Lecture 12: Limitation on quantum advantage for total functions

Rajat Mittal

IIT Kanpur

There are two kinds of Boolean functions, total and partial. Total functions are defined on the entire domain $\{0,1\}^n$. On the contrary, partial functions are defined on a subset $D \subset \{0,1\}^n$ of the entire domain (also known as promise problems).

We have seen multiple algorithms in the quantum computing world.

- Algorithms like Deutsch, Bernstein-Vazirani and Simon gave exponential advantage but solved a promise problem (partial functions).
- Grover and associated algorithm only gave a quadratic advantage but were defined on every input (total functions).

For total functions, we have seen that OR gives a quadratic advantage, but some functions like Parity and Majority do not give any speedup. The natural question is, are there total functions where randomized and quatum query complexity can be separated exponentially?

The answer to this question is negative, and we will prove that even deterministic and quantum query complexity are polynomially related. We know, from as early as early 2000's, that $D(f) = O(Q(f)^6)$ where $D(f)$ and $Q(f)$ are deterministic and quantum queries respectively (see [2]).

We will present a more recent proof with a better exponent.

Theorem 1. *Given a Boolean function $f : \{0,1\}^n \to \{0,1\}$, let $D(f)$ be its deterministic query complexity and $Q(f)$ be its quantum query complexity. Then,*

$$D(f) = O(Q(f)^4).$$

Additionally, this will allow us to see one of the most celebrated result in the recent times in the field of query complexity, Huang's sensitivity theorem [3].

We will start this lecture by introducing *sensitivity related* complexity measures motivated from the lower bounding techniques of the previous lectures.

1 Sensitivity and related measures

Once again, our central object of study is a Boolean function $f : \{0,1\}^n \to \{0,1\}$, where every input x is a string of n bits. The lower bounds we have seen on OR function (and sometimes Parity), do not depend upon the exact function, but the fact that there is an input where changing a bit changes the function. Let $f : \{0,1\}^n \to \{0,1\}$ be a function and x be an input. Let $y_1, y_2, \cdots, y_m$ be inputs such that each of them are Hamming distance 1 from x and $f(y_i) \neq f(x)$. We saw that the lower bound proofs for OR can be modified to imply that the deterministic query complexity is m, randomized query complexity is $\Omega(m)$ and quantum query complexity is $\Omega(\sqrt{m})$.

To formalize this, we define when an input is sensitive at an index. A position in the input x can be indexed by an $i \in [n]$.

Fix a Boolean function $f : \{0,1\}^n \to \{0,1\}$. For an index $i \in [n]$, define x^i to be the input where the i-th bit is flipped in x. An index i is called *sensitive* for input x if $f(x) \neq f(x^i)$. The local sensitivity $\mathrm{s}(f,x)$ at an input x is the number of sensitive indices in the input x. The *sensitivity* of the function, $\mathrm{s}(f)$, is the maximum possible sensitivity $\mathrm{s}(f,x)$ over all inputs x.

Exercise 1. Can you give an example of a function and an input where sensitivity is n (what about 1)?

Exercise 2. Show that sensitivity is a lower bound on deterministic query complexity.

Exercise 3. Can you think of a function whose sensitivity is $o(n)$? Can you think of a function and an input whose sensitivity is 0?

A very similar, but slightly complex, measure is called *block sensitivity*.

For a block of indices $B \subseteq [n]$, define x^B to be the input where *all* bits in block B are flipped. As before, a block B is called *sensitive* for input x if $f(x) \neq f(x^B)$. The local block sensitivity $\mathrm{bs}(f, x)$ at an input x is the maximum number of disjoint sensitive blocks possible in the input x respectively. As you might guess, the block sensitivity of a function is the maximum possible block sensitivity $\mathrm{bs}(f, x)$ over all inputs x.

Exercise 4. Show that block sensitivity is a lower bound on deterministic query complexity.

We moved from sensitivity to block sensitivity because we hope that it might allow us to give better lower bounds.

Exercise 5. Show that $\mathrm{bs}(f, x) \geq \mathrm{s}(f, x)$. Can you show a function and an input where this inequality is strict?

Our first step to prove Theorem 1 will be to upper bound $D(f)$ in terms of known complexity measures, $\mathrm{bs}(f)$ and $deg(f)$.

