CSE599s: Extremal Combinatorics October 26, 2011

Lecture 8 Monotone Circuits for Clique Must be Large

Lecturer: Anup Rao

1 Monotone Circuits and Functions

Given the difficulty of proving lower bounds on general circuits, most success stories have to do with restricted classes of circuits. Last time, we considered the setting of linear functions and linear circuits. Today we shall discuss a different kind of restriction.

A *monotone* function $f : \{0,1\}^n \to \{0,1\}$ is a function that has the property that increasing the value of any input can only increase the value of the output. A monotone circuit is a boolean circuit that only uses $\wedge$ and $\vee$ gates (recall $x \wedge y = 1$ if and only if $x = y = 1$, and $x \vee y = 0$ if and only if $x = y = 0$).

Claim 1. *Every monotone function has a monotone circuit of size* 2^n.

Many interesting functions are in fact monotone. For example, the decision version of the CLIQUE problem is a monotone function: given an input graph, output 1 if and only if the graph has a clique of size k. Since this problem is NP-hard, showing that there is no polynomial sized circuit computing it would show that P is not equal to NP.

Remark 2. *By using DeMorgan's law, and the fact that $\wedge, \vee, \neg$ form a boolean basis, you can always rewrite every circuit so that the only negations are applied directly to the inputs, and the rest of the circuit is made of $\wedge$ and $\vee$ gates.*

2 A lower bound for the monotone complexity of CLIQUE

We can represent a graph on n vertices using $\binom{n}{2}$ bits where each bit indicates whether an edge is present or not. Given any graph G represented this way, and any set $S \subseteq [n]$, set

$$K_S(G) = \begin{cases} 1 & \text{if } G \text{ contains a clique on the vertices of } S, \\ 0 & \text{else.} \end{cases}$$

For a parameter k, set

$$K_k = \bigvee_{S \subseteq [n], |S| = k} K_S$$

to be the function that outputs 1 if and only if the input has a clique of size k. Both K_k and K_S are monotone. Beautiful ideas of Razborov (that were built on by others) lead to the following theorem:

Theorem 3. *Let $\epsilon > 0$ be any constant. Then for k large enough in terms of ϵ, if $k < n^{1/3-\epsilon}$, any monotone circuit that computes K_k must have size at least $2^{\sqrt{k}}$.*

2.1 Ready, Steady,

In order to motivate some of the ideas in the proof, let us start by considering a special case. Imagine that we are given a circuit where the gates can be divided into two layers. The bottom layer is all $\wedge$ gates, and the top layer is all $\vee$ gates. In other words, there are sets of edges $E_1, \dots, E_r$, and

$$K_k = \bigvee_i \bigwedge_{e \in E_i} x_e.$$

In this case, we shall try to prove that r must be very large. The first idea to do this is something that is reminiscent of the lower bound on the size of a resolution proof for the pigeon hole principle:

Idea 4. *We restrict the inputs to be cliques.*

Each term $\bigwedge_{e \in E_i} x_e$ can be made much better if we assume that the only inputs that have a k-clique will be those that have edges exactly in one k-clique. Let S_i be the set of vertices that are touched by the edges of E_i. Then under this assumption, we might as well replace each $\bigwedge_{e \in E_i}$ with K_{S_i}! Indeed, if the input does contain a clique, then by our assumption, the edges of E_i are included only if the edges of S_i are included. On the other hand, if the input does not contain a clique, K_{S_i} is always smaller than $\bigwedge_{e \in E_i} x_e$, so our circuit must still work. Thus we are now left with the circuit

$$\bigvee_i K_{S_i}.$$

This starts to make the circuit look like the definition of K_k, for which we know $r = \binom{n}{k}$ must be large. Observe that if $|S_i| < k$ for some i, we can make the circuit fail by putting a clique on S_i. However, if all $|S_i| \geq k$ and there are less than $\binom{n}{k}$ sets, there must be some k-set that does not contain any of the S_i's. The circuit will fail on this k-set.

Simple as that proof was, it actually contains the beginnings of several ideas that are needed in the general case.

