6.841: Advanced Complexity Theory Fall 2012

Lecture 5 — September 20, 2012

Prof. Dana Moshkovitz *Scribe: Madars Virza*

1 Overview

In the last lecture we looked at lower bounds for constant-depth circuits, proving that PARITY cannot be computed by constant-depth circuits, i.e. $\mathsf{PARITY} \notin \mathsf{AC}_0$.

General circuit lower bounds for explicit functions are quite weak: the best we can prove after years of effort is that there is a function, which requires circuits of size $5n - o(n)$. In this lecture we will examine what happens if we place natural restrictions on a circuit. Namely, we will prove that detecting a clique in a graph requires superpolynomial circuits.

2 Monotone functions and monotone circuits

Definition 1. *A function f is called* **monotone** *if for all $x \leq y$ we have $f(x) \leq f(y)$.*

An alternative definition is that f is monotone if changing an input bit from 0 to 1 cannot change the value of the function from 1 to 0.

We note that many graph properties (defined as a Boolean functions over Boolean adjacency matrices) are monotone, i.e., adding additional edges cannot destroy the property. Examples of such properties include the graph being connected, containing a clique or having Hamiltonian cycle.

Definition 2. *A Boolean circuit is* **monotone** *if it contains AND and OR gates* only.

It is not hard to see that those two definitions are closely related. If function f is computed by a monotone circuit, then f is monotone, because clearly setting a bit cannot unset the value of any wire. The converse also holds: if f is a monotone function, then there exists a monotone circuit that computes it. Cnsider all minterms of f and construct the circuit as OR of ANDs of variables in each midterm.

However, there is no guarantee that monotone circuits that compute a monotone function will be of small size, because a monotone function can have exponentially many minterms. This motivates an interesting question: what functions can be computed by monotone circuits of polynomial size?

3 Razborov's monotone circuit lower bound

In his seminal paper [1] Razborov proved that detecting if a graph contains a clique requires monotone circuits of superpolynomial size.

We can represent each undirected graph of n vertices as a $\binom{n}{2}$ bit vector $(x_{1,2}, x_{1,3}, \ldots, x_{n-1,n})$, such that $x_{i,j} = 1$ if and only if $(i, j) \in G$.

Theorem 3 (Razborov)**.** *For all n and $0 \leq k \leq n$ denote by $\mathsf{CLIQUE}_{k,n}$ a $\binom{n}{2}$ variable Boolean function that outputs 1 if and only if graph represented by x contains a clique of size at least k.*

There exists some constant $\epsilon > 0$ such that for all n and $k \leq n^{1/4}$ function $\mathsf{CLIQUE}_{k,n}$ doesn't have monotone circuits of size $2^{\epsilon\sqrt{k}}$.

Note that this result is almost ideal: if we could prove similar claim for general circuits, then we would have proved $\mathsf{NP} \not\subseteq \mathsf{P/poly}$. Unfortunately, the proof of Razborov's lower-bound doesn't extend to general circuits.

3.1 Proof of Razborov's lower bound

For every $S \subseteq V$ we can define $\mathsf{C}_S(G)$ to be the indicator function that outputs 1 iff S is a clique in G. Then, of course, $\mathsf{CLIQUE}_{k,n}(G) \triangleq \bigvee_{|S|=k} \mathsf{C}_S(G)$.

We will prove our main result by proving two claims:

1. every small monotone circuit that computes $\mathsf{CLIQUE}_{k,n}(G)$ is essentially computing an OR of small number of $\mathsf{C}_S(G)$'s
2. computing a small number of $\mathsf{C}_S(G)$'s is not sufficient to even approximate $\mathsf{CLIQUE}_{k,n}(G)$

We will focus on how well the circuit does on two subproblems:

- sparsest **YES** instances: having k-clique and no other edges
- densest **NO** instances: complete $k-1$-partite graphs, where partitions are chosen of nearly equal sizes

Intuitively, k-cliques form hardest "yes" instances, because to answer 1, the circuit must test that all edges are present; similarly, Turán graphs, the densest K_{k-1}-free graphs, should form the hardest "no" instances, because the circuit cannot cheat by testing cliques of fewer than k vertices, as it will almost certainly detect one, while graph doesn't have k-clique.

