Quantum algorithms (CO 781/CS 867/QIC 823, Winter 2011)
Andrew Childs, University of Waterloo
LECTURE 17: Quantum walk algorithm for formula evaluation

Grover's algorithm for unstructured search can be viewed as evaluating the logical OR of n bits in $O(\sqrt{n})$ quantum queries. It is natural to ask about the quantum query complexity of other Boolean functions. In this lecture we will describe a quantum walk algorithm that evaluates a balanced binary AND-OR tree with n leaves in $O(\sqrt{n})$ quantum queries. In fact, a similar approach shows that the quantum query complexity of evaluating *any* Boolean formula expressed in terms of AND, OR, and NOT gates is $\sqrt{n^{1+o(1)}}$.

Games and formulas

Consider a two-player game between Andrea and Orlando that works as follows. The players alternate turns, with Andrea going first and Orlando going second. On each turn there are d possible moves, and there are a total of k turns. Suppose that the winner is determined by some black box function $f : \{1, 2, \ldots, d\}^k \to \{0, 1\}$ indicating who wins the game ($0 =$ Andrea, $1 =$ Orlando) after a particular sequence of k moves. The goal is to determine who wins the game, assuming Andrea and Orlando both play optimally.

This problem is equivalent to evaluating an AND-OR formula. Consider the tree of moves (a balanced d-ary tree of height k) and let internal vertices evaluate to 0 or 1 depending on who wins the game if the players start from the corresponding (partial) sequence of moves. Orlando wins if he can make any move that leads to the outcome 1, so vertices representing his moves correspond to OR gates. Andrea wins if she can make any move that gives 0—i.e., she only loses if all her moves give 1—so her vertices correspond to AND gates.

What is the query complexity of evaluating this balanced d-ary AND-OR tree? Let us first consider randomized classical algorithms. Notice that it is sometimes possible to avoid evaluating all the leaves: for example, if we learn that one input to an AND gate is 0, then we do not need to evaluate the other inputs to know that the gate evaluates to 0. In the case where all inputs are 1, we must evaluate all of them; but the inputs to an AND gate are given by the outputs of OR gates, and an OR gate evaluating to 1 is exactly the case where it is possible to learn the value of the gate without knowing all of its inputs. Similarly, the hardest input to the OR gate is precisely the output of an AND gate for which it is possible to learn the output without evaluating all inputs.

With these observations in mind, a sensible classical algorithm is as follows. Suppose that to evaluate any given vertex of the tree, we guess a random child and evaluate it (recursively), only evaluating other children when necessary. By analyzing a simple recurrence, one can show that this algorithm uses

$$O\left(\left(\frac{d-1+\sqrt{d^2+14d+1}}{4}\right)^k\right) = O\left(n^{\log_d \frac{d-1+\sqrt{d^2+14d+1}}{4}}\right) \tag{1}$$

queries, where $n = d^k$ is the input size (e.g., for $d = 2$, $O(n^{0.753})$). In fact, it is possible to show that this algorithm is asymptotically optimal. Notice, in particular, that the classical query complexity becomes larger as d is increased with n fixed. In the extreme case where $k = 1$, so that $n = d$, we are simply evaluating the AND gate, which is equivalent (by de Morgan's laws) to evaluating an OR gate, and which we know takes $\Omega(n)$ queries.

A quantum computer can evaluate such games faster if k is sufficiently small. Of course, the $k = 1$ case is solved in $O(\sqrt{n})$ queries by Grover's algorithm. By applying Grover's algorithm recursively, suitably amplifying the success probability, it is possible to evaluate the formula in $\sqrt{n}O(\log n)^{k-1}$ queries, which is nearly optimal for constant k. This can be improved slightly to $O(\sqrt{n}c^k)$ queries for some constant c using a variant of Grover's algorithm that allows noisy inputs. But both of these algorithms are only close to tight when k is constant. Indeed, for very low degree (such as $d = 2$, so that $k = \log_2 n$), nothing better than the classical algorithm was known until 2007. Here we will describe how to solve that problem in only $O(\sqrt{n})$ quantum queries.