1.1 Upper bound on deterministic query complexity

The first upper bound will be in terms of degree and block sensitivity. For a Boolean function $f : \{0,1\}^n \to \{0,1\}$,

$$D(f) \leq \deg(f)\mathrm{bs}(f). \tag{1}$$

To prove this upper bound, we need to show a deterministic query algorithm which makes at most $\mathrm{bs}(f)\deg(f)$ many queries. In every iteration we will query at most $\deg(f)$ many variables. After every iteration, we will look at the restricted function (setting the variables according to queries already done). If the function is constant, we output the value, otherwise again query all variables of a maxonomial (a monomial with highest degre) in the next iteration. Since the degree can only decrease after setting a few variables, we query at most $\deg(f)$ many variables in each iteration.

We just need to show that function does become constant in at most $\mathrm{bs}(f)$ many iterations. Suppose not, and x and y be two inputs such that $f(x) = 0$ and $f(y) = 1$. We will show that there are $\mathrm{bs}(f) + 1$ many disjoint sensitive blocks for x. It will shown using the following claim.

Claim. If M is a maxonomial for f, then it contains a sensitive block for every input.

Proof. Let z be an input. Set everything outside of M according to x.

Exercise 6. Show that the restricted function is not constant. What happens to monomial M?

So there is a setting of variables of M (say z') which makes the function take value different from z. The subset of variables of M where z and z' differ form the sensitive block. □

Using the claim, we get a sensitive block for x at every iteration. Another sensitive block will be given by y.

Exercise 7. Prove the above two assertions.

By construction, these blocks are disjoint (subset of variables queried at the i-th iteration). Hence we get $\mathrm{bs}(f) + 1$ many disjoint sensitive blocks at x, contradicting that maximum block sensitivity at any input is $\mathrm{bs}(f)$.

Hence, the algorithm will run for at most $\mathrm{bs}(f)$ many iterations, giving an upper bound of $\deg(f)\mathrm{bs}(f)$ on the deterministic query complexity.

To prove $D(f) = O(Q(f)^4)$, we would like to upper bound $\deg(f)$ and $\mathrm{bs}(f)$ by known lower bounds on $Q(f)$, approximate degree of f and $\lambda(f)$. Here $\lambda(f)$ was the biggest eigenvalue of the sensitivity graph of f. For these lower bounds on $Q(f)$, see the previous lecture.

The proof of Theorem 1 will be completed by two proving two results.

– $\mathrm{bs}(f) \leq \widetilde{\deg}(f)^2$ [4]
– $\deg(f) \leq \lambda(f)^2$ [3]

The next two sections will prove these results.

1.2 Upper bound on Block sensitivity by approximate degree

The aim of this subsection is to show

$$\mathrm{bs}(f) = O(\widetilde{\deg}(f)^2). \tag{2}$$

The result was proved by Nisan and Szegedy [4]. Fortunately, we have seen this result already (almost)!

Exercise 8. Can you remember where?

It is a slight extension of the result that approximate degree of OR_n is $\sqrt{n}$ [5]. The same proof can be used to show that even the approximate degree of Promise-OR_n is also $\sqrt{n}$; where Promise-OR_n is only defined on Hamming weight 0 and 1 inputs and has the same value as OR_n.

Specifically, let f has maximum sensitivity at z, we can consider the *translated function*

$$f'(x) = f(x \oplus z).$$

Convince yourself that f' has a Promise OR structure on $\mathrm{s}(f)$ many bits.

Exercise 9. Show that the approximate degree of f is same as f'. Show that this implies,

$$\mathrm{s}(f) = O(\widetilde{\deg}(f)^2).$$

In other words, if you look at the function OR, it has kind of a flower structure (center at all 0 input and n petals coming out of it). For a function with sensitivity $\mathrm{s}(f)$, a similar *translated* flower structure is there with $\mathrm{s}(f)$ petals.

The bound on block sensitivity, Equation 2, follows by a similar argument where each block corresponds to a petal. Since we can always translate a function without affecting its block sensitivity and approximate degree, assume that f achieves its block sensitivity at 0 (all 0) input.

Formally, let $B_1, B_2, \cdots, B_b$ be maximum disjoint sensitive blocks possible for f where $b = \mathrm{bs}(f, 0) = \mathrm{bs}(f)$. Fix up all other variables which are not in any of the sensitive blocks according to x. We will abuse the notation and call the restricted function f.

Exercise 10. Show that the approximate degree of a restriction is always less than the approximate degree of the original function, so it is enough to show Equation 2 for the restricted function.

