Lecture 5: A simple searching algorithm; the Deutsch-Jozsa algorithm

January 31, 2006

In the previous lecture we discussed Deutsch's Algorithm, which gives a simple example of how quantum algorithms can give some advantages over classical algorithms in certain restricted settings. We will start this lecture by discussing another simple example, and then move on to the Deutsch-Jozsa Algorithm, which generalizes Deutsch's Algorithm to functions from n bits to 1 bit.

A simple searching problem

Suppose that we are given a function of the form $f : \{0,1\}^2 \to \{0,1\}$, but this time we are *promised* that it is one of these four functions:

f_{00}

input	output
00	1
01	0
10	0
11	0

f_{01}

input	output
00	0
01	1
10	0
11	0

f_{10}

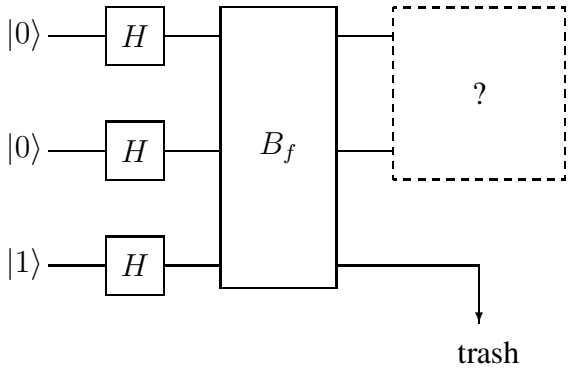
input	output
00	0
01	0
10	1
11	0

f_{11}

input	output
00	0
01	0
10	0
11	1

The goal is to determine which one it is. In other words, the function takes value 1 on one input and value 0 on all others, and the goal is to find the input on which the function takes value 1. Thus, we can think of this problem as representing a simple searching problem. As before, we assume that access to the function is restricted to evaluations of the transformation B_f. Now is a good time to mention some terminology: we say that an evaluation of B_f on some input (quantum or classical) is a *query*.

Classically, three queries are necessary and sufficient to solve the problem. In the quantum case, one query will be sufficient to solve the problem. Here is a circuit diagram describing part of the procedure. (The missing part will be filled in later.)



Let us analyze the state of the above circuit at various instants to try and determine what (if anything) would work for the omitted part of the circuit. The state after the Hadamard transforms can be written

$$\left(\frac{1}{2}|00\rangle + \frac{1}{2}|01\rangle + \frac{1}{2}|10\rangle + \frac{1}{2}|11\rangle\right)\left(\frac{1}{\sqrt{2}}|0\rangle - \frac{1}{\sqrt{2}}|1\rangle\right).$$

After the transformation B_f is performed, I claim that the state is

$$\left(\frac{1}{2}(-1)^{f(00)}|00\rangle + \frac{1}{2}(-1)^{f(01)}|01\rangle + \frac{1}{2}(-1)^{f(10)}|10\rangle + \frac{1}{2}(-1)^{f(11)}|11\rangle\right)\left(\frac{1}{\sqrt{2}}|0\rangle - \frac{1}{\sqrt{2}}|1\rangle\right).$$

The reasoning is the same as in the analysis of Deutsch's Algorithm—we are again using the *phase kickback* effect.

The last qubit is independent of the first two, so when it goes in the trash we are left with one of these four possibilities for the state of the first two qubits:

$$\begin{aligned}
f = f_{00} &\Rightarrow |\phi_{00}\rangle = -\frac{1}{2}|00\rangle + \frac{1}{2}|01\rangle + \frac{1}{2}|10\rangle + \frac{1}{2}|11\rangle \\
f = f_{01} &\Rightarrow |\phi_{01}\rangle = \frac{1}{2}|00\rangle - \frac{1}{2}|01\rangle + \frac{1}{2}|10\rangle + \frac{1}{2}|11\rangle \\
f = f_{10} &\Rightarrow |\phi_{10}\rangle = \frac{1}{2}|00\rangle + \frac{1}{2}|01\rangle - \frac{1}{2}|10\rangle + \frac{1}{2}|11\rangle \\
f = f_{11} &\Rightarrow |\phi_{11}\rangle = \frac{1}{2}|00\rangle + \frac{1}{2}|01\rangle + \frac{1}{2}|10\rangle - \frac{1}{2}|11\rangle
\end{aligned}$$

There is something special about these four states: they form an *orthonormal set*. This means that they are unit vectors and are *pairwise orthogonal*:

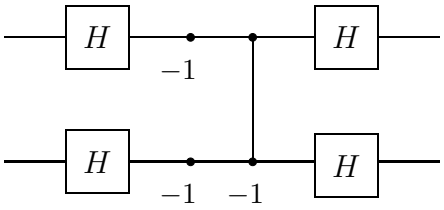
$$\langle\phi_{ab}|\phi_{cd}\rangle = \begin{cases} 1 & \text{if } a = c \text{ and } b = d \\ 0 & \text{otherwise.} \end{cases}$$