NAND formulas

We will find it helpful to re-write the AND-OR tree in terms of NAND gates. Note that $\bar{x} = \text{NAND}(x)$, and

$$\text{AND}(x_1, x_2, \ldots, x_n) = \text{NAND}(\text{NAND}(x_1, x_2, \ldots, x_n)) \tag{2}$$

$$\text{OR}(x_1, x_2, \ldots, x_n) = \text{NAND}(\text{NAND}(x_1), \text{NAND}(x_2), \ldots, \text{NAND}(x_n)) \tag{3}$$

(where the latter follows by de Morgan's laws). Hence we can rewrite any formula using AND, OR, and NOT gates solely in terms of NAND gates. Then the formula can be specified simply by giving a tree in which internal vertices correspond to NAND gates on their children. Furthermore, we can assume that all leaves of the tree correspond to 0 inputs, since a 1 input could be encoded using a NOT gate (a NAND gate on one input).

Notice that when we rewrite an AND-OR tree in terms of NAND gates, negated outputs occur together with negated inputs, and these pairs of successive NOT gates can be eliminated. In other words, the AND-OR tree is equivalent to a balanced d-ary tree of NAND gates, up to complementation of the output and possibly the inputs (where latter depends on whether we have odd or even number of levels k). Thus we can focus solely on evaluating this NAND tree. Letting $\text{NAND}(v)$ denote the value of the NAND subformula under v, we are ultimately trying to compute $\text{NAND}(\text{root})$.

Here we will focus on the case $d = 2$ (the balanced binary tree). Let H be the adjacency matrix of the corresponding tree, with NOT gates added for inputs evaluating to 1. Note that this H can be simulated with queries $= (\text{time})^{1+o(1)}$ using sparse Hamiltonian techniques (it is a graph of maximum degree 3 with the internal edges fixed, and the edges at the leaves specified by the black box). Also, the corresponding discrete-time quantum walk can be simulated using a constant number of queries per step. For either case, our goal is to show that the value of the formula affects the spectrum of H in a way that can be detected using phase estimation.

Spectral analysis (qualitative version)

We would like to understand the spectrum of H. Since this is the adjacency matrix of a tree, its spectrum can be understood recursively. In particular, considering the eigenvalue equation $H|E\rangle = E|E\rangle$ at the vertex v, we have

$$\begin{aligned} \langle v|H|E\rangle &= \sum_{w \in \partial(v)} \langle w|E\rangle && (4) \\ &= \langle p|E\rangle + \sum_c \langle c|E\rangle = E\langle v|E\rangle && (5) \end{aligned}$$

where p is the parent of v, and the sum runs over children c of v (so there are 0, 1, or 2 terms in the sum). In other words,

$$\langle p|E\rangle = E\langle v|E\rangle - \sum_c \langle c|E\rangle. \tag{6}$$

We will see that eigenstates with small values of E play a special role, so let us first consider the eigenstate of H with eigenvalue precisely $E = 0$. Denoting such a state as $|\psi\rangle$, we have

$$\langle p|\psi\rangle = -\sum_c \langle c|\psi\rangle. \tag{7}$$

We claim that these states evaluate the NAND tree in a certain sense. To understand this, let's analyze the case of a single NAND gate.

First consider the case where both inputs are 0. Then NAND(root) = 1, and the adjacency matrix of the graph is

$$H = \begin{pmatrix} 0 & 1 & 1 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \end{pmatrix}. \tag{8}$$

This matrix has eigenvalues $\pm\sqrt{2}$ (with eigenvectors $(\pm\frac{1}{\sqrt{2}}, \frac{1}{2}, \frac{1}{2})$) and 0 (with eigenvector $(0, \frac{1}{\sqrt{2}}, -\frac{1}{\sqrt{2}})$. Notice that the 0 eigenvector has zero overlap on the root. In fact, we could see this without explicitly computing the eigenvectors because, applying (7) at either leaf,

$$\langle \mathrm{root}|\psi\rangle = -\sum_{\text{children } c \text{ of a leaf}} \langle c|\psi\rangle = 0 \tag{9}$$

as a leaf has no children.