Construct a function f' on b variables such that

$$f'(z) = f(x_{1,z_1}, x_{2,z_2}, \cdots, x_{b,z_b}),$$

where $x_{i,0}$ sets variables of block B_i according to x and $x_{i,1}$ sets variables of block B_i according to x^{B_i}.

Exercise 11. Show that the approximate degree of f' is at least $\sqrt{b}$ (notice the flower structure).

The only thing left to show is that $\widetilde{\deg}(f') \leq \widetilde{\deg}(f)$ (to prove Equation 2).

Exercise 12. Show this by converting an approximating polynomial of f to f'. Hint: use the fact that $z_i = 0$ (respectively $z_i = 1$) means all variables in that block are 0 (respectively 1).

2 Huang's result: upper bound on degree using $\lambda(f)$

We are left with showing $\deg(f) \leq \lambda(f)^2$. You will show in the assignment that $\lambda(f) \leq s(f)$. So the relation between degree and lambda will imply $\deg(f) \leq s(f)^2$. This was the intended result of the breakthrough by Huang [3].

The quantity $\lambda(f)$, also called spectral sensitivity of f, was introduced in [3] and was formalized in [1].

First, we remind you of the spectral sensitivity for a Boolean function $f : \{0,1\}^n \to \{0,1\}$. To define spectral sensitivity, we need the concept of *sensitivity graph* of the function f, a subgraph of Boolean hypercube.

Exercise 13. What is a Boolean hypercube (as a graph)?

The sensitivity graph of f, say G_f, is a subgraph of Boolean hypercube, i.e., there are 2^n vertices (for each input). An edge x, y is present in G_f iff $f(x) \neq f(y)$ and x, y is an edge in Boolean hypercube (they have Hamming distance 1).

Exercise 14. Find a function f whose sensitivity graph is the Boolean hypercube itself.

Exercise 15. How many edges are there in the sensitivity graph of OR_n.

Exercise 16. Show a subgraph of Boolean hypercube which is not a sensitivity graph for any function f.

We are interested in the eigenvalues of the adjacency matrix, say A_f, of the graph G_f. We first notice that the graph G_f is bipartite.

Exercise 17. Show that Boolean hypercube is bipartite.

That means, if u is an eigenvalue of G_f, then so is $-u$ (assignment). That means we can talk about the maximum eigenvalue (without clarifying if absolute value needs to be taken before taking maximum).

The spectral sensitivity of f, called $\lambda(f)$, is the maximum eigenvalue (also called spectral norm) of the adjacency matrix of G_f.

Since the eigenvalue of a matrix is bounded by the maximum row sum (why), $\lambda(f) \leq \mathrm{s}(f)$. For $\lambda(f) \leq \widetilde{\deg}(f)$, refer to [1]. This completes the relationships given in Figure **??**.

The main result of this section is the following upper bound on $\deg(f)$ in terms of λ settling sensitivity conjecture.

Theorem 2 ([3]).
For any Boolean function $f : \{0,1\}^n \to \{0,1\}$,

$$\deg(f) \leq \lambda(f)^2.$$

The first simplification is that we can assume $\deg(f) = n$. If not, pick the monomial in the polynomial representation of f with highest degree, and set all other variables to some values. For the restricted function, $\deg(f)$ is same but $\lambda(f)$ can only be smaller (assignment).

That means we can assume $\deg(f) = n$ (any counterexample to Theorem 2 can be converted into a counterexample with full degree). In other words, we just need to prove that $\lambda(f) \geq \sqrt{n}$ when $\deg(f) = n$.

What can we say about sensitivity graph of f when $\deg(f) = n$? Define $V_0 = \{x : f(x) = \mathrm{PARITY}(x)\}$ and $V_1 = \{x : f(x) \neq \mathrm{PARITY}(x)\}$.

Exercise 18. Show that $\deg(f) = n$ is equivalent to saying that $|V_0| \neq |V_1|$.

The problems statement changes to, given that $|V_0| > 2^{n-1}$ (if $|V_0| < |V_1|$ then consider $1 - f$), show that $\lambda(f) \geq \sqrt{n}$.

Exercise 19. Show that there is no edge between V_0 and V_1. Inside V_0 (and V_1), the edges are exactly the edges of Boolean hypercube.