Whenever you have an orthonormal set like this, it is always possible to build a quantum circuit that exactly distinguishes the states. In fact, the corresponding unitary transformation is easy to obtain: you let the vectors form the columns of a matrix and then take the conjugate transform. In this case we want this matrix:

$$U = \begin{pmatrix} -\frac{1}{2} & \frac{1}{2} & \frac{1}{2} & \frac{1}{2} \\ \frac{1}{2} & -\frac{1}{2} & \frac{1}{2} & \frac{1}{2} \\ \frac{1}{2} & \frac{1}{2} & -\frac{1}{2} & \frac{1}{2} \\ \frac{1}{2} & \frac{1}{2} & \frac{1}{2} & -\frac{1}{2} \end{pmatrix}.$$

You can check that $U|\phi_{ab}\rangle = |ab\rangle$ for all four choices of $a, b \in \{0, 1\}$.

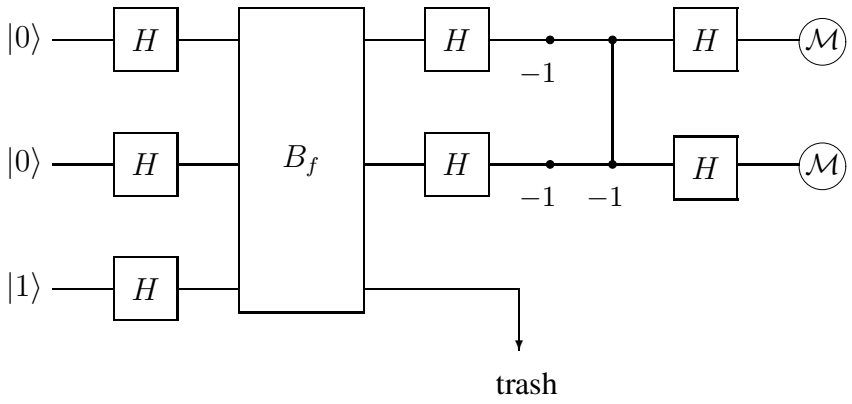
In case you are interested, it is possible to build a small circuit that uses gates that we have already seen to perform the unitary transformation U:



The gate consisting of a single circle labeled -1 corresponds to the matrix

$$\begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix},$$

which is also called a *phase flip* or σ_z gate. The complete circuit looks like this:



The Deutsch-Jozsa Algorithm

Our next algorithm is a generalization of Deutsch's Algorithm called the Deutsch-Jozsa Algorithm. Although the computational problem being solved is still artificial and perhaps not particularly impressive, we are moving toward more interesting problems.

This time we assume that we are given a function $f : \{0,1\}^n \to \{0,1\}$, where n is some arbitrary positive integer, and we are promised that one of two possibilities holds:

1. f is **constant**. In other words, either $f(x) = 0$ for all $x \in \{0,1\}^n$ or $f(x) = 1$ for all $x \in \{0,1\}^n$.
2. f is **balanced**. This means that the number of inputs $x \in \{0,1\}^n$ for which the function takes values 0 and 1 are the same:

$$|\{x \in \{0,1\}^n : f(x) = 0\}| = |\{x \in \{0,1\}^n : f(x) = 1\}| = 2^{n-1}.$$

The goal is to determine which of the two possibilities holds.

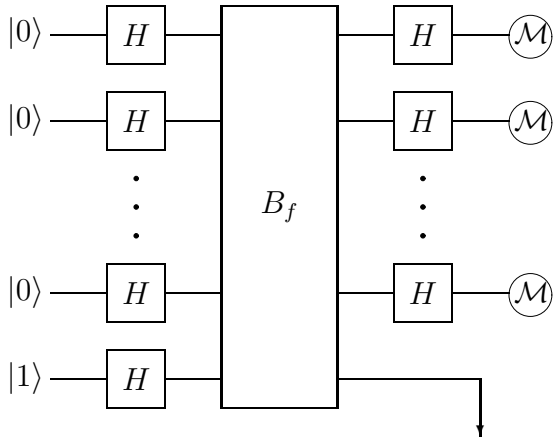
As for the previous two algorithms, we assume that access to the function f is restricted to queries to a device corresponding to the transformation B_f defined similarly to before:

$$B_f |x\rangle |b\rangle = |x\rangle |b \oplus f(x)\rangle$$

for all $x \in \{0,1\}^n$ and $b \in \{0,1\}$.