Now suppose one (but only one) input is 1. Then we still have NAND(root) = 1. In this case, a direct calculation shows that there is no zero-energy eigenstate. But even without doing that calculation, we can see that there is no zero-energy eigenstate with overlap on the root by applying (7) as above at the leaf connected to the root.

Finally, suppose that both inputs are 1. Then we can see by inspection that there is a zero-energy eigenstate with amplitude $1/\sqrt{3}$ at the root and $-1/\sqrt{3}$ at the leaves, and 0 at the vertices in between.

In general, the idea is to show that zero-energy eigenstates with support on the root exist precisely when the formula evaluates to zero. In particular, let's show

Theorem. *If* NAND(root) $= 0$*, then* $|\langle \mathrm{root}|\psi\rangle| > 0$ *for some* $|\psi\rangle$ *with* $H|\psi\rangle = 0$*. If* NAND(root) $= 1$*, then* $\langle \mathrm{root}|\psi\rangle = 0$ *for any* $|\psi\rangle$ *with* $H|\psi\rangle = 0$*.*

Proof. First consider the case where the NAND is 1. Here we can actually prove the statement where the root is replaced by an arbitrary vertex p with NAND$(p) = 1$. This is helpful both because it gives us a stronger induction hypothesis, and because it will be useful for the case where NAND(root) = 0. Let's use induction on the distance of p from a leaf.

The base case is where some child of p is a leaf. Then by (7) where v is that leaf, we have $\langle p|\psi\rangle = 0$ as in (9).

For the induction step, we know that some child v of p has NAND$(v) = 0$. Then both its children c must have NAND$(c) = 1$. Now by (7), $\langle p|\psi\rangle = -\sum_c \langle c|\psi\rangle = 0$, since $\langle c|\psi\rangle = 0$ by the induction hypothesis.

Now consider the case where $\textsc{nand}(\text{root}) = 0$. Here we use induction on the height of the tree.

In the base case, the tree is a single vertex. Then the only possible state is $|\psi\rangle = |\text{root}\rangle$, which is indeed a zero-energy eigenstate.

For the induction step, let's construct the state $|\psi\rangle$. Consider each child v of the root in turn. Let c be a fixed child of v with $\textsc{nand}(c) = 0$ (note that $\textsc{nand}(\text{root}) = 0$ implies $\textsc{nand}(v) = 1$ for both children v of the root, so v cannot be a leaf, and must have some child c with $\textsc{nand}(c) = 0$). By the induction hypothesis, we know there is some eigenstate of the subtree rooted at c with nonzero overlap on c. Indeed this is also an eigenstate of the subtree rooted at v (assigning zero amplitude to v and to everything under v that is not under c), since by the $\textsc{nand} = 1$ case the eigenstates of the subtree rooted at v have zero overlap on v, which does not affect the eigenvalue condition at c. Finally, construct an eigenstate of the entire tree from the eigenstates of the two subtrees, with the amplitude at the root to be determined. Applying the eigenvalue condition (7), we have $\langle\text{root}|\psi\rangle = -\langle c|\psi\rangle \neq 0$ (which can be rescaled if necessary to be the same for both subtrees) as desired. □

The algorithm

Having understood the qualitative features of the eigenstates of H, we are in a position to define the formula evaluation algorithm. The basic idea is to perform phase estimation on the quantum walk (either continuous- or discrete-time, although the continuous-time case is simpler) starting from $|\text{root}\rangle$. If $\textsc{nand}(\text{root}) = 0$ then we measure a phase of 0 with probability at least $|\langle\text{root}|\psi\rangle|^2$, where $H|\psi\rangle = 0$. But if $\textsc{nand}(\text{root}) = 1$, then we are guaranteed to see a phase different from 0, provided we estimate the phase to high enough precision. To understand the running time, we must make the previous theorem quantitative: how large is $|\langle\text{root}|\psi\rangle|^2$ in the case where $\textsc{nand}(\text{root}) = 0$, and how big is the gap from 0 in the case where $\textsc{nand}(\text{root}) = 1$?