This means that the eigenvalues of G_f are union of eigenvalues of the subgraph on V_0 and V_1. We know that inside V_0 and V_1, the edges are exactly like the Boolean hypercube. In other words we are interested in the eigenvalues of the induced subgraph on V_0 and V_1. For any V with more than half the vertices, we need to show that the induced subgraph from Boolean hypercube (say G_V) has eigenvalue more than $\sqrt{n}$. This will finish the proof.

An interesting lemma relates the eigenvalues of the induced subgraph with the eigenvalues of the original graph. It is called *Cauchy's interlacing theorem* [3], we will only use the following special case of it.

Lemma 1. *Let G be a graph on k vertices and its eigenvalues be $\lambda_1 \leq \lambda_2 \cdots \leq \lambda_k$. If G_V is the induced subgraph on V with l vertices, then*

$$\|G_V\| \geq \lambda_l,$$

where $\|G_V\|$ denotes the maximum eigenvalue of G_V.

Proof. The adjacency matrix of G is an $k \times k$ matrix. The eigenvectors corresponding to bigger eigenvalues, $\{\lambda_k, \lambda_{k-1}, \cdots, \lambda_l\}$, span a vector space of dimension $k - l + 1$, say $\mathcal{S}_1$. The vector space corresponding to l standard basis vectors e_v where $v \in V$, say $\mathcal{S}_2$, spans a subspace of dimension l.

Exercise 20. Since the sum of dimensions of $\mathcal{S}_1$ and $\mathcal{S}_2$ is more than k, show that their intersection is non-empty.

For the common vector v, $Av = A_V v$ (where A, A_V are the adjacency matrices of G, G_V respectively), and the length of Av is more than λ_l times the length of v. So, we get

$$\|G_V\| := \|A_V\| \geq \lambda_l.$$

□

The adjacency matrix of Boolean hypercube (say H_n) has dimensions $2^n \times 2^n$. Arrange the eigenvalues of H in increasing order, $\lambda_1 \leq \lambda_2 \leq \cdots \leq \lambda_{2^n}$. From Lemma 1, the maximum eigenvalue of G_V is more than $\lambda_{2^{n-1}+1}$.

What is $\lambda_{2^{n-1}+1}$? You will show in the assignment that the eigenvalues of Boolean hypercube has very simple structure. It has eigenvalue $-n + 2k$ with multiplicity $\binom{n}{k}$.

Exercise 21. What bound will this give on $\|G_V\|$ when $|V| > 2^{n-1}$?

Unfortunately the interlacing theorem applied on H_n doesn't seem to be of much help. It turns out, a small modification of the adjacency matrix of H_n will do the trick. The idea is to introduce a signing of the Boolean hypercube (assign -1 to some of the 1 entries of the matrix), that *compresses* the eigenvalues. In particular, we will try to make half the eigenvalues $\sqrt{n}$ and other half to be $-\sqrt{n}$. Applying interlacing theorem on that signed matrix will give the result.

Proof of Theorem 2. The main idea of the proof is to construct a *signing* of the adjacency matrix of the Boolean hypercube. A signing of a $\{0, 1\}$ matrix is assigning negative sign to some non-zero entries of the matrix. Let A_s be a signing of a $\{0, 1\}$ matrix, then you will show in the assignment

$$\|A\| \geq \|A_s\|.$$

We will construct a signing s of Boolean hypercube such that half of its eigenvalues (2^{n-1} of them) will be $\sqrt{n}$ and the other half will be $-\sqrt{n}$. If A is the adjacency matrix of G_V,

$$\|A_V\| \geq \|(A_s)_V\|.$$

Here A is the adjacency matrix of H_n and A_V denote the induced matrix on the subset V.

By Lemma 1, $\|(A_s)_V\|$ should be greater than the $2^{n-1} + 1$ highest eigenvalue of A_s, which is $\sqrt{n}$.

The only task is to construct the signing with required properties. Notice that we want to have half the eigenvalues $\sqrt{n}$ and other half to be $-\sqrt{n}$. This implies that we want,

$$A_S^2 = nI.$$

(The trace being 0 ensures that there are equal number of $\sqrt{n}$ and $-\sqrt{n}$ eigenvalues).

Looking at every non-diagonal entry of A_S^2, it basically arises from two sums from a 4-cycle in the Boolean hypercube. If (x, y) are at Hamming distance 2 differing at i, j,

$$A_S^2(x,y) = A_S^2(x, x^{\oplus i})A_S^2(y, y^{\oplus j}) + A_S^2(x, x^{\oplus j})A_S^2(y, y^{\oplus i}).$$

This can only be 0, if all 4-cycles have odd number of -1's.

