Quantum algorithms (CO 781/CS 867/QIC 823, Winter 2013)
Andrew Childs, University of Waterloo

# LECTURE 16: Span programs

Having discussed lower bounds on quantum query complexity, we now turn our attention back to upper bounds. The framework of *span programs* is a powerful tool for understanding quantum query complexity. Span programs are closely related to the quantum adversary method, and can be used to show that the (generalized) adversary method actually characterizes quantum query complexity up to constant factors.

For simplicity, we restrict our attention to the case of a (possibly partial) Boolean function $f\colon S \to \{0,1\}$ where $S \subseteq \{0,1\}^n$. Many (but not all) of the considerations for this case generalize to other kinds of functions.

## The dual of the adversary method

Recall that the adversary method defines a quantity

$$\mathrm{Adv}^{\pm}(f) := \max_{\Gamma} \frac{\|\Gamma\|}{\max_{i\in\{1,\ldots,n\}} \|\Gamma_i\|} \tag{1}$$

such that $Q(f) = O(\mathrm{Adv}^{\pm}(f))$. Although not immediately obvious from the above expression, it can be shown that $\mathrm{Adv}^{\pm}(f)$ is the value of a *semidefinite program* (SDP), a kind of optimization problem in which a linear objective function is optimized subjected to linear and positive semidefiniteness constraints.

Unfortunately, the details of semidefinite programming are beyond the scope of this course. For a good introduction in the context of quantum information, see Lecture 7 of Watrous's lecture notes on Theory of Quantum Information.

A useful feature of SDPs is that they can be solved efficiently. Thus, we can use a computer program to find the optimal adversary lower bound for a fixed (finite-size) function. However, while this may be useful for getting intuition about a problem, in general this does not give a strategy for determining asymptotic quantum query complexity.

Another key feature of SDPs is the concept of semidefinite programming duality. To every primal SDP, phrased as a maximization problem, there is a dual SDP, which is a minimization problem. Whereas feasible solutions of the primal SDP give lower bounds, feasible solutions of the dual SDP give upper bounds. The dual problem can be constructed from the primal problem by a straightforward (but sometimes tedious) process. Semidefinite programs satisfy *weak duality*, which says that the value of the primal problem is at most the value of the dual problem. Furthermore, almost all SDPs actually satisfy *strong duality*, which says that the primal and dual values are equal. (In particular, this holds under the *Slater conditions*, which essentially say that the primal or dual constraints are strictly feasible.)

To understand any SDP, one should always construct its dual. Carrying this out for the adversary method would require some experience with semidefinite programs, so we simply state the result here. The variables of the dual problem can be viewed as a set of vectors $|v_{x,i}\rangle \in \mathbb{C}^d$ for all inputs $x \in S$ and all indices $i \in [n] := \{1,\ldots,n\}$, for some dimension $d$. For $b \in \{0,1\}$, we define the $b$-complexity $C_b := \max_{x\in f^{-1}(b)} \sum_{i\in[n]} \||v_{x,i}\rangle\|^2$. Since strong duality holds, we have the following.

**Theorem.** *For any function $f\colon S \to \{0,1\}$ with $S \subseteq \{0,1\}^n$, we have*

$$\mathrm{Adv}^{\pm}(f) = \min_{\{|v_{x,i}\rangle\}} \max\{C_0, C_1\} \tag{2}$$

*where the minimization is over all positive integers $d$ and all sets of vectors $\{|v_{x,i}\rangle \in \mathbb{C}^d \colon x \in S, i \in [n]\}$ satisfying the constraint*

$$\sum_{i\colon x_i \neq y_i} \langle v_{x,i}|v_{y,i}\rangle = 1 - \delta_{f(x),f(y)} \quad \forall\, x \neq y. \tag{3}$$

By constructing solutions of the adversary dual, we place upper bounds on the best possible adversary lower bound. But more surprisingly, one can construct an algorithm from a solution of the adversary dual, giving an upper bound on the quantum query complexity itself.