More formally, we will define two distributions:

- let **YES** distribution be generated by picking a k vertices out of n at random and placing a clique on the selected vertices
- let **NO** distribution be generated by choosing a function $c : [n] \to [k-1]$ uniformly at random and adding all edges (i, j) for which $c(i) = c(j)$ [1]

and prove that:

[1]Graphs generated in this way are not the densest possible, but with high probability are close and easier to reason about.

1. if C is a monotone circuit of size $S < 2^{\sqrt{n}/2}$ then there exist m sets of vertices $S_i \subseteq [n]$ such that $C(G)$ can be *approximated* by $\bigvee_{i=1}^{m} \mathsf{C}_{S_i}(G)$:

   - $\Pr_{G \leftarrow \mathbf{YES}}[\bigvee_{i=1}^{m} \mathsf{C}_{S_i}(G) \geq C(G)] > 0.9$
   - $\Pr_{G \leftarrow \mathbf{NO}}[\bigvee_{i=1}^{m} \mathsf{C}_{S_i}(G) \leq C(G)] > 0.9$

   In particular, this results holds if $m = (p-1)^2 \cdot l!$, $p = 10\sqrt{k} \log n$ and $|S_i| \leq l \triangleq \frac{\sqrt{k}}{10}$ for all $1 \leq i \leq m$.

2. $\mathsf{CLIQUE}_{k,n}(G)$ cannot be approximated by $\bigvee_{i=1}^{m} \mathsf{C}_{S_i}(G)$ of the said parameters.

Proof of the first claim uses ideas from the proof of the second claim, so we will begin by proving the second claim.

3.2 $\mathsf{CLIQUE_{k,n}}$ cannot be approximated by small $\mathsf{CLIQUE_S}$'s

Fix $S \subseteq [n]$ and consider two cases depending on whether S is small or large:

(a) $|S| \leq l$. Consider graph G drawn from **NO** distribution. With high probability all vertices in S are in different parts of G: by birthday bound the expected number of collisions ($u, v \in S$ having $c(u) = c(v)$) is $\binom{|S|}{2}\frac{1}{k-1}$, which is less than 0.01 for sufficiently large n. Therefore, by Markov's inequality the probability that all edges from S are present is at least 0.99 and

$$Pr_{G \leftarrow \mathbf{NO}}[\mathsf{C}_S(G) = 1] \geq 0.99]$$

.

(b) $|S| > l$. Consider graph G drawn from **YES** distribution. As G is sparse, a random clique will be hidden from C_S:

$$\Pr_{G \leftarrow \mathbf{YES}}[\mathsf{C}_S(G) = 1] \leq \frac{\binom{n-l}{k-l}}{\binom{n}{k}} \leq \left(\frac{2k}{n}\right)^l \leq n^{-0.7l}$$

Note that we don't actually need part (b) to prove our second claim (we are promised that all S_i are of size at most l), but we will use the result for arbitrary set size to prove our first claim.

3.3 Every small circuit that approximates $\mathsf{CLIQUE_{k,n}}$ is essentially computing bunch of C_S's

We will prove our claim by induction, traversing the circuit and replacing gates by OR's of C_S, starting from bottom (replacing input wires by indicators for 1-cliques) and working our way up, finally replacing the output gate. We will ensure that replaced gate approximates the original gate on $> 1 - \frac{1}{10s}$ fraction of inputs drawn both from YES and NO distribution. Therefore by union bound the probability that all replaced behave as original ones (and therefore the circuit is well-approximied) will be $1 - s \cdot \frac{1}{10s} > 0.9$ as required.

