\rfloor + 1\right)\sin^{-1}\left(1/\sqrt{N}\right)\right).$$

The limit of this probability is 1 as N goes to infinity, and it is always at least $1/2$. Here are some sample values:

N	success prob.
2	.5
4	1.0
8	.9453125
16	.9613190
32	.9991823
64	.9965857
128	.9956199
256	.9999470
512	.9994480
1024	.9994612
2048	.9999968
4096	.9999453

So, even in the worst case, repeating the algorithm some small constant number of times and evaluating f at the output each time will find the unique x such that $f(x) = 1$ with very high probability.

In the general case when we do not know that $a = 1$, the situation is more challenging. For instance, if $a = 4$ but we still choose

$$k = \left\lfloor \pi\sqrt{N}/4 \right\rfloor,$$

the success probability goes to 0 in the limit of large N. This is because we are effectively rotating twice as far as we should. There are different strategies for dealing with this problem. One possibility is simply to choose a *random* value

$$k \in \left\{1, \ldots, \sqrt{N} + 1\right\}.$$

Still operating under the assumption that $a \geq 1$, the algorithm will fail to find an x with $f(x) = 1$ with probability at most $3/4$. Repeating some constant number of times until a "good" x is found quickly decreases error.

A somewhat better strategy is to apply the method above for successively larger values. Specifically, instead of choosing a random k in the range $1, \ldots, \sqrt{N} + 1$, do the following:

1. Set $m = 1$.
2. Choose $k \in \{1, \ldots, m + 1\}$ uniformly and run Grover's algorithm for this choice of k. If the algorithm finds an x such that $f(x) = 1$, then output x and halt.
3. If $m > \sqrt{N}$ then "fail". Else, set $m = \lfloor (8/7)m \rfloor$ and goto step 2.

This will succeed in finding $x \in A$ with probability at least 1/4 after

$$O(\sqrt{N/a})$$

queries. By repeating step 2 a constant number of times during each iteration of the loop, the error decreases exponentially. (The number 8/7 just happens to be a number that works for the analysis, which we will not discuss. The point is that m increases exponentially, but not too fast to cause the algorithm to fail.)

Finally, we still need to deal with the special cases where $a = 0$ or $b = 0$. Obviously if $a = 0$, the output x of the algorithm will never satisfy $f(x) = 1$ because there are no such values of x. It is easy to check directly that Grover's Algorithm will output a choice of $x \in \{0, 1\}^n$ that is uniformly distributed in this case. Supposing that we do not know a, if we run Grover's Algorithm as described above and always measure a value of x for which $f(x) = 0$, we can reasonably conclude with high confidence that $a = 0$.

It is also the case when $b = 0$ that Grover's Algorithm will output a uniformly distributed choice of x. Of course in this case $f(x) = 1$ for all x, so the problem is trivially solved.

This represents the end of our discussion of Grover's algorithm, and of quantum algorithms in general. The next topics coming up in the course are quantum error correction and quantum cryptography, with a short discussion of a more general mathematical description of quantum information than the one we have been using coming first.