Observe that if we replace $|v_{x,i}\rangle \to \alpha|v_{x,i}\rangle$ for all $x \in f^{-1}(0)$ and $|v_{y,i}\rangle \to |v_{y,i}\rangle/\alpha$ for all $y \in f^{-1}(1)$, we don't affect the constraints (3), but we map $C_0 \to \alpha^2 C_0$ and $C_1 \to C_1/\alpha^2$. Taking $\alpha = (C_1/C_0)^{1/4}$, we make the two complexities equal. Thus we have

$$\mathrm{Adv}^{\pm}(f) = \min_{\{|v_{x,i}\rangle\}} \sqrt{C_0 C_1}. \tag{4}$$

Note that the constraint (3) for $f(x) = f(y)$, where the right-hand side is zero, can be removed without changing the value of the optimization problem. (For functions with non-Boolean output, one loses a factor strictly between 1 and 2 in the analogous relaxation.) To see this, suppose we have a set of vectors $\{|v_{x,i}\rangle\}$ satisfying the constraint (3) for $f(x) \neq f(y)$ but not for $f(x) = f(y)$. Simply let $|v_{x,i}\rangle = |v'_{x,i}\rangle|x_i \oplus f(x)\rangle$ for all $x \in S$ and all $i \in [n]$. Then $\||v'_{x,i}\rangle\| = \||v_{x,i}\rangle\|$, and for the terms where $x_i \neq y_i$, we have $\langle v'_{x,i}|v'_{y,i}\rangle = \langle v_{x,i}|v_{y,i}\rangle$ if $f(x) \neq f(y)$ and $\langle v'_{x,i}|v'_{y,i}\rangle = 0$ if $f(x) = f(y)$.

## Span programs

The dual of the adversary method is equivalent to a linear-algebraic model of computation known as *span programs*. This model was first studied in the context of classical computational complexity. It was connected to quantum algorithms for formula evaluation by Reichardt and Špalek, and was subsequently related to the adversary method by Reichardt.

A span program for a function $f\colon \{0,1\}^n \to \{0,1\}$ consists of a target vector $|t\rangle \in \mathbb{C}^D$, sets of input vectors $I_{i,b} \subset \mathbb{C}^D$ for all $i \in [n]$ and $b \in \{0,1\}$, and a set of free input vectors $I_{\mathrm{free}} \subset \mathbb{C}^D$. The set of available input vectors for input $x$ is $I(x) := I_{\mathrm{free}} \cup \bigcup_{i\in[n]} I_{i,x_i}$. We say that a span program computes $f$ if $|t\rangle \in \operatorname{span} I(x)$ if and only if $f(x) = 1$.

The complexity of a span program is measured by its *witness size*. If $f(x) = 1$, then there is a linear combination of vectors from $I(x)$ that gives $|t\rangle$; the witness size of $x$ is the smallest squared length of the coefficients for any such linear combination. If $f(x) = 0$, then there is a vector that has inner product 1 with $|t\rangle$ that is orthogonal to all available input vectors; the witness size of $x$ is the smallest squared length of the vector of inner products of this vector with *all* input vectors (of course, these inner products are zero for the available input vectors). The witness size of $f$ is the largest witness size of any $x \in S$, or equivalently, the geometric mean of the largest witness sizes of 0- and 1-inputs.

The smallest witness size of any span program computing $f$ is precisely $\mathrm{Adv}^{\pm}(f)$, and there is a close relationship between span programs and dual adversary solutions. Given a dual adversary

solution with vectors $|v_{x,i}\rangle$, one can construct a matrix whose rows are the vectors $\bigoplus_{i\in[n]}\langle\bar{x}_i|\langle v_{x,i}|$. Take the columns of this matrix in block $i$ and subblock $b$ to be the vectors in $I_{i,b}$, let the target vector be the all ones vector, and let there be no free input vectors. It can be shown that this gives a span program for $f$ whose witness size is exactly the complexity of the dual adversary solution. Furthermore, every span program can be put into a canonical form for which this translation can be reversed to produce a dual adversary solution: taking the vectors of a canonical span program to be the columns of a matrix, the rows give dual adversary vectors for $x \in f^{-1}(0)$ and the witness vectors for $x \in f^{-1}(1)$ give the remaining dual adversary vectors. For more detail on this translation, see Lemma 6.5 of arXiv:0904.2759 (see the rest of that paper for more than you ever wanted to know about span programs).

We focus on dual adversary solutions here, as these are simpler to work with for the applications we consider. However, for other applications it may be useful to work directly with span programs instead; in particular, (non-canonical) span programs offer more freedom when trying to devise upper bounds.