Exercise 22. Convince yourself of the previous statement. Also check that diagonal entries are fine.

In other words, a signing with odd number of -1's will finish the proof. Such a signing is trivial for $n = 2$.

Exercise 23. Can you construct such a signing for $n = 3$?

Formally, the signing can be defined inductively by,

$$(A_1)_s = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, (A_n)_s = \begin{pmatrix} A_{n-1} & I \\ I & -A_{n-1} \end{pmatrix}$$

You can easily show the following properties of this signing by induction.

- $(A_n)_s$ is a signing of H_n (it follows the structure of Boolean hypercube).
- Trace of $(A_n)_s$ is 0.
- $(A_n)_s^2 = nI$.

From the third property, each eigenvalue is either $\sqrt{n}$ or $-\sqrt{n}$. From the trace property, the multiplicity of each eigenvalue is 2^{n-1}. Thus, we have the signing with required property, showing that if V is a subset of vertices of H_n such that $|V| > 2^{n-1}$, then $\|G_V\| \geq \sqrt{n}$.

By the discussion before the proof, this implies that $\lambda(f) \geq \sqrt{n}$ for any f with degree n. □

Huang's result, using the already known relationship between block sensitivity and degree [4], implies that $\mathrm{bs}(f) = O(\mathrm{s}(f)^4)$. We only know a function for which $\mathrm{bs}(f) = \Omega(\mathrm{s}(f)^2)$ [6]. It is an open problem to bridge this gap.

There have been interesting developments after this discovery, as mentioned before, it was proven that $\lambda(f) = O(\widetilde{\deg}(f))$ in [1]. They were able to use this to show that for any Boolean function f, $\deg(f) = O(\widetilde{\deg}(f)^2)$. This is known to be optimal by OR function.

3 Assignment

Exercise 24. Show that $\mathrm{s}(f), \mathrm{bs}(f)$ are $\Theta(n)$ for any function on n variables which only depends upon the Hamming weight of the input.

Exercise 25. Find the $\mathrm{bs}f$ and $\mathrm{s}f$ for the addressing function.

Exercise 26. Can you think of an f and z to separate $\mathrm{s}_z(f)$ and $\mathrm{bs}_z(f)$.

Exercise 27. Suppose A is the adjacency matrix of a bipartite graph. Show that if u is an eigenvalue of A, then so is $-u$.

Exercise 28. Why is the λ of restricted function smaller than the λ of the original function?

Exercise 29. Let H_n be the Boolean hypercube on n elements. Show that H_n has eigenvalue $-n+2k$ with multiplicity $\binom{n}{k}$ for $0 \leq k \leq n$.

Hint: Use induction and structure of the adjacency matrix of Boolean hypercube.

Exercise 30. Just by looking at the eigenvalues of Boolean hypercube and Cauchy's interlacing theorem, you can come up with a statement like: if the degree n coefficient of f is *big enough* then $\lambda(f) \geq \sqrt{n}$. Make this statement precise and prove it.

Exercise 31. Show that if A_s is a signing of A, then

$$\|A\| \geq \|A_s\|.$$

Exercise 32. What is the lambda of PARITY?

Exercise 33. Show that for all Boolean functions f, $\lambda(f) \leq s(f)$.

References

1. S. Aaronson, S. Ben-David, R. Kothari, S. Rao, and A. Tal. Degree vs. approximate degree and quantum implications of huang's sensitivity theorem. *STOC 2021*, 2021.
2. H. Buhrman and R. de Wolf. Complexity measures and decision tree complexity: a survey. *Theoretical Computer Science, Volume 288, Issue 1, Pages 21-43*, 2002.
3. H. Huang. Induced subgraphs of hypercubes and a proof of the sensitivity conjecture. *Annals of Mathematics, Volume 190, Pages 949-955*, 2019.
4. N. Nisan and M. Szegedy. On the degree of boolean functions as real polynomials. *Computational Complexity, volume 4, pages 301–313*, 1994.
5. R. Paturi. On the degree of polynomials that approximate symmetric boolean functions. *STOC 1992*, 1992.
6. D. Rubinstein. Sensitivity vs. block sensitivity of boolean functions. *Combinatorica, Volume 15, Pages 297–299*, 1995.