2.2 Go!

The idea is to show that any monotone circuit can be approximated by an OR of clique functions as before. Given any monotone circuit of size $2^{\sqrt{k}}$ that computes K_k, we shall show how to approximate each gate f by a function f^* that is either a constant or $\bigvee_i K_{S_i}$, where here $|S_i| \leq \sqrt{k}$, and there are at most t terms in the OR. Note that every monotone function can be written as an OR of AND's, but in general we cannot bound the number of terms by the size of the circuit. For example the function $\bigwedge_{i=1}^n (x_i \vee y_i)$ has 2^n terms when written as an OR of AND's. This is seems like a major obstacle. Razborov hurdles by using the sunflower lemma.

Recall that a sunflower (lecture 7) is a collection of sets, where the i'th set is of the form $Z_i \cup C$, the Z_i's are disjoint and non-empty and C is also disjoint from all the Z_i's. The lemma is that if any family of sets has more than $\ell!(p-1)^\ell$ sets of size at most ℓ, then there must be a sunflower with p petals in the family.

If you have the OR of a sunflower of cliques, then you can replace it with K_C, where C is the core. This can only increase the value of the circuit. Maybe it will increase it too much? To avoid this danger, we restrict our clique-free inputs as well. We shall focus on graphs that are $(k-1)$-partite (and hence do not have a k-clique). Then we have the following lemmas:

Lemma 5. *If G is a random $(k-1)$-partite graph, and $S \subseteq [n]$ is a set with $|S| \le \sqrt{k}$, then* $\Pr[K_S(G) = 1] > 1/2$.

Proof The probability that any fixed pair of vertices is excluded in a random $(k-1)$-partite graph is exactly $1/(k-1)$. Thus the probability that any edge is excluded is at most

$$\binom{\sqrt{k}}{2}/(k-1) = (1/2)(k-\sqrt{k})/(k-1) < 1/2.$$

■

Lemma 6. *If $U_1, \ldots, U_p$ are a sunflower with core C and sets of size $\le \sqrt{k}$, then $\bigvee_i K_{U_i} \le K_C$, and if G is a random $(k-1)$-partite graph,* $\Pr[\bigvee_i K_{U_i}(G) < K_C(G)] < 2^{-p}$.

Proof If there is a clique on any U_i, then there is certainly a clique on C, so $K_C \ge K_{U_i}$. Sample a random $(k-1)$-partite graph by coloring each of the vertices of C with colors from $[k-1]$, and then do the same for the rest of the graph. We have

$$\Pr[K_{U_i}(G) = 1] = \Pr[K_C(G) = 1] \cdot \Pr[K_{U_i}(G) = 1 | K_C(G) = 1].$$

By Lemma 5, $\Pr[K_{U_i}(G) = 1] > 1/2$, so $\Pr[K_{U_i}(G) = 0 | K_C(G) = 1] < 1/2$. Given the coloring on C, the events $K_{U_i}(G) = 1$ are mutually independent. Thus

$$\Pr\left[\bigvee_i K_{U_i}(G) = 0 \middle| K_C(G) = 1\right] < 2^{-p}.$$

■

Lemma 6 means we can always replace a sunflower configuration in our approximators by the core. In order to use the lemma, for a small positive constant $\alpha > 0$, we set

$$t = 2^{(1+\alpha)\sqrt{k}\log k} \ge (\sqrt{k})! \cdot (3\sqrt{k}\log k)^{\sqrt{k}},$$

which will guarantee that (for k large enough), any t sets of size $\sqrt{k}$ contain a sunflower with $p = 3 \cdot \sqrt{k}\log k$ petals. Next we formally define the approximating functions.

- If $f = x_e$ is an input variable corresponding to the edge e, then it computes the function $f^* = K_e$.
- If $f = g \vee h$,
  $$g^* \vee h^* = K_{U_1} \vee \cdots \vee K_{U_c},$$
  where the U_i's are distinct sets. f^* is obtained by repeatedly replacing the sunflowers with their cores until there are no more sunflowers (this may result in $f^* = 1$).
- If $f = g \wedge h$,
  $$g^* \wedge h^* = \bigvee_{i,j} K_{S_i} \wedge K_{T_j}.$$

  In this case, we shall do three approximation steps:

1. a^* is obtained by replacing each term $K_{S_i} \wedge K_{T_j}$ with $K_{S_i \cup T_j}$.
2. b^* is obtained by dropping all terms K_U, where $|U| > \sqrt{k}$ (if all sets are dropped we are left with the function 0).
3. f^* is obtained by repeatedly replacing the sunflowers with their cores until there are no more sunflowers (this may result in $f^* = 1$).