## Unstructured search

We now give a simple example of an optimal dual adversary solution, namely for unstructured search. Let $f\colon S \to \{0,1\}$ be defined by $f(x) = \textsc{or}(x)$ with $S = \{x \in \{0,1\}^n\colon |x| \leq 1\}$ the set of inputs with Hamming weight at most 1. Take dimension $d = 1$; let $|v_{0,i}\rangle = 1$ for all $i \in [n]$ and $|v_{x,i}\rangle = x_i$. The constraint (3) gives

$$\sum_{i\colon 0\neq(e_j)_i} \langle v_{0,i}|v_{e_j,i}\rangle = \langle v_{0,j}|v_{e_j,j}\rangle = 1 \tag{5}$$

for all $j \in [n]$ (where $e_j \in \mathbb{C}^n$ is the $j$th standard basis vector) and

$$\sum_{i\colon (e_j)_i\neq(e_k)_i} \langle v_{e_j,i}|v_{e_k,i}\rangle = \langle v_{e_j,j}|v_{e_k,j}\rangle + \langle v_{e_j,k}|v_{e_k,k}\rangle = 0 \tag{6}$$

for $j \neq k$, so the constraint is satisfied.

The 0- and 1-complexities of this solution are

$$C_0 = \sum_{i\in[n]} 1 = n \tag{7}$$

$$C_1 = \max_j \sum_{i\in[n]} \delta_{i,j} = 1. \tag{8}$$

Since $\sqrt{C_0C_1} = \sqrt{n}$, we see that $\mathrm{Adv}^{\pm}(f) \leq \sqrt{n}$, demonstrating that the previously discussed adversary lower bound is the best possible adversary lower bound.

It is easy to extend this dual adversary solution to one for the total OR function. For any $x \neq 0$, simply let $|v_{x,i}\rangle = \delta_{i,j}$, where $j$ is the index of any particular bit for which $x_j = 1$ (e.g., the first such bit). Then the constraints are still satisfied, and the complexity is the same. As an exercise, you should work out an optimal dual adversary for AND.

## Function composition

A nice property of the adversary method (in both dual and primal formulations) is its behavior under function composition. Given functions $f\colon \{0,1\}^n \to \{0,1\}$ and $g\colon \{0,1\}^m \to \{0,1\}$, we define

$f \circ g \colon \{0,1\}^{nm} \to \{0,1\}$ by $(f \circ g)(x) = f(g(x_1, \ldots, x_m), \ldots, g(x_{nm-m+1}, \ldots, x_{nm}))$. Here we focus on upper bounds, for which we have the following.

**Theorem.** $\mathrm{Adv}^{\pm}(f \circ g) \leq \mathrm{Adv}^{\pm}(f)\,\mathrm{Adv}^{\pm}(g)$.

*Proof.* Let $\{|v_{x,i}\rangle \colon x \in \{0,1\}^n, i \in [n]\}$ be an optimal dual adversary solution for $f$, and let $\{|u_{y,j}\rangle \colon y \in \{0,1\}^m, j \in [m]\rangle\}$ be an optimal dual adversary solution for $g$. Let $y = (y^1, \ldots, y^n)$ where each $y^i \in \{0,1\}^m$. Then define

$$|w_{y,(i,j)}\rangle = |v_{g(y),i}\rangle \otimes |u_{y^i,j}\rangle \tag{9}$$

where $g(y)$ denotes the vector with $g(y)_i = g(y^i)$.

We claim that this is a dual adversary solution for $f \circ g$. To see this, we compute

$$\sum_{(i,j) \colon y^i_j \neq z^i_j} \langle w_{y,(i,j)} | w_{z,(i,j)} \rangle = \sum_{i \in [n]} \langle v_{g(y),i} | v_{g(z),i} \rangle \sum_{j \colon y^i_j \neq z^i_j} \langle u_{y^i,j} | u_{z^i,j} \rangle \tag{10}$$

$$= \sum_{i \in [n]} \langle v_{g(y),i} | v_{g(z),i} \rangle (1 - \delta_{g(y^i),g(z^i)}) \tag{11}$$

$$= \sum_{i \colon g(y^i) \neq g(z^i)} \langle v_{g(y),i} | v_{g(z),i} \rangle \tag{12}$$