In fact, we will need to modify the walk slightly: we attach a path of length two to the root with a weight of $\alpha = n^{-1/4}$ on the edge to the new root. Notice that because the path has even length, the value of $\textsc{nand}(\text{root})$ is unchanged. (There are other ways to obtain the same effect, such as adding an unweighted path of even length of order $\sqrt{n}$, or adding a balanced binary tree, but the weighted path of length 2 is perhaps the simplest choice.)

Notice that while this algorithm shares many features in common with the quantum walk search algorithms we have seen so far, it does not exactly fall into Szegedy's framework of quantum walk search—for example there is no notion of "marked vertices." Indeed, it is not possible to check locally whether the formula evaluates to 0 or 1; one can show that that $\Omega(\sqrt{n})$ leaves must be evaluated to prove that the formula takes a certain value (this is called the *certificate size* or *nondeterministic complexity* of the formula).

Spectral analysis (quantitative version)

To show that this algorithm works, one must give a quantitative analysis of the spectrum of H showing that phase estimation with precision $O(1/\sqrt{n})$ suffices to distinguish $\textsc{nand}(\text{root}) = 0$ from $\textsc{nand}(\text{root}) = 1$.

Theorem. *If* $\textsc{nand}(\text{root}) = 0$*, then* $|\langle\text{root}|\psi\rangle| \geq \frac{1}{\sqrt{2}}$ *for some* $|\psi\rangle$ *with* $H|\psi\rangle = 0$*. If* $\textsc{nand}(\text{root}) = 1$*, then any* $|\psi\rangle$ *with* $H|\psi\rangle = E|\psi\rangle$*, where* $E < \frac{1}{16\sqrt{n}}$*, has* $\langle\text{root}|\psi\rangle = 0$*.*

Proof. First consider the case where NAND(root) = 0. We already showed how to recursively construct an eigenstate of the tree with nonzero overlap on the root (before adding a weighted path of length 2 to the root). In the un-normalized eigenstate we constructed, every nonzero amplitude was ±1, and there was only nonzero amplitude on every other level of the tree, with support on twice as many vertices each time we move down two levels. Thus the total amplitude squared is at most

$$\sum_{i=0}^{(k/2)-1} 2^i = 2^{k/2} - 1 \tag{10}$$

and, normalizing, we have amplitude at least $2^{-k/4} = n^{-1/4}$ on the root. For the tree with the weighted path of length 2 added to the root, this gives an un-normalized eigenstate with $|\langle \mathrm{root}|\psi\rangle| = |\langle c|\psi\rangle|/\alpha \geq 1$ (where c is the grandchild of root, namely the root of the original tree). After normalizing the final state, we find an overlap at least $1/\sqrt{2}$ on the root.

Now consider the case where NAND(root) = 1. Then one can show the following.

Lemma. *Let c be the root of the original tree, and v its parent (the child of the* root*). Suppose* NAND$(c) = 1$*. Then for any eigenstate $|E\rangle$ with $E < \frac{1}{16\sqrt{n}}$ (assuming $E > 0$ without loss of generality, since the graph is bipartite), we have $0 > \frac{\langle c|E\rangle}{\langle v|E\rangle} \geq -4\sqrt{n}E$.*