It turns out that classically this problem is pretty easy given a small number of queries if we allow randomness and accept that there may be a small probability of error. Specifically, we can randomly choose say k inputs $x_1, \ldots, x_k \in \{0,1\}^n$, evaluate $f(x_i)$ for $i = 1, \ldots, k$, and answer "constant" if $f(x_1) = \cdots = f(x_k)$ and "balanced" otherwise. If the function really was constant this method will be correct every time, and if the function was balanced, the algorithm will be wrong (and answer "constant") with probability $2^{-(k-1)}$. Taking $k = 11$, say, we get that the probability of error is smaller than $1/1000$. However, if you demand that the algorithm is correct every time, then $2^{n-1} + 1$ queries are needed in the worst case.

In the quantum case, 1 query will be sufficient to determine with certainty whether the function is constant or balanced. Here is the algorithm, which is called the *Deutsch-Jozsa Algorithm*:



There are n bits resulting from the measurements. If all n measurement results are 0, we conclude that the function was constant. Otherwise, if at least one of the measurement outcomes is 1, we conclude that the function was balanced.

Before we analyze the algorithm, it will be helpful to think more about Hadamard transforms. We have already observed that for $a \in \{0,1\}$ we have

$$H |a\rangle = \frac{1}{\sqrt{2}} |0\rangle + \frac{1}{\sqrt{2}} (-1)^a |1\rangle ,$$

which we can also write as

$$H |a\rangle = \frac{1}{\sqrt{2}} \sum_{b \in \{0,1\}} (-1)^{ab} |b\rangle .$$

If we instead had two qubits, starting in state $|x\rangle$ for $x = x_1x_2 \in \{0,1\}^2$, and applied Hadamard transforms to both, we would obtain

$$\begin{aligned}(H \otimes H)|x\rangle &= \left(\frac{1}{\sqrt{2}} \sum_{y_1 \in \{0,1\}} (-1)^{x_1 y_1} |y_1\rangle\right) \left(\frac{1}{\sqrt{2}} \sum_{y_2 \in \{0,1\}} (-1)^{x_2 y_2} |y_2\rangle\right) \\ &= \frac{1}{2} \sum_{y \in \{0,1\}^2} (-1)^{x_1 y_1 + x_2 y_2} |y\rangle .\end{aligned}$$

The pattern generalizes to any number of qubits. Writing $H^{\otimes n}$ to mean $H \otimes \cdots \otimes H$ (n times), we have

$$H^{\otimes n}|x\rangle = \frac{1}{\sqrt{2^n}} \sum_{y \in \{0,1\}^n} (-1)^{x_1 y_1 + \cdots + x_n y_n} |y\rangle$$

for every $x \in \{0,1\}^n$. We also use the shorthand

$$x \cdot y = \sum_{i=1}^{n} x_i y_i \pmod 2$$

so that we may write

$$H^{\otimes n}|x\rangle = \frac{1}{\sqrt{2^n}} \sum_{y \in \{0,1\}^n} (-1)^{x \cdot y} |y\rangle .$$

The fact that $x \cdot y$ is defined modulo 2 is irrelevant for the previous expression (because $x \cdot y$ appears as an exponent of -1), but it will be convenient later to use this shorthand again and have the quantity defined modulo 2.

Using these formulas, the Deutsch-Jozsa Algorithm becomes easier to analyze. The state after the first layer of Hadamard transforms is performed is

$$\frac{1}{\sqrt{2^n}} \sum_{x \in \{0,1\}^n} |x\rangle \left(\frac{1}{\sqrt{2}}|0\rangle - \frac{1}{\sqrt{2}}|1\rangle\right).$$

When the B_f transformation is performed, the state will change to

$$\frac{1}{\sqrt{2^n}} \sum_{x \in \{0,1\}^n} (-1)^{f(x)} |x\rangle \left(\frac{1}{\sqrt{2}}|0\rangle - \frac{1}{\sqrt{2}}|1\rangle\right).$$

Once again we are seeing the *phase kick-back* effect. Now, the last qubit is discarded and n Hadamard transforms are applied. The resulting state is

$$\frac{1}{\sqrt{2^n}} \sum_{x \in \{0,1\}^n} (-1)^{f(x)} \left(\frac{1}{\sqrt{2^n}} \sum_{y \in \{0,1\}^n} (-1)^{x \cdot y} |y\rangle\right),$$

which simplifies to

$$\sum_{y\in\{0,1\}^n} \left(\frac{1}{2^n} \sum_{x\in\{0,1\}^n} (-1)^{f(x)+x\cdot y} \right) |y\rangle \, .$$

Now, all that we really care about is the probability that the measurements all give outcome 0. The amplitude associated with the classical state $|0^n\rangle$ is

$$\frac{1}{2^n} \sum_{x\in\{0,1\}^n} (-1)^{f(x)}.$$

Thus, the probability that the measurements all give outcome 0 is

$$\left| \frac{1}{2^n} \sum_{x\in\{0,1\}^n} (-1)^{f(x)} \right|^2 = \begin{cases} 1 & \text{if } f \text{ is constant} \\ 0 & \text{if } f \text{ is balanced,} \end{cases}$$

so the algorithm works as claimed.