$$= 1 - \delta_{(f \circ g)(y),(f \circ g)(z)}. \tag{13}$$

Finally, since $\||w_{y,(i,j)}\rangle\| = \||v_{g(y),i}\rangle\| \cdot \||u_{y^i,j}\rangle\|$, using (2) gives

$$\mathrm{Adv}^{\pm}(f \circ g) \leq \max_y \sum_i \||v_{g(y),i}\rangle\|^2 \sum_j \||u_{y^i,j}\rangle\|^2 \tag{14}$$

$$\leq \mathrm{Adv}^{\pm}(f)\,\mathrm{Adv}^{\pm}(g) \tag{15}$$

as claimed. □

Note that here we needed the constraint (3) in the case where $f(x) = f(y)$.

In particular, combining this with the dual adversary for OR and a similar solution for AND, this shows that $\mathrm{Adv}^{\pm}(f) \leq \sqrt{n}$ for the $n$-input balanced binary AND-OR tree.

## An algorithm from a dual adversary solution

The dual adversary is significant not just because it gives upper bounds on $\mathrm{Adv}^{\pm}(f)$, but because it directly gives a quantum algorithm for evaluating $f$ with quantum query complexity $O(\mathrm{Adv}^{\pm}(f))$. (Note that the construction is not necessarily time-efficient—it may use many more elementary gates than queries—but many known algorithms developed using span programs have subsequently led to explicit, time-efficient algorithms.)

In particular, this shows that the quantum query complexity of the balanced binary AND-OR tree is $O(\sqrt{n})$. This was originally shown, up to some small overhead, using a continuous-time quantum walk algorithm based on scattering theory. The classical query complexity of this problem is $O(n^{\log_2(\frac{1+\sqrt{33}}{4})}) = O(n^{0.753})$, and no better quantum algorithm was known for many years. From the perspective of span programs, the formula evaluation algorithm can be seen a method of recursive evaluation with no need for error reduction.

Similarly to the quantum walk search algorithms we discussed previously, the algorithm for the adversary dual uses a product of two reflections. Let $A = \mathrm{Adv}^{\pm}(f)$, and let $\Delta$ be the projector onto $\mathrm{span}\{|\psi_x\rangle \colon f(x) = 1\}$ where

$$|\psi_x\rangle := \frac{1}{\sqrt{\nu_x}}\Big(|0\rangle + \tfrac{1}{\sqrt{2A}} \sum_{i\in[n]} |i\rangle|v_{x,i}\rangle|x_i\rangle\Big) \tag{16}$$

with $\{|v_{x,i}\rangle\}$ an optimal dual adversary solution. Here the normalization factor is

$$\nu_x = 1 + \frac{1}{2A} \sum_{i\in[n]} \||v_{x,i}\rangle\|^2 \leq \frac{3}{2}. \tag{17}$$

The reflection $2\Delta - I$ requires no queries to implement. Let $\Pi_x = |0\rangle\langle 0| + \sum_{i\in[n]} |i\rangle\langle i| \otimes I \otimes |x_i\rangle\langle x_i|$ be the projector onto $|0\rangle$ and states where the query and output registers are consistent. Then the reflection $2\Pi_x - I$ can be implemented using only two queries to the oracle $O_x$.

The algorithm runs phase estimation with precision $\Theta(1/A)$ on the unitary $U := (2\Pi_x - I)(2\Delta - I)$, with initial state $|0\rangle$. If the estimated phase is 1, then the algorithm reports that $f(x) = 1$; otherwise it reports that $f(x) = 0$. This procedure uses $O(A)$ queries. It remains to see why the algorithm is correct with bounded error.

First, we claim that if $f(x) = 1$, then $|0\rangle$ is close to the 1-eigenspace of $U$. We have $\Pi_x|\psi_x\rangle = |\psi_x\rangle$ for all $x$ and $\Delta|\psi_x\rangle = |\psi_x\rangle$ for $f(x) = 1$, so clearly $U|\psi_x\rangle = |\psi_x\rangle$. Furthermore, $|\langle 0|\psi_x\rangle|^2 = 1/\nu_x \geq 2/3$ for all $x$, so surely $\|\Pi_x|0\rangle\|^2 \geq 2/3$. Thus the algorithm is correct with probability at least $2/3$ when $f(x) = 1$.