Given this lemma, we claim there is a contradiction if $E < \frac{1}{16\sqrt{n}}$ and $\langle \mathrm{root}|E\rangle \neq 0$. We are assuming NAND(root) = 1, so NAND$(v) = 0$ and NAND$(c) = 1$. Now consider the form of $|E\rangle$ as determined by the eigenvalue condition. Since we are considering $E \neq 0$, we need to use the general eigenvalue condition (6). Applying H at the root gives $\alpha\langle v|E\rangle = E\langle \mathrm{root}|E\rangle$, and applying H at v gives

$$\alpha\langle \mathrm{root}|E\rangle + \langle c|E\rangle = E\langle v|E\rangle \tag{11}$$

so

$$\frac{\langle c|E\rangle}{\langle v|E\rangle} = E - \alpha\frac{\langle \mathrm{root}|E\rangle}{\langle v|E\rangle} \tag{12}$$

$$= E - \frac{\alpha^2}{E}. \tag{13}$$

Now by the Lemma, we have

$$E - \frac{\alpha^2}{E} \geq -4\sqrt{n}E \tag{14}$$

and since $\alpha^2 = \frac{1}{\sqrt{n}}$ and $\sqrt{n}E < \frac{1}{16}$, we have

$$E \geq \frac{1}{\sqrt{n}E} - 4\sqrt{n}E \tag{15}$$

$$\geq 16 - \tfrac{1}{4} \tag{16}$$

which is a contradiction, since $E < \frac{1}{16\sqrt{n}}$. □

It remains to establish the lemma. The analysis of the recursion is somewhat intricate, so we omit the details here, and only describe the main idea. The strategy is to compute amplitude ratios recursively from the leaves toward the root. Dividing the eigenvalue condition (6) by $\langle v|E\rangle$, we have

$$\frac{\langle p|E\rangle}{\langle v|E\rangle} = E - \sum_c \frac{\langle c|E\rangle}{\langle v|E\rangle}. \tag{17}$$

We claim that for small enough eigenvalues ($E < \frac{1}{16\sqrt{n}}$), the ratio $\frac{\langle p|E\rangle}{\langle v|E\rangle}$ is small and positive if $\textsc{nand}(v) = 0$, and negative with large absolute value if $\textsc{nand}(v) = 1$.

A leaf v has $\textsc{nand}(v) = 0$. Since it has no children, (17) gives $\frac{\langle p|E\rangle}{\langle v|E\rangle} = E$, which is indeed small.

For a non-leaf vertex with $\textsc{nand}(v) = 0$, both children c of v have $\textsc{nand}(c) = 1$, so we have

$$\begin{aligned} \frac{\langle p|E\rangle}{\langle v|E\rangle} &= E - 2\left(-\frac{1}{\text{big}}\right) && (18) \\ &= E + 2 \times \text{small} && (19) \end{aligned}$$

which is small.

If $\textsc{nand}(v) = 1$, then at least one child c of v has $\textsc{nand}(c) = 0$. This could be an only child, or the other child c' could have $\textsc{nand}(c') = 0$ or $\textsc{nand}(c') = 1$; the weakest bound comes from the latter case. Here equation (17) gives

$$\begin{aligned} \frac{\langle p|E\rangle}{\langle v|E\rangle} &= E - \frac{1}{\text{small}} - \left(-\frac{1}{\text{big}}\right) && (20) \\ &\leq E - \text{big} + \text{small} && (21) \end{aligned}$$

which is negative, with large absolute value.

There are at least two ways of making this rigorous. The original paper [FGG] shows that an appropriate amplitude ratio ($\frac{\langle p|E\rangle}{\langle v|E\rangle}$ for $\textsc{nand}(v) = 0$ and $\frac{\langle v|E\rangle}{\langle p|E\rangle}$ for $\textsc{nand}(v) = 1$) can only slightly more than double in magnitude when moving two levels from the leaves toward the root; hence the ratio can grow from E to at most of order $\sqrt{n}E$. The subsequent paper [ACRSZ] (which also considers more general formulas) states explicit bounds on the amplitude ratios at arbitrary vertices of the tree and establishes these bounds by induction, moving one level at a time.