In this way we have defined an approximation f^* for every gate f of the circuit. Let q denote the output gate of the circuit.

The structure of the rest of the proof will be similar to our warm-up case. We shall first show the following two lemmas:

Lemma 7. *If G is a random $(k-1)$-partite graph, then* $\Pr[q^*(G) > q(G)] < 1/2$.

Lemma 8. $q^* \neq 0$.

If $q^* \neq 0$, then Lemma 5 implies that q^* accepts a random $(k-1)$-partite graph with probability at least $1/2$, which implies that $q(G) = 1$ for some $(k-1)$-partite graph, a contradiction. Next we prove the two lemmas.

Proof of Lemma 7 We proceed inductively on the gates of the circuit.

- For an input gate f, $f^* = f$, so the lemma is true.
- If $f = g \vee h$, by Lemma 6, replacing each sunflower by its core does not change its value except with probability 2^{-p}. Since there are at most t^2 replacement steps,

$$\Pr[f^*(G) \neq g^*(G) \wedge h^*(G)] < t^2 2^{-p}.$$

- If $f = g \wedge h$,
  1. $K_{S_i}(G) \wedge K_{T_j}(G) \geq K_{S_i \cup T_j}(G)$, so $a^*(G) \leq g^*(G) \wedge h^*(G)$.
  2. Dropping terms can only decrease the value, so $b^*(G) \leq a^*(G) \leq g^*(G) \wedge h^*(G)$.
  3. By Lemma 6, $\Pr[f^*(G) \neq b^*(G)] < t^2 2^{-p}$.

By the union bound, $\Pr[q^*(G) > q(G)] < 2^{\sqrt{k}} t^2 2^{-p} \leq 2^{\sqrt{k} + 2(1+\alpha)\sqrt{k}\log k - 3\sqrt{k}\log k} < 1/2$.
■

Proof of Lemma 8 We claim that there there is a k-clique G such that $q^*(G) \geq q(G) = 1$. G will be a k-clique that does not contain any set U dropped in approximating the $\wedge$ functions. Indeed, each $\wedge$ can generate at most t^2 sets U that are dropped. Each such U is contained in exactly $\binom{n-\sqrt{k}}{k-\sqrt{k}}$ sets of size k. We have,

$$\frac{2^{\sqrt{k}} t^2 \binom{n-\sqrt{k}}{k-\sqrt{k}}}{\binom{n}{k}} < t^2 \left(\frac{2k}{n-\sqrt{k}}\right)^{\sqrt{k}} < t^2 \left(\frac{4k}{n}\right)^{\sqrt{k}} \leq 2^{\sqrt{k}(2+(3+2\alpha)\log k - \log n)} < 1,$$

for α small enough, since $k < n^{1/3-\epsilon}$.

So such a k-clique G does exist. We shall prove inductively that $f^*(G) \geq f(G)$ for every gate f of the circuit, which will prove the lemma.

- For any input gate f, $f = f^*$.
- If $f = g \vee h$, by Lemma 6, $f^*(G) \geq g^*(G) \vee h^*(G) \geq g(G) \vee h(G)$.
- If $f = g \wedge h$,
  1. For any sets S_i, T_j, $K_{S_i}(G) \wedge K_{T_j}(G) = K_{S_i \cup T_j}(G)$, so $a^*(G) = g^*(G) \wedge h^*(G)$.
  2. Since G does not contain any clique U that has been dropped, $b^*(G) = a^*(G) = g^*(G) \wedge h^*(G)$.
  3. By Lemma 6, $f^*(G) \geq b^*(G) = g^*(G) \wedge h^*(G) \geq g(G) \wedge h(G)$.

■