On the other hand, we claim that if $f(x) = 0$, then $|0\rangle$ has small projection onto the subspace of eigenvectors with eigenvalue $e^{i\theta}$ for $\theta \leq c/A$, for some constant $A$. To prove this, we use the following:

**Lemma** (Effective spectral gap lemma)**.** *Let $|\phi\rangle$ be a unit vector with $\Delta|\phi\rangle = 0$; let $P_\omega$ be the projector onto eigenvectors of $U = (2\Pi - I)(2\Delta - I)$ with eigenvalues $e^{i\theta}$ with $|\theta| < \omega$ for some $\omega \geq 0$. Then $\|P_\omega \Pi|\phi\rangle\| \leq \omega/2$.*

Let

$$|\phi_x\rangle := \frac{1}{\sqrt{\mu_x}}\Big(|0\rangle - \sqrt{2A} \sum_{i\in[n]} |i\rangle|v_{x,i}\rangle|\bar{x}_i\rangle\Big), \tag{18}$$

where the normalization factor is

$$\mu_x = 1 + 2A \sum_{i\in[n]} \||v_{x,i}\rangle\|^2 \leq 1 + 2A^2. \tag{19}$$

For any $y$ with $f(y) = 1$, we have

$$\langle\psi_y|\phi_x\rangle = \frac{1}{\sqrt{\nu_y \mu_x}}\Big(1 - \sum_{i\colon y_i \neq x_i} \langle v_{y,i}|v_{x,i}\rangle\Big) = 0, \tag{20}$$

so $\Delta|\phi_x\rangle = 0$. Also, observe that $\Pi_x|\phi_x\rangle = |0\rangle/\sqrt{\mu_x}$. By the effective spectral gap lemma, $\|P_\omega|0\rangle\| \leq \sqrt{\mu_x}\omega \leq \sqrt{1 + 2A^2}\,\omega \approx \sqrt{2}A\omega$. Thus, choosing $\omega = \sqrt{\frac{2}{3}} \cdot \frac{1}{A}$ gives a projection of at most $1/\sqrt{3}$, so the algorithm fails with probability at most $1/3$ (plus the error of phase estimation, which can be made negligible, and the small error from approximating $1 + 2A^2 \approx 2A^2$, which is negligible if $A \gg 1$).

It remains to prove the lemma.

*Proof.* We apply Jordan's lemma, which says that for any two projections acting on the same finite-dimensional space, there is a decomposition of the space into a direct sum of one- and two-dimensional subspaces that are invariant under both projections. (We something closely related on the second assignment when computing the spectrum of a product of reflections.)

We can assume without loss of generality that $|\phi\rangle$ only has support on $2 \times 2$ blocks of the Jordan decomposition in which $\Delta$ and $\Pi$ both have rank one. If the block is $1 \times 1$, or if either projection has rank 0 or 2 within the block, then $U$ acts as either $\pm I$ on the block; components with eigenvalue $-1$ are annihilated by $P_\omega$, and components with eigenvalue $+1$ are annihilated by $\Pi$.

Now, by an appropriate choice of basis, restricting $\Delta$ and $\Pi$ to any particular $2 \times 2$ block gives

$$\bar{\Delta} = \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix} \tag{21}$$

$$\bar{\Pi} = \begin{pmatrix} \cos\frac{\theta}{2} \\ \sin\frac{\theta}{2} \end{pmatrix} \begin{pmatrix} \cos\frac{\theta}{2} & \sin\frac{\theta}{2} \end{pmatrix} \tag{22}$$

where $\frac{\theta}{2}$ is the angle between the vectors projected onto within the two subspaces. A simple calculation shows that $(2\bar{\Pi} - I)(2\bar{\Delta} - I)$ is a rotation by an angle $\theta$, so its eigenvalues are $e^{\pm i\theta}$. Since $\Delta|\phi\rangle = 0$, the component of $|\phi\rangle$ in the relevant subspace is proportional to $\binom{0}{1}$, and

$$\left\| \bar{\Pi} \begin{pmatrix} 0 \\ 1 \end{pmatrix} \right\| = \left\| \sin\frac{\theta}{2} \begin{pmatrix} \cos\frac{\theta}{2} \\ \sin\frac{\theta}{2} \end{pmatrix} \right\| = \left|\sin\frac{\theta}{2}\right| \leq \frac{\theta}{2} \tag{23}$$

as claimed. □