3.3.1 Handling OR gates

If $f = \bigvee_{i=1}^{m} \mathsf{C}_{S_i}$ and $g = \bigvee_{i=1}^{m} \mathsf{C}_{T_i}$ (for $|S_i| \leq l$, $|T_i| \leq l$), we would be tempted to replace gate $f \vee g$ as $f \vee g = \bigvee_{i=1}^{m} \mathsf{C}_{S_i} \vee \bigvee_{i=1}^{m} \mathsf{C}_{T_i}$. However, we cannot afford to double number of sets each time we replace a gate, because the second claim crucially depends on our ability to approximate final gate by small number of circuit indicators.

To reduce this back to OR of at most m indicators, we use the Sunflower Lemma by Erdös and Rado [2]:

Lemma 4 (Sunflower Lemma)**.** *Given at least $(p-1)^l \cdot l!$ sets Z_i of size at most l, it is possible to find choose p of them, $Z_1, \ldots, Z_p$ such that for any $k \neq j$: $Z_k \cap Z_j = \bigcup_{i=1}^{p} Z_i$.*

The name comes from the following resemblance: apart from common intersection (sunflower's center) each two sets are disjoint as are sunflower's petals; so each set is an union of petal and the center.

We can apply Sunflower lemma to reduce OR of $2m$ indicator variables in the following way: as long as we have more than m sets (initially: S_i's and T_i's), we will find a sunflower $Z_1, \ldots, Z_p$ among them and replace $\bigvee \mathsf{C}_{Z_i}$ by a single indicator variable $\mathsf{C}_{\cap Z_i}$.

Note that doing such replacement never damages YES instances (if graph was declared as having a k-clique by some indicator C_{Z_i}, then it is also flagged by all subindicators, notably, $\mathsf{C}_{\cap Z_i}$).

However, such replacement might introduce a mistake for a NO instance. This happens exactly when sunflower's center has a clique, but all petals have missing edges. We claim that probability of this happening is small.

By proof of first claim we know that $\Pr_{G \leftarrow \mathbf{NO}}[\mathsf{C}_{Z_i}(G) = 0] < \frac{1}{2}$. But, then $\Pr_{G \leftarrow \mathbf{NO}}[\mathsf{C}_{Z_i}(G) = 0 | \mathsf{C}_{\cap Z_i}(G) = 1] < \frac{1}{2}$ as center having a clique only increases the probability of having a clique overall. Conditioned on what happens in the center, events $\mathsf{C}_{Z_i}(G)$ are independent (since they depend on values of c on disjoint sets), therefore $\Pr_{G \leftarrow \mathbf{NO}}[\bigwedge \mathsf{C}_{Z_i}(G) = 0 | \mathsf{C}_{\cap Z_i}(G) = 1] < \left(\frac{1}{2}\right)^p$.

3.3.2 Handling AND gates

If $f = \bigvee_{i=1}^{m} \mathsf{C}_{S_i}$ and $g = \bigvee_{i=1}^{m} \mathsf{C}_{T_i}$, then we can open $f \wedge g$ up as follows: $f \wedge g \geq \bigvee_{1 \leq i,j \leq m} \mathsf{C}_{S_i \cup T_j}$ [2]. There are two potential issues: we now have m^2 indicators instead of m, and the cardinalities of sets are now bounded by $2l$ instead of l.

The first problem can be handled in exactly the same way as we handled OR's: as long as we have more than m sets we apply the sunflower trick. The second problem is handled by discarding sets of more than l vertices. Discarding indicators can introduce false negatives, but as we proved in claim 1, they have only a probability of $n^{-0.7l}$ of detecting a YES instance, so we don't lose much.

Filling the details is left as an exercise.

[2] And we actually have equality for our two distributions, as NO instances are maintained and YES instances consist of just one clique

References

[1] A. A. Razborov, *Lower bounds on the monotone complexity of some Boolean functions*, Dokl. Akad. Nauk. SSSR, 281(4):798-801, 1985

[2] P. Erdös, R. Rado, *Intersection theorems for systems of sets*, J. Lond. Math. Soc., 35:85